YMT 313 Web Tasarımı ve Programlama

Dr. Erkan TANYILDIZI Derleme Ders Notları INTERNET ILE ILGILI KISA HATIRLATMA

Internet Nedir?

- Internet, birçok bilgisayar sisteminin birbirine bağlı olduğu, dünya çapında yaygın olan ve sürekli büyüyen bir iletişim ağıdır.
- Internet, insanların her geçen gün gittikçe artan "üretilen bilgiyi saklama/paylaşma ve ona kolayca ulaşma" istekleri sonrasında ortaya çıkmış bir teknolojidir.

Internet Nedir?

- İnsanların kendi arasında etkileştiği, bilgi değiş-tokuşu yapabildiği ve kendi yazısız kuralları olan büyük bir topluluktur.
- Pek çok yararlı bilginin bir tuşa basmak kadar yakın olduğu dev bir kütüphanedir.

Internet Nedir?

- Kişilerin değişik konularda fikirlerini serbestçe söyleyebilecekleri ortamlar barındıran bir demokrasi platformudur.
- Evden alış-veriş, bankacılık hizmetleri, radyo-televizyon yayınları, günlük gazete servisleri vb gibi uygulamaları ile aslında internet aynı zamanda bir hayat kolaylaştırıcıdır.

İnternet adreslerindeki uzantılar

- .com : ticari kurumlar
- .gov : devlet kurumları
- .net : internet servis sağlayıcıları
- .org : kar amacı gütmeyen kuruluşlar, dernekler
- .mil : askeri kurumlar
- .edu : üniversiteler, eğitim kurumları
- .k12 : ilkokul, lise vb eğitim kurumları
- .ac : akademik kurumlar
- .int : uluslararası kurumlar
- .arts : sanat ve kültür kurumları
- .firm : ticari firmalar
- .info: bilgi servisi sunan siteler
- .nom : kişisel
- .name : kişisel
- .rec : eğlence siteleri
- .stor : alışveriş siteleri

İnternet adreslerindeki uzantılar

- Bazı ülke kısaltmaları:
- tr:Türkiye,
- jp:Japonya,
- uk:Ingiltere,
- it:İtalya,
- ch:Isviçre,
- ca:Kanada,
- ru:Rusya,
- id:Endonezya,
- nl:Hollanda,
- de:Almanya,
- fr:Fransa,
- il:İsrail,
- no:Norveç,
- se:İsveç,
- br:Brezilya,

INTERNET

 TCP/IP tabanlı herkese açık ağdır. Özel bir ağ olmayıp, kimseye ait değil ve kimse tarafından kontrol edilemez.

• **TCP/IP:** (Transmission Control Protocol / Internet Protocol)

- Ağ ortamında bilgisayarların iletişimini sağlayan protokoldür.
 - İlk olarak Amerikan Savunma Bakanlığı tarafından hazırlanmış ve 1970 in sonlarında genel kullanım için geliştirilmiştir.

Intranet

 Bir firmanın kendi iletişim alanı içerisindeki ağ ortamında bilgisayarlar arası iletişime verilen addır. Internet'in özel bir versiyonudur.

World Wide Web

 Dünyadaki bilgisayarların birbiriyle iletişim kurabildiği, görüntü, ses, veri paylaşımının yapılabildiği global bir ağdır. Bir web browser kullanılarak görüntülenebilen Internet'in grafiksel yüzüdür.

Web Page (Web Sayfası)

 Internet üzerinde görüntülenebilen her dosya web sayfasıdır.
 Web sayfası ASCII karakterleri kullanılarak yazılan HTML denen bir işaretleme dili ile yazılır.

Web Browser (Web Tarayıcısı)

 World Wide Web üzerinde bulunan sayfaları yükleyip görüntülemeyi sağlayan program.

Web Server

 HTML sayfalarını, Web Tarayıcınıza gönderen Internet üzerindeki sunucu makinelerde çalışan programdır. Web Serverlar, HTML, CGI (Common Gateway Interface), ASP (Active Server Page) vs gibi sayfaları istemcilere gönderebilmektedir.

HTML (Hyper Text Markup Language)

 Web Sayfası hazırlama dilidir. Metinlerin görünümünü, konumunu vs. şekillendirmek için metin ve etiketlerle kontrol edilen oldukça basit bir işaretleme dilidir.

- URL (Uniform Resource Locator- Tekbiçimli Kaynak Konumlayıcı)
 - Internet üzerideki adreslerin genel adıdır. http://www.firat.edu.tr/index.htm üniversitemizin web sayfasının URL'sidir. Bu URL üç bölümden oluşur. Soldan sağa okursak:
 1) Protokol (http://, ftp://, news: vb.);
 2) Alan adı (protokolden, ondan sonraki bölü işaretine kadar olan kısım; www.firat.edu.tr gibi);

3) ve dosya yolu (ilk bölü işaretinden sonraki kısım-index.htm).

Hyperlink

Tıklandığında bağlı olduğu diğer bir sayfanın açılmasını sağlayan bir bağlantıdır.

HTTP (Hyper Text Transfer Protocol)

• HTML sayfalarının Web tarayıcınıza aktarılmasında kullanılan protokoldür.

• E-Posta

- Elektronik mektup (e-mail), kişilerin ve kuruluşların internet üzerinden hızlı bir biçimde haberleşmesi için kullanılan bir sistemdir.
- E-posta almak için POP3 ya da IMAP protokolü kullanılır.

POP3 (Post Office Protokol)

- Elektronik posta almakta kullanılan bir protokoldür.
- Oldukça geniş kullanımı olan POP tek bilgisayar kullanımında en iyisidir çünkü offline mesaj erişimini desteklemektedir, mesajlar önce istemciye çekilmekte sonra sunucudan silinmektedir.

IMAP(Internet Mail Access Protocol)

e-posta sunucunuzdaki e-postalarınıza erişebilmeniz için kullanılan bir metottur. Başka bir deyimle istemcide çalışan bir eposta programının uzaktaki e-postaları sanki istemcideymiş gibi erişimine izin verir. Örneğin, IMAP sunucuda saklanan epostalarınıza evinizdeki kişisel bilgisayarınızdan, işyerinizdeki iş istasyonundan ya da seyahatte diz üstü bilgisayarınızdan mesajlarınızı bu bilgisayarlar arasında transfer etmeden ulaşabilirsiniz.

SMTP(Server Mail Transfer Protocol)

- E-posta gönderme protokolü (SMTP), bir e-posta göndermek için sunucu ile istemci arasındaki iletişim şeklini belirleyen protokoldür
- Outlook, Eudora, Kmail, Thunderbird, Evolution, Sylpheed gibi eposta istemcileri, e-postalarınızı gönderilmek üzere sunucunuza iletirken SMTP servisinden faydalanırlar.

• FTP (File Transfer Protocol)

 Internet üzerinden dosya aktarımını sağlayan protokoldür. Örneğin ftp://arsiv.firat.edu.tr Üniversitemizin arşiv FTP sinin URL sidir.

Telnet

- Internet ağı üzerindeki çok kullanıcılı bir makineye uzaktaki başka bir makineden bağlanmak için geliştirilen bir TCP/IP protokolü ve bu işi yapan programlara verilen genel isimdir.
- Örnek olarak Telnet ile bir POP3 sunucuya bağlanıp mailleri okuyabilir veya bir ftp sunucuya bağlanıp ftp klasörlerini araştırabilirsiniz.

ISP (Internet Service Provider=Internet Servis Sağlayıcı)

 Kişiler öncelikle bir ISP'den aldıkları kullanıcı adı ve şifresiyle Internet'e bağlanırlar. Bu bağlantı ISP firmaları (Örneğin Superonline,TTNET gibi) tarafından belli bir ücret karşılığında satılır.

DNS (Domain Name Server=Alan Adı Sunucusu)

 Web sitelerinin alan adlarının hangi IP'ye yönlendirileceğinin saklandığı sunucudur.

World Wide Web

- World Wide Web (ya da kısaca Web) birbirleri ile iletişim kuran iki bileşenden meydana gelir. Bunlar;
 - 1. Web Browser (Client: İstemci)
 - 2. Web Server (Server: Sunucu)

Web Nasıl Çalışır

- 1. İstemci makinenin kullanıcısı Web tarayıcısında bir hyperlink 'e tıklar.
- 2. Browser DNS (Domain Name Server) a bağlanıp istemde bulunduğu sayfanın server IP sini alır.
- 3. Ardından Aldığı IP ye bir çağrı gönderir ve sayfayı ister.
- 4. Web Server da bu isteğe cevap verir ve http üzerinden sayfayı Web Browser a gönderir.

Web Nasıl Çalışır



Web sayfalarının hazırlanması

Programlama %40

Tasarım %40

Yönetim %20 (En uzun süreç)

Nereden başlanır?

- Internet programcılığının birleştirici unsuru HTML'dir.
- HTML; tek başına bir programlama dili olmamakla birlikte, sayfalarımızın web ortamında görüntülenmesini sağlayan (HTTP) protokole bağlı çalışır. Çünkü programlarımızın çıktıları, HTML ile web sayfalarına dönüştürülerek browserlar'da görüntülenir.
- Sadece HTML bilgisi ile işlevleri sınırlı web siteleri hazırlamak mümkün olsa da Internet Programcılığına başlangıç olarak mutlaka HTML'i öğrenmek gerekir.

HTML nedir?

- HTML, Hyper Text Mark-up Language(Hiper Metin İşaretleme Dili)'ın kısaltılmışıdır.
- HTML browserlar tarafından tanınıp yorumlanabilen metin tabanlı bir dildir ve metnin browser tarafından yorumlanması sonucu da web sayfaları elde edilir.
- Yazdığınız kodların browser tarafından alınıp yorumlanabilmesi için, dosyalarınızın uzantısının ".htm" veya ".html" olması gerekir.

HTTP ve HTML

- HTTP, Web Server ile Web Browser 'ın birbirleri ile haberleştiği bir protokoldür.
- HTML dokümanları metin ve etiketlerden meydana gelir. HTML etiketleri açılı parantez içerisinde (<>) yazılır. HTML etiketlerinin çoğu slash(/) karakteri ile biten etiket ile kapatılarak bloklar halinde kullanılır.

• Örnek:

- **HTML** bir metin işaretleme dilidir.
- Yukarıdaki örnekte **HTML** koyu olarak yazılacaktır.

Web grafiklerinin hazırlanması

- Zor olan işlem, HTML bilmek ve sayfaların içine içeriğin yerleştirilmesi değil, bu içeriğin hazırlanması, yani tasarım aşamasıdır.
- Sayfalarda yer alan grafikler genellikle metinlerden daha da zor hazırlanırlar. Çünkü sayfada yer alacak renkler ve grafik tasarımları sitenin genel çizgisini belirleyeceği için, özenle ve uyumlu bir şekilde hazırlanmalıdır.
- Grafik tasarım programları (Corel Draw, Photoshop, Paint Shop Pro, Flash...) ile hazırlanacak site grafiklerinde görselliğin yanında dikkat edilecek bir başka unsur da grafik dosyalarının boyutlarının küçük olmasıdır. Çünkü ne kadar fazla grafik kullanılırsa, site o kadar yavaş açılacaktır ve bir site ne kadar görsel olursa olsun, eğer yavaş açılıyorsa pek de cazip olmayacaktır. Bu tür programlar ile arka fonlar, butonlar, başlıklar ve bannerlar gibi sabit grafikleri oluşturabilirsiniz.

Web alanı ve adresi almak

- Grafikleri ve metinleri hazır hale getirdikten sonra bunları bir HTML editörü ile web sayfası haline getirebiliriz.
- Hazırladığımız web sayfalarını kullanıcılara ulaştırabilmeniz için bunları yerleştireceğiniz Internet üzerinde bir disk alanına yani web alanına (hosting) ihtiyacınız olacaktır.
- Web alanına Internet erişimini, direkt web sitemizin alanını aldığımızda bize verilen adresten ya da daha kolay hatırlanabilir ve kalıcı bir yönlendirme (domain) adresi alarak bu adresi web alanımıza yönlendirerek sağlayabiliriz.

Web alanı seçimi ve dosya aktarımı

- Web alanı konusunda en iyi çözüm yine Internet üzerinde (bedava/ücretli) sabitdisk alanı (host) sağlayan bir sunucu bulup sayfalarınızı oraya yerleştirmektir.
- Web alanı kapasitesi ihtiyaçlara göre belirlenmelidir. Örneğin; sunucunun kullanılan veri tabanını ve programlama dilini desteklemesi gerekir.
- Son olarak, eğer alacağınız alanın FTP desteği varsa, sayfaları FTP programları ile sunucu üzerindeki alana gönderebilirsiniz. FTP kullanamıyorsanız ya da FTP desteği yok ise web üzerinden dosya transferi yapabilmelisiniz.

Ücretsiz Hosting Hizmeti Veren Bazı Sitelerin İncelemeleri

Site	Alan Miktarı	Reklam	FTP Desteği
Kolayweb.com	10-15 MB	Var	Yok
Mynet.com	10 MB	Var	Yok
Geocities.com	12 MB	Var	Var
Freeservers.com	30 MB	Yok	Yok
Brinkster.com	20 MB	Var	Var



Temel Gereksinimler

Sayfa tasarımı için:

- HTML editörü/editörleri
- Grafik tasarım ve gerekirse animasyon programı/programları

Sayfaları Web' de yayınlamak için:

- Sayfaların barındırılacağı (ücretli/ücretsiz) bir Web alanı
- Sorunsuz bir Internet bağlantısı
- Dosya transfer (FTP) programı
- Sayfaların farklı çözünürlükler ve farklı browserlar'da test edilmesi

Web Hazırlama Yazılımları

- Web hazırlamak için birçok araca sahibiz. Bunları üç sınıfta toplayabiliriz.
 - HTML editörleri
 - Web Sitesi geliştirme araçları
 - Web Uygulaması geliştirme araçları

HTML Editörleri

- HTML yazmak için NotePad 'i kullanabileceğimiz gibi yardımcı programlar da vardır. Metin tabanlı, kod yazmayı gerektiren fakat bunun yanısıra rutin bazı işlemleri kolaylaştıran editörlerdir. Bunlardan bir kısmı;
- Hot Dog
- HotMetal Pro
- HomeSite
- 1st Page 2000
- Basit bir metin editörü.

Web Sitesi Geliştirme Araçları

- Bu tür ürünler tek başına bir web sayfası yapmak yerine Web Sitesi hazırlamada kullanılır. Bu grup programlar HTML editörü ile birlikte linkler kontrol edebilen araçlar ve Web Sitesini, Web Server a yayınlayabilecek araçlar içerir. Bu kategorideki programlar görsel, kodlamayla uğraştırmayı gerektirmeyen editörlerdir. Bunlar;
 - Fusion
 - Visual Page
 - Dreamweaver, Flash
 - FrontPage

Web Uygulamaları Geliştirme Araçları

- Web Sitesi yapma araçları sabit içerikli statik Web Sitesi yapmada kullanılır. Web Uygulamaları Geliştirme Araçları bir Veritabanı ile bağlantılı içeriğe sahip web sitesi hazırlamada kullanılır.
- Bu araçları kendi kendini bir veri tabanından güncelleyen dinamik Web Siteleri oluşturmakta kullanırız. Bu araçlardan bazıları;
- ColdFusion
- NetDynamics
- Dreamweaver UtraDev
- Visual InterDev
- Visual Studio.Net, vs...

Hangi programı kullanmalıyım?

- Kolay kullanılabilen,
- Ne görüyorsanız onu alabileceğiniz (What You See Is What You Get-WYSIWYG)
- Site yönetimi ve dosyaların sunucuya aktarılması kolay,
- HTML dışında da farklı dosya tiplerini tanıyabilen,
- Renkli kodlama ile HTML etiketlerini görüntüleyen,
- İyi bir dokümantasyonu olan,
- Şablonlama ve proje desteği veren,
- Özelleştirebilen


- On yıl öncesine kadar görsel olmayan düzenleyiciler piyasasında Homesite'la başa baş giden bu program, yeni sürümlerinin sık sık göçmesi, firmanın yanlış taktikler izlemesi gibi nedenlerle popülerliğini yitirmiştir.
- Artıları: Birçok özelleştirebilme seçeneği, tablo ve frame'lerin yaratılmasında kullanılabilen sihirbazlar, tümleştirilebilen tarayıcı desteği, site yönetimi, ftp istemcisi, renkli kodlama.
- Eksileri: Sık sık göçmesi, zayıf dokümantasyon ve çalışmayı engelleyici ses efektleri.

Allaire Homesite

Piyasadaki en iyi HTML kod düzenleyicisi olduğu iddia edilmektedir.

Artıları: En gelişmiş özelleştirebilme seçeneklerine sahiptir. Bir kod düzenleyicisi olsa da, sistemde IE4+ yüklüyse, sınırlı miktarda görsel tasarıma izin vermektedir. Bütünleşik FTP istemcisi, şablonlama desteği, siteleri projeler altında gruplandırma sistemi, yazım kontrolü, Perl-PHP-ASPVBScript- Java-CSS- SMIL-ColdFusion gibi dilleri otomatik tanıma, Dreamweaver'la bütünleşme, istenen şekilde kodu girintileyebilme, çok iyi bir dokümantasyon, Homesite'la tümleşik çalışan bir sürümüyle gelmesi.

Eksileri: Windows 9x'de bellek kaçırması ve maalesef sadece Windows'ta çalışması

Netscape Composer

- Netscape Gold ve Communicator paketlerinin içinden çıkan bir düzenleyicidir. Genel olarak, sayfalar WYSIWYG modunda düzenlenir, ancak istenirse ekstra HTML kodu da sayfaya eklenebilir.
- Artıları: Sayfaları tek tuşa basarak bir Web sunucusuna aktarabilirsiniz. Genel görünümü bir kelime işlemci programına benzer. Yazım kontrolü desteği İngilizce bir sözlükle beraber gelir, ancak kelimeleri öğrenme özelliği vardır.
- Eksileri: Site yönetimi, dokümantasyon, özelleştirebilme seçenekleri yoktur. Ayrıca, tablo desteği gerçekten zayıf ve kullanması da ayrıca zordur. HTML dışında hiçbir dosya formatını da tanıyamaz.

Macromedia Dreamweaver

Homesite'ın HTML kodu alanındaki liderliğini, WYSIWYG düzenleyiciler alanında paylaşan programdır.

Artıları: Dreamweaver'ın ilk göze çarpan özelliği, görsel düzenleyiciler arasında en iyi tablolama desteğini sunmasıdır. HTML koduna fazla müdahale etmemesi, blokların kolayca seçilebilmesi, seçilen etiketlerin nerdeyse tüm özelliklerinin kolayca ayarlanabilmesi, tümleşik FTP istemcisi ve site yönetimi, çok iyi DHTML ve Frame desteği, Internet üzerinden de yüklenebilen etiketlerle genişletilebilmesi, tabloları katmanlara, katmanları tablolara çevirebilmesi, rehber resim desteği olması ve Homesite'la bütünleşmesidir.

Eksileri: Homesite kadar iyi bir doğrudan HTML kodu yazımı olmaması ve maalesef sadece Windows ve MacOs'da çalışması nedeniyle kullanımı sınırlıdır.

Microsoft Frontpage

Microsoft firmasının ürünüdür. MS Word'e benzerliği nedeniyle birçok kullanıcıya kullanımı kolay gelen bir düzenleyicidir.

Artıları: Frontpage en başta kullanması ve öğrenmesi kolay bir düzenleyicidir. Ayrıca Microsoft'un ISS Web sunucusuyla tümleşmesi çok iyidir. Frontpage görsel arabirimi sayesinde hemen hemen hiçbir HTML kodu yazmadan sayfa oluşturabilmeyi sağlar.

Eksileri: Frontpage, önceden sizin düzenlemiş olabileceğiniz kodları, tekrar düzenler. Bu nedenle sayfa kodlarına elle müdahale etseniz bile, bir sonraki açışınızda bu ince ayarları silebilir

Web Tasarımını Destekleyici Teknolojiler

- İstemci Taraflı Teknolojiler
- Sunucu Taraflı Teknolojiler

İstemci Taraflı Teknolojiler

- Web sayfalarına görsel ve işlevsel zenginlik katmak için geliştirilmiş küçük programcıklardan oluşan scriptler ve HTML'i destekleyen ve tasarımı zenginleştiren teknolojilerdir.
- Örneğin: XML, CSS, JavaScript, Flash uygulamaları vb.

CSS nedir?

- CSS (Cascading Style Sheets-Geçişli Stil Sayfaları), Bir HTML dokümanındaki HTML takılarının sitillerini belirleyen tekniktir.
- CSS Microsoft Word ün stili gibi çalışır. Stili önceden tanımlarız. Daha sonrada sayfadaki elemanlarda stile başvuru yaparsınız. CSS ile madde imlerini, metin fontlarını, kullanılan başlık fontunu vs. değiştirebiliriz.
- CSS kodlarını web sayfası içerisinde 3 şekilde tanımlayabiliriz;
 - 1. HTML takılarının sitillerini, takıların içinde(Local)
 - <h1 style= font-family:Verdana,Arial> CSS öğreniyorum </h1>
 - 2. Sayfanın başında "<style>" takıları arasında (Global)
 - <style>
 - h1.buyukyazilar {font-family:verdana; color:yellow}
 - h1.digerleri {font-family:arial; color:black}
 - h2 {font-family:verdana; color:pink}
 - </style>
 - 3. Link olarak başka bir dosyadan alarak
 - <link rel="stylesheet" type="text/css" href="sitillerim.css">

CSS ile sayfalarınızı yönetin!

- CSS, HTML takılarına, HTML ile yapılamayacak fakat işinizi kolaylaştıracak bir çok özellik katabilir. Örneğin; hemen hemen tüm web sayfalarında gördüğünüz değişik link sitilleri, CSS kullanılarak yapılmıştır.
- CSS ile sayfalarınızı daha hızlı hale getirebilir ve dosyanın yükünü hafifletebilirsiniz. Sayfalarınız için hazırlayacağınız bir CSS dosyasını kullanarak tüm sayfalarınıza aynı özellikleri verebilir, dolayısıyla hız ve tutarlılık sağlamış olursunuz.

XML nedir?

- XML(Extensible Markup Language- Genişletilebilir İşaretleme Dili)
- HTML ile pek çok açıdan benzerlik gösteren bir işaretleme dilidir. HTML için bir alternatif değildir.
- XML işaretler içine gömülü olan verilerin yapılandırması için kurallar tanımlar. Kendi terimlerimizi (vocabulary) tasarlamayı ve kendi verilerimizi temsil etmeyi, web verisinin depolanmasını ve iletilmesini sağlar.
- Verinin tanımlanması ve tarif edilmesi için kullanılır. HTML'deki yapının aksine XML'de kullanılacak olan tag'ler önceden tanımlı değildir.

Java Script nedir?

- Java Script, Netscape tarafından geliştirilmiş bir script dilidir. Düz yazı formundadır.
- Browserlara bağlıdır ve yorumlanır. HTML'nin içine gömülür. Etiketi <script>...</script> şeklindedir.
- Java Script; etkileşimli web sayfası yapımında HTML'nin yetersiz kalması nedeniyle C/C++ dilleri baz alınarak geliştirilen, "düz yazı" formunda bir script (komutlar dizisi) dilidir.
- Java Script kodları herhangi bir metin editörü (Win 95/98 Not defteri vb.) ile düz yazı formunda yazılabilir.

Java Script nasıl çalışır?

- Java Script kodları istemci tarafında, yani ziyaretçinin web tarayıcısı tarafından yorumlanır ve yaptırılmak istenen işlemler gerçekleşir. Ama bu işlemlerin gerçekleşmesi için mutlaka bir olay gereklidir. Örneğin; sayfanın açılması kapanması, ziyaretçinin sayfada herhangi bir yeri tıklaması veya klavyede herhangi bir tuşa basması gibi...
- Java Script kodlarını web sayfası içerisinde 2 şekilde tanımlayabiliriz; HTML kodları içerisinde <SCRIPT>....</SCRIPT> etiketleri arasında yazarak,
- Düz yazı formunda ve .js uzantısı ile farklı bir dosya oluşturup, bu dosyaya HTML içerisinde link vererek:
- <SCRIPT LANGUAGE="javaScript" src="dosyaadi.js"> </SCRIPT>

Flash

- Flash, Macromedia tarafından geliştirilmiş Internet'te etkileşimli vektörel grafik ve animasyon hazırlanmasında kullanılan bir standarttır.
- Flash içerisinde bulunan bütün görüntü, ses, text elemanları flash'a özgü bazı sıkıştırma teknikleri sayesinde çok küçük boyutlara sıkıştırılabilir. Bu sıkıştırma oranları program içerisinden kalitelerini değiştirmek suretiyle ayarlanabilir. Örneğin; 1 birim tutan bir ses/görüntü dosyası 0.1 ile 0.9 arasında boyutlarda son kullanıcıya ulaştırılabilir.

Flash nasıl kullanılır?

- Flash ile en küçük reklam pankartından (banner), en karmaşık animasyonun yapılması, hatta bütün bir web sitesinin oluşturulması mümkündür. Interaktif CD'lerin hazırlanması, oyunlar, veritabanı uygulamaları, PHP ve XML gibi birçok dil ile kullanılabilme, Flash'ın ileri düzey maharetleri arasında sayılabilir.
- Flash ile ses ile biteşlemler (bitmap) birleştirilebilir. Yani bir dosyanın içine müzik yada ses efekti rahatlıkla gömebiliriz. Bu bize etkileşim daha üst seviyelere çekebilme şansını verir.
- Web tasarımcıları Flash'ı çekici, ölçümlendirilebilir, çok az yer kaplayan gezinilebilir yüzeyler, teknik canlandırmalar, uzun süreli animasyonlar ve diğer şaşırtıcı efektleri yapmak amacıyla kullanırlar.

Sunucu Taraflı Teknolojiler

- Genel olarak Common Gateway Interface (Ortak Ağ Geçidi Arabirimi) adı da verilen bu teknolojilerin temel avantajları:
 - Netscape Navigator / Internet Explorer /Opera gibi web tarayıcılarından bağımsız olmaları (platform bağımsızlığı),
 - Kullandığınız teknolojilerin, scriptlerin sunucu tarafında saklı kalması,
 - Kaynakların yönetimi, kontrolü ve uzaktan sistem yönetimi olanağı sağlamaları,
 - Şirketler arası bilgi paylaşımı için ideal bir ortam oluşturmalarıdır.
 - Günümüzde yaygın olarak kullanılan başlıca sunucu taraflı teknolojiler: PHP ve ASP'dir.

PHP (Personal Home Page)

- HTML'in geliştirilmesinden sonra ihtiyaçların ve web olanaklarının çeşitlenmesi ile bazı eksikliklerinin tamamlanma ihtiyacı doğmuştur.
- Programlama PHP mantığı; web sayfası isteminde bulunan kullanıcılara, adres satırından çağırılan dosya içerisindeki PHP ile ilgili satırları işleyerek dosya içeriğini kullanıcının browserının anlayabileceği HTML formatında sunar.
- PHP genel anlamda Server-Side (Sunucu-Taraflı) bir dildir.

PHP'nin avantajları

- Rakipleri ile arasında performans ve fiyat açısından açık farklılar vardır.
- Fiyatı: Php ücretsiz dağıtılmaktadır.
- Hız: 5000 satırlık bir bilgiyi veritabanına aynı makina üzerinde PHP+MySQL
 3 (üç) saniyede girerken, ASP+MsSQL 28 saniyede girmektedir.
- Açık kaynak kodu: Hataların veya yeni özelliklerin güncellenmesi/eklenmesi gibi olaylarda büyük avantajlar sağlamaktadır.
- Kütüphane: Php ile web sayfalarında daha çok kullanıcı etkileşimli kodlar kullanmaktadır. Örnek olarak; Ziyaretçiler için oluşturulan, forum, sayaç, anket gibi kodlarda Php çok rağbet görmektedir.

ASP (Active Server Pages)

- Yine sunucu taraflı bir teknoloji olan ASP, (Active Server Pages/Aktif Sunucu Sayfaları) kullanıcı tarafına etkileşimli, dinamik web sayfaları göndermek için kullanılır.
- Çok geniş bir kullanım alanı olan ASP, scripting diye tabir edilen VBScript, JavaScript gibi her türlü ham kodu kullanabilir, sunduğu kolay anlaşılır doğal dili ile orta derece HTML bilen kimseler tarafından bile kolayca kavranabilir.

ASP-PHP rekabeti

- Günümüzde büyük şirketlerde Internet ve Intranet uygulamalarda, Microsoft firması kendi ürünü olan ASP için büyük çabalar sarf etmektedir.
- Yeni teknolojiler (.Net) ile ASP için yeni zeminler hazırlayarak iş istihdamı konusunda ASP'yi bir adım öne taşımaya çaba göstermektedir.
- Sunucu fiyatları olarak PHP'ye göre üç kat daha fazla oranda olmasına rağmen, büyük şirketler tarafından tercih edilmektedir.

Java nedir?

- Java (yani Java Applet), Sun Microsystems tarafından geliştirilmiş nesneye yönelik programlama dilidir ve işletim sistemlerinden bağımsız olarak çalışır.
- Etiketi <applet>...</applet> şeklindedir.
- Java Applet hazırlayabilmeniz için Sun firmasının ücretsiz olarak dağıttığı JDK (Java Development Kit – Java Geliştirme Ortamı) programını download etmeniz gerekmektedir.
 - <APPLET code=hotbutton.class height=20 width=122>
 - <PARAM NAME="clip" VALUE="">
 - <PARAM NAME="activeshadow" VALUE="fff9e3">
 - <PARAM NAME="shadowcolor" VALUE="ffedaf">
 - <PARAM NAME="align" VALUE="icon">
 - </APPLET>

Bağlantılar (Hyperlinks)

Bağlantılar (Hyperlinks)

- Sayfalara link (bağlantılar) oluşturmak için <a> Anchor (çapa) etiketi kullanılır.
- <a> etiketini kullanarak;
 - Kendi Web Sitenizde başka bir Web Sayfasına http bağlantısı,
 - Başka bir Web Sitesinin bir sayfasına http bağlantısı,
 - Dosya download etmek için bir ftp bağlantısı,
 - Mail mesajı oluşturmak için bir mailto bağlantısı oluşturabilirsiniz.

Bağlantılara göre kullanılacak komutlar:

• ...

- Bu komutla oluşturduğumuz bağlantı ile yeni bir sayfa açabilir, kullanıcıyı farklı bir internet adresine yönlendirebilir, kullanıcının kendisine sunduğunuz bir dosyaya ulaşmasını sağlayabilirsiniz. Yani bu tanıma göre bildiğimiz bağlantıları oluşturmak mümkün.
- <a>... etiketi arasına yazdığımız yazılar bağlantı özelliğine sahip olacaktır, yazının bağlantı olduğu eğer aksi belirtilmemişse browser tarafından altı çizili ve mavi renkli gösterilir.
- Bağlantıya açıklama eklemek için title özelliği kullanılır.
 - Firat Üniversitesi

• meyve resmi

- Bağlantıya tıkladığında browser o anda açık bulunan sayfa ile aynı dizinde bulunan meyve.gif resmini açacaktır.
- a href="midi.zip"> midi dosyası indir
 - Browser kullanıcıya midi.zip dosyasını açmak mı yoksa diske kaydetmek mi istediğini soran bir pencere açar. Bunun sebebi browser htm, txt, jpg, gif,.. uzantılı dosyaları görüntüleyebilirken zip, doc, xls, mp3 gibi dosyaları görüntüleyememesidir.
- 2.sayfaya gitmek için tıklayın
 - Bağlantıya tıklandığında aynı dizinde bulunan sayfa2.html isimli başka bir html dökümanı açılacaktır.

• Farklı dizinlerde bulunan dosyalara bağlantı

- kedi resmi
 çok güzel karanfil
 BMW otomobil sayfası
- İnternet adresine giden link örneği
 - Her şey benim
- Bu ise bir ftp adresine verilen link örneği.
 - tıklayın dosyaları indirin
- Buradaki linke tıklandığında kullanıcının ilgili mail programı açılacak ve mail'in send to (kime) kısmına verdiğimiz mail adresi otomatik olarak yazılacaktır.
 - İletişim

Bağlantılar-Yer İmi

- ... ve ...
- Sayfa içi (dahili) linkleri bu komutu kullanarak hazırlayabiliriz.
- Örneğin sayfanın üst kısmında sayfa indeksini gösteren bir menü olsun. Kullanıcı bu menüde istediği başlığa tıkladığında ilgili konu açılsın.
- Böyle bir sayfa hazırlamak için yapacağımız şeyler :
 - 1 "tıklandığında" açılacak konuyu işaretlemek
 ...
 - 2 browser'a, hazırlayacağımız menüye "tıklandığında" bu işaretli konuya gitmesini bildirmek.

...

Sıralı listeler (ordered list)

1. Sıralı listeler (ordered list)

Sırasız listeler (unordered list)

2. Sirasiz listeler (unordered list)

T anımlama listeleri (definition list)

3. Tanımlama listeleri (definition list)

<u>Sirali Listeler</u>

<u>Sıralı Listeler</u>

....

<u>Sirasiz Listeler</u>

<u>Sırasız Listeler</u>

....

<u>Tanımlama Listeleri</u>

<u>Tanımlama listeleri</u>

....

- Diyelim ki kullanıcı sayfadaki bir linki tıkladığında, başka bir sayfanın belli bir bölümünün açılmasını istiyoruz.
- Bunun için linke tıklandığında açılacak yazıyı <a name>... ile işaretledikten sonra bağlantı

etiketini şu şekilde yazıyoruz:



```
<a href="sayfa2.htm#ilgiliyer"> Bu linke tıklandığında başka bir
sayfanın ilgili kısmı açılacak </a>
<a name="ilgiliyer"> Başka bir sayfadaki linke tıklandığında
burası açıldı </a>
```

Resimlere bağlantı özelliği kazandırmak

 Bunun için resmi yerleştirmek için kullandığımız: etiketini
 <a href>... etiketinin arasına alıyoruz.

- resim.gif tıklanacak resmi, sayfa1.htm resme tıklandığında açılacak sayfayı gösteriyor.
- Border komutu ise resimde bağlantı özelliği olduğunu belirten çerçeveyi kontrol ediyor, 0 (sıfır) değeri bu çerçeveyi tamamen yok eder. Bu komutu değişik sayılarla deneyebilirsiniz.

Target parametresi

 Bağlantının açılacağı pencereyi belirtmek için target parametresi kullanılır.

• ...

target="_blank"	Bağlantı yeni bir pencerede açılır.
target="_self"	Bağlantı aynı pencere içerisinde açılır.
target="_top"	Bağlantı aynı pencere içerisinde en üstten itibaren açılır.
target="_parent"	Açılan bağlantı, o anda açık sayfayı oluşturmuş bir ana sayfa varsa onun yerine konur.
target="çerçeve adı"	Frame komutu ile çerçeve oluşturulmuşsa bağlantının adı verilen çerçevede açılmasını sağlar.("Çerçeveler" konusunda tekrar değineceğiz.)

Resim ve Bağlantılar

- Bir resme hyperlink (bağlantı) eklemek için ve <A> etiketlerini birlikte kullanırız.
- Örnek:
 - HTML> <BODY>
 -
 - image1.htm -- Resim Örneği <br clear="left">

 -
 - image2.htm -- Resmin Align Özelliği <br clear="left">

 -
 - image3.htm -- BR etiketinin clear özelliği <brlear="left">

 -
 - image4.htm -- Width ve Height Özelliği <br clear="left">

 -
 - images.htm -- Sayfanın Kendisi
 - </BODY></HTML>



- ve etiketlerini kullanarak bir resmin belli bölgelerine hyperlink verilebilir. Web Site Geliştirme programları Image Map araçlarına sahiptir.
- Bir resim üzerine harita üç farklı şekilde tanımlanır;
 - Rect (Dikdörtgensel)
 - Circle (Dairesel)
 - Poly (Çokgensel)

Image MAP

AAP> Etiketi

 Resmin üzerinde tıklanabilir alanlar ve tıklandığında gideceği hedef tanımlanır. Bu etiketin name özelliğine bir isim verilir. Bu isim daha sonra haritaya başvuru yapmada kullanılır.

AREA> Etiketi

- <AREA> etiketi ile <MAP> etiket bloğunun arasında tıklanabilir alanlar ve hedef linkler tanımlanır.
- <AREA> etiketinin özellikleri;
 - SHAPE: Tıklanabilir alanın şeklini belirttiğimiz özellik. Recti Circ, Poly değerlerini alabilir.
 - **COORDS:** Bu özelliğe de şeklin koordinatlarını atarız.
 - HREF: Bu özelliğine de hedef Web Sayfasını belirtiriz.

Image MAP

- COORDS Özelliği
- Şekil rect ise; dikdörtkenin sol üzt köşesi ve sağ alt köşesinin koordinatları sıra ile yazılır.
- Şekil circ ise; çemberin merkesinin koordinatı ve daha sonra pixel olarak yarıçapı yazılır.
- **Şekil poly ise;** herhangi bir noktadan başlanıp teker teker koordinatlar sıra ile yazılır.



Image MAP

- Son olarak etiketinin usemap özelliğine haritanın adı yazılarak haritaya çağrı yapılır.
- Örnek: map.htm
- HTML> <HEAD>
- <TITLE>Untitled Document</TITLE></HEAD>
- <MAP name="harita">



- <AREA shape=poly coords="29, 215, 29, 175, 71, 186, 112, 211, 76, 227, 95, 259, 40, 244" href="image2.htm" alt="image2.htm">
- <AREA shape="circle" coords="160, 208, 35" href="image3.htm" alt="image3.htm">
- </MAP>
- BODY>
-
- </BODY> </HTML>



Özel Karakterler

- HTML metin içerisine özel karakterleri yazmak için aşağıdaki format kullanılır. xxx ISO karakter kodu olmak üzere; &#xxx; şeklindedir. Örneğin copyright sembolü için "©" yazılır.
- Buna ilaveten HTML çok sık kullanılan bazı özel karakterler için aşağıdaki gibi kısa yola sahiptir.
 - <!-- non-breaking space () -->
 - " <!-- quote character (") -->
 - < <!-- less-then charakter (<) -->
 - > <!-- greater-then charakter (>) -->
 - & <!-- ampersand charakter (&) -->
 - © <!-- copyright symbold charakter (©) -->
• Tablo oluşturmak için HTML de beş etiket vardır. Bu etiketler;

ЕТІКЕТ		AÇIKLAMA
<table></table>		

 Tablonun başlangıç ve bitimini tanımlamada kullanırız. || | | Tablonun altına ya da üstüne başlık metni eklemede kullanılır. Bunun için Align özelliğine top ya da bottom değeri atanır. |
| | | Tabloya genel bir başlık vermek için kullanılır. |
| | | | | |
 Table Row: Tablo satırı oluşturmada kullanılır. || | | | Table Header: Tabloya başlık hücresi ekler. Başlık hücresi bold ve ortalı yazılır. |
| | | | Table Data: Tablonun satırına veri elemanı (hücre) eklemede kullanılır. |

Sonbahar	Kış	İlkbahar	Yaz
Eylül	Aralık	Mart	Haziran
Ekim	Ocak	Nisan	Temmuz
Kasım	Şubat	Mayıs	Ağustos

Sonbahar	Kış	İlkbahar	Yaz
Eylül	Aralık	Mart	Haziran
Ekim	Ocak	Nisan	Temmuz
Kasım	Şubat	Mayıs	Ağustos

	ÖLÇÜLER	
	Boy	Kilo
1. Zafer	1.77	80
2. Mustafa	1.82	75
3. Osman	1.75	83



 Tablolar satır ve sütunlardan oluşur. Tabloya genel bir başlık atayabiliriz (thead). Her sütun için de kendi başlığını oluşturmak mümkündür. Tablonun sona erdiği satırdan sonraki satıra açıklama koyabiliriz (caption). Ayrıca tablo hücrelerini yanındaki veya altındaki hücrelerle birleştirebiliriz:

Tablo Başlığı (thead)			
Sütun Başlığı #1 #2		Sütun Başlığı #3	Sütun Başlığı #4
hücre	hücre	hücre	hücre
hücre hücre		hücre	hüara
hüd	cre	hücre	nucre
Tablonun alt yazısı(caption)			

 Şimdi basit bir tablo yapmak için gerekli etiketleri öğrenelim. Öncelikle ... etiketlerini yazıyoruz ve arasını doldurmaya başlıyoruz. etiketi ile satırları, etiketi ile de sütunları oluşturuyoruz.





hücre1	<mark>hücre2</mark>
hücre3	<mark>hücre4</mark>

 hücre1

 hücre1

hücre2

hücre2

hücre3

hücre4

hücre4

hücre4

• <TABLE> Etiketi özellikleri

ÖZELLİK	AÇIKLAMA
ALIGN	Tablonun sayfadaki hizalamasını kontrol eder. Left, center veya right olabilir (varsayılanı left tir).
WIDTH	Tablonun genişliğini sabit olarak belirteceksek bu özelliğe pixel olarak bir sayı, sayfanın yüzdesi büyüklüğünde olmasını istiyorsak yüzdeli bir sayı yazılır. (WIDTH= "100" ya da WIDTH="100%")
BORDER	Border yani çerçeve büyüklüğü pixel olarak bildirilir. "0" verilirse çerçevesizdir.
CELLSPACING	Hücreler arasındaki çerçevenin kalınlığını belirtir. Varsayılan olarak çerçeve kalınlığı 1 dir.
CELLPADDING	Hücre çerçevesi ile içeriği arasındaki mesafeyi pixel olarak belirler. Varsayılan mesafe 1 pixeldir.

<TR> Etiketi

• TR etiketi iki tane özelliğe sahiptir. Bunlar;

ÖZELLİK	AÇIKLAMA
ALIGN	Satırdaki hücrelerin içeriğinin yatay hizalamasını kontrol eder. left, right, center olabilir.
VALIGN	Satırdaki hücrelerin içeriğinin dikey hizalamasını kontrol eder. top, middle, bottom olabilir.

<TH> ve <TD> Etiketleri

 Tablo Başlığı (Table Header=TH) ve Tablo Verisi (Table Data=TD) etiketlerinin özellikleri;

ÖZELLİK	AÇIKLAMA	
ALIGN	Hücre içeriğinin yatay hizalamasını kontrol eder.	
VALIGN	Hücre içeriğinin dikey hizalamasını kontrol eder.	
NOWRAP	Kelime aralarında satır sonu kırmalarını pasif yapar. (Word Wrapping)	
ROWSPAN	Bir hücrenin genişletileceği satır sayısını belirtir. Varsayılanı 1 dir.	
COLSPAN	Bir hücrenin genişletileceği sütun sayısını belirtir. Varsayılanı 1 dir.	
WIDTH	Hücrenin pixel olarak genişliği	
HEIGHT	Hücrenin pixel olarak yüksekliği	

Tabloda başlık ve gövde

- Tablolar da HTML sayfasında olduğu gibi başlık (head) ve gövde (body) bölümlerine ayrılabilir.
- Tabloda başlığı <thead> gövdeyi etiketleri arasına yazarız. <caption> etiketi ile ikinci bir açıklama vermek mümkündür.
- Sütun başlıklarına gelince, her bir başlık etiketi ile belirtilir ve bunlar etiketinde olduğu gibi ...
 /tr> arasına yazılır.
- Tabloda satır ve sütunları belirten ve etiketleri ... arasına alınır.

Tahlolar

```
<thead>Tablo Başlığı (thead)</thead>
<caption align="bottom">alt-yaz1 (caption)</caption>
>
 1.Sütun
 2.Sütun
 3.Sütun
>
 hücre1
 hücre2
 hücre3
>
 hücre4
 hücre5
 hücre6
>
 hücre7
 hücre8
 hücre9
alt-yazı (caption)
```

Tablo Baslığı (thead)

1.Sütun	2.Sütun	3.Sütun
hücre1	hücre2	hücre3
hücre4	hücre5	hücre6
hücre7	hücre8	hücre9

- align="..." width="..." height="...">
- border parametresi çerçevenin kalınlığını belirtir. border=0 çerçevenin görünmemesini sağlar.



 cellpadding parametresi hücre içi marj değerini belirtir.
 cellpadding=0 hücre ile içinde bulunan unsurun (metin/grafik) bitişik olmasını sağlar.



 cellspacing parametresi hücreler arası marjı belirler.

hücre1 hücre2	hücre1 hücre1 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hücre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2 hucre2
nücre1	
hücre2	
nücre1	
nücre2	

- align parametresi tabloyu hizalamada kullanılır .
- align=left sola,
- align=right sağa dayalı yapar,
- align=center ortalar .



hücre1

hücre2

```
>
  hücre1
  >
  hücre2
  >
  hücre1
  >
  hücre2
  >
  hücre1
hücre1
  >
hücre2
  hücre2
```

- width ve height parametreleri resimler konusunda gördüğümüz gibi tabloda en ve boy uzunluğunu belirtir.
- Tablonun değer verilmediğinde sahip olduğu normal ölçülerinden küçük değerler verilirse bu değerler dikkate alınmaz, bir başka deyişle tabloyu büyütebiliriz fakat küçültemeyiz.



hücre1

hücre2

 hücre1 <tt> hücre2

hücre1 hücre2 hücre1 hücre2

- align="..." valign="...">
- bgcolor parametresi hücreyi renklendirmede kullanılır.

```
>
    hücre1
   hücre1
   (tr)
    hücre2
hücre2
   >
    hücre3
   hücre4
   >
    hücre4
   \langle /tr \rangle
```

 background parametresi ile hücreye grafik-artalan yerleştirebiliriz.



- width ve height parametreleri ile hücrenin boyutlarını belirleyebiliriz.
- Yalnız height komutu ile tek sütunlu bir tabloda her hücrenin yüksekliğini değiştirebilirken, width komutu ile her hücreyi değiştiremeyiz.
- En büyük width değeri tüm sütun için geçerli olacaktır.
- Aynı şekilde tek satırlı tabloda width değerini her hücre için değiştirebilirken en büyük height değeri tüm satır için geçerli olacaktır.

hücre1

hücre2

hücre3

hücre4

hücre1hücre2hücre3hücre3hücre4

```
hücre1hücre2hücre3hücre4(table border="1" cellpadding="7"><br/>(tr><br/>(td width=40 height=30>hücre1vidth=40 height=30>hücre1vidth=70 height=30>hücre2vidth=90 height=30>hücre3vidth=90 height=30>hücre4vidth=120 height=30>hücre4vidth=>
```

• align parametresi hücre içinde yatay hizalama yapar.

```
hucre1
hucre2
hucre3
hücre1
 hücre2
 hücre3
```

• valign parametresi hücre içinde düşey hizalama yapar.



Hücreleri Birleştirme

- •
- Aynı satırdaki hücreleri birleştirmek için colspan,
- Aynı sütundaki hücreleri birleştirmek için de rowspan parametresini kullanıyoruz.
- Aşağıdaki gibi bir tablo yapımız olsun :



Hücreleri Birleştirme

- A ve B hücrelerini birleştirmek için A hücresine ait etiketine colspan=2 parametresini ekliyoruz ve B hücresine ait B etiketini siliyoruz.
- E F ve G hücrelerini birleştirmek için E hücresine ait etiketine colspan=3 parametresini ekliyoruz ve F ve G hücrelerine ait F , G etiketlerini siliyoruz.



Hücreleri Birleştirme

- Aynı şekilde E ve I hücrelerini birleştirmek için E hücresine ait etiketine rowspan=2 parametresini ekliyoruz ve I hücresine ait I
- C G ve K hücrelerini birleştirmek için C hücresine ait etiketine rowspan=3 parametresini ekliyoruz ve G ve K hücrelerine ait G



- HTML 'in en büyük avantajlarından biri de ekstra boşlukları ve carriage return (satır kırma, paragraf, sekme vs.) karakterleri kaldırarak sayfayı otomatik olarak biçimlendirmesidir.
- Bu durum aynı zamanda en büyük problemdir. Sayfanın şeklini ayarlarken metinleri istediğimiz yerlere konumlandıramayız. Bunun için tabloları kullanabiliriz.

• Örnek:

- <HTML> <BODY background="bg2.gif">
- <TABLE border="0">
- <TR>
- <TD width="10"> </TD>
- <TD> <H1>FIRAT ÜNİVERSİTESİ </H1> </TD>
- <TD>Web Sitemiz Hakkinda</TD>
- </TR>
- <TR>
- <TD width="10">Öğrenciler</TD>
- <TD colspan="2">Üniversitemiz Elektronik ve Bilgisayar alanında deneyimli öğretmenler yetiştirmektedir.</TD>
- </TR>

• <TR>

- <TD>Personel</TD>
- <TD colspan="2">
-
- Bigisayar Sistemleri
- Elektronik
- Kontrol ve Telekom
-
- </TD>
- </TR>

• <TR>

- <TD> E-Mail Göner
- </TD>
- </TR>
- <TR>
- <TD></TD>
- </TR>
- </TABLE>
- </BODY>
- </HTML>



Yukarıdaki gibi bir sayfayı oluşturmanın bir başka yöntemi ise "frame (çerçeve)" veya "div" kullanmaktır.



Frame (Çerçeve) Kullanımı

Frame (Çerçeve) Kullanımı

- Çerçeve (frame) 'yi bir browser penceresinden birden fazla web sayfasını görüntülemek olarak tanımlayabiliriz.
- Çerçeveler (frames), HTML'e sonradan eklenmiş bir özelliktir. Microsoft Internet Explorer 3.0 ile Netscape Navigator 2.0 ve üstü sürümleri frameleri destekler.
- Çerçeve'ler sayfanın bir tarafı sabit kalırken, diğer tarafını, kullanıcının isteğine göre değiştirme imkanı verir.
- Görüntülenen sayfalardan birindeki linkin tıklanması ile başka bir çerçevedeki içeriği değiştirebiliriz.

Frame (Çerçeve) Kullanımı

 Aşağıdaki etiketleri kullanarak ana frame penceresini oluştururuz.

ΕΤΙΚΕΤ		AÇIKLAMA
<frameset></frameset>		Çerçeveli sayfayı belirtmede kullanılır.
<frame/>	yok	Bir çerçeve sayfası için ad ve kaynak belirtmede kullanılır.
<noframe></noframe>		Çerçeveyi desteklemeyen browserların göstereceği alternatif sayfa bu etiket arasına yazılır.

<FRAMESET> Etiketi

- Çerçeveli belge oluşturacağımız zaman, çerçevenin yapısını belirtmek için <FRAMESET> ve <FRAME> etiketlerini kullanırız.
 <FRAMESET> i kullanarak çerçeveleri satırlar ve sütunlar olarak ya da her ikisini kullanacak şekilde tanımlayabilirsiniz.
- <FRAMESET> etiketi iki tane özelliğe sahiptir;
 - COLS: Sütun çerçevenin sayısını ve büyüklüğünü belirler
 - ROWS: Satır çerçevenin sayısını ve büyüklüğünü belirler.
- Aşağıdaki üç metottan birini kullanarak satır ve sütunları tanımlayabiliriz.
 - Sayfa yüksekliği ve genişliğinin bir yüzdesi olarak ("50%" gibi)
 - Piksel cinsinden sabit bir sayı ile ("100" gibi)
 - Kullanılabilir boşluğun kalanını kullanarak ("*" ile)

<FRAMESET> Etiketi

- Aşağıdaki tabloda bir web sayfasının çerçevesinin nasıl tanımlanacağının birkaç örneği vardır.
- <FRAMESET COLS="100,100,*">
- Üç sütunlu çerçeve; ilk ikisi 100 piksel genişliğinde, üçüncüsü ise kalan boşluğun tamamı büyüklüğünde.
- <FRAMESET COLS="100,30%,*">
- Üç sütunlu çerçeve; birincisi 100 piksel, ikincisi browserın %30 u ve üçüncüsü de kalan boşluğun tamamı büyüklüğünde.
- FRAMESET ROWS="20%,*,30%">
- Dört satırlı çerçeve; birincisi web tarayıcısının %20 si yüksekliğinde, ikincisi kullanılabilecek alanın tamamı, üçüncüsü ise tarayıcının %30 u yüksekliğindedir.
- Not: <FRAMESET> etiketi <BODY> ve <HEAD> etiketinin dışında kullanılması nedeni ile diğer çoğu etiketlerden farklıdır.
<FRAME> Etiketi

 <FRAMESET> ile çerçeve sayfası tanımlandıktan sonra her bir çerçeve penceresinin kaynağını belirtmeliyiz. Bunu da
 <FRAME> etiketi ile yaparız. Özellikleri;

ÖZELLİK	AÇIKLAMA	
NAME	Pencerenin adını belirler	
SRC	Pencerede gösterilecek kaynak dosyayı belirler	
FRAMEBORDER	Piksel olarak Çerçeve kalınlığı	
NORESIZE	Bu özellik yazıldığında çullanıcı çerçeveyi boyutlandıramaz.	
SCROLLING	Kaydırma çubuğunun olup olmayacağına işaret eder. Auto, Yes ya da No değerlerini alabilir. Varsayılan Auto dur.	

 <FRAMESET> etiketinin içerisinde her bir satır ya da sütun çerçeve için <FRAME> etiketini kullanmak gerekir.

 Örneğimizde 3 adet html dosyası var. Bunlardan frame.htm dosyası çerçeve komutlarını içeriyor. Bu komutlar browser'a görüntülenmekte olan pencereyi ikiye bölmesini, bunlardan birisinde htm1.htm dosyasını, diğerinde htm2.htm dosyasını görüntülemesini bildiriyor.





frame.htm dosyası:

<html>

```
<head>
<title>Çerçeveler</title>
</head>
```

```
<noframes>
<body>
</body>
</noframes>
```

</html>

 <body bgcolor="..."> etiketi ile htm1.htm dosyasının artalan rengi kırmızı, htm2.htm dosyasının artalanı ise mavi yapılmış, kolaylık olması açısından dosyaların adlarını görüntülemeleri için bunlar da yazılmıştır.

htm1.htm dosyasi:

```
<html>
<head> </head>
<body bgcolor="red">
<font size="7" color="#ffffff">htm1.htm</font>
</body>
</html>
```

```
htm2.htm dosyası:
```

```
<html>
<head> </head>
<body bgcolor="blue">
<font size="7" color="#ffffff">htm2.htm</font>
</body>
</html>
```

- Çerçeve oluşturmada kullandığımız etiket frameset,
- cols ifadesi açılacak çerçevelerin yan yana sütunlar şeklinde olacağını belirtiyor,
- rows yazarsak çerçeveler alt alta satırlar görünümünde açılacaktır.



- cols="*,*" ifadesi ile açılacak çerçevelerin pixel cinsinden ebatlarını belirliyoruz.
- Buraya örneğin cols="140,500" gibi sayı yazılabilir,
- cols="25%,75%" ile browser penceresinin o anki ebadına göre verilen % oranlarına göre şekil alması sağlanabilir,
- * sembolü ile açılacak çerçevenin ebadı browser'a bırakılır.
- cols="140,*" ifadesi ilk pencerenin 140 pixel olacağını ikincinin ise ebadının browser'a bırakıldığını gösteriyor.

- Bu şekilde aynı zamanda açılacak çerçeve sayısını da belirtmiş oluyoruz. cols=".." ifadesine iki değer verirsek bu iki pencere aç anlamındadır, 3 değer 3 pencere açar.
- <frame name="..." src="..."> etiketi çerçevelere, bağlantıların TARGET kısmında kullanabilmek amacıyla isim (name) vermeyi sağlıyor. src pencerede görüntülenecek html dosyasının yerini gösteriyor.
- Açılacak çerçeve sayısı kadar

<frame name="..." src="..."> etiketi kullanıyoruz.

- FRAMESET etiketi ile kullanabileceğimiz diğer parametreler şunlar;
- frameborder="..." (sayı) Sınır çizgisinin kalınlığını (veya çerçeveler arası mesafeyi) belirler.

FRAMESET etiketini HTML dosyanızın </head>...<body> etiketleri arasında kullanmayı unutmayın !

- FRAME etiketi ile kullanabileceğimiz parametreler;
- marginwidth="..." marginheight="..." (sayı) Sayfanın solundaki (marginleft) ve üstündeki (margintop) kenar boşluklarını belirler.
- scrolling=".." (yes, no, auto) Kaydırma çubuklarının durumunu belirler.
- noresize Pencere boyutlarının sabit olmasını sağlar.

- <NOFRAME> Etiketi
- Eğer Web Tarayıcısı FRAME i desteklemiyorsa alternatif gösterilecek sayfa bu etiket arasında oluşturulacak olan <BODY> etiket bloğuna yazılır.
- HTML>
- <HEAD> <title>Çerçeveler </title> </HEAD>
- <frameset cols="200,*">
- <frame name="sol" src="sol.htm">
- <frame name="sag" src="ana.htm">
- </frameset>
- oframes
- BODY>
- <h2>Browserınız çerçeveleri desteklemiyor</h2>
- </BODY>
- </noframes>
- </HTML>

- Yan yana sütünlar ya da alt alta satırlar görünümünde çerçeveler açmayı öğrendik peki tablolarda gördüğümüz hücreleri birleştirme gibi bir özellik çerçevelerde de var mı? maalesef yok, fakat aynı görünümü elde etmek mümkün.
- Bu etkiyi sağlamak için FRAMESET etiketini açıp kaynak dosyaların hepsini belirtmeden yeni bir FRAMESET açıyoruz, içiçe açtığımız bu etiketlerin kaynak dosyalarını eksiksiz olarak belirtip, etiketleri sonlandırıyoruz.

Conserve a management of experiment of experiment of the second of	
htm1.htm	htm2.htm
htm3.htm	htm4.htm







```
<frameset cols="*,*">
<frameset rows="*,*">
    <frame name="a" src="htm1.htm">
        <frame name="b" src="htm2.htm">
        </frameset>
        <frame name="c" src="htm3.htm">
        </frameset>
    </frameset>
```



Çalışma:

 Uygulamada en az üç çerçeveden oluşan sayfa hazırlanıp sol çerçevede linkler olacak ve target özelliğinin farklı değerleri bu örnek üzerinde denenecek...

- <IFRAME> Etiketi
- Bu tag ile HTM kodları arasına bir çerçeve açılıp bu çerçevede başka bir sayfa gösterilebilir.
- Özellikleri:

ÖZELLİK	AÇIKLAMA
SRC	Çerçevede gösterilecek kaynak web sayfası
WIDTH	Çerçevenin genişliği
HEIGHT	Çerçevenin yüksekliği
SCROLLING	Kaydırma çubuğu durumu (yes, no, auto)
FRAMEBORDER	Çerçevenin kalınlığı
ALIGN	Çerçevenin sayfadaki hizalaması

- Örnek:
- BODY>
- IFRAME src="http:\\www.firat.edu.tr" align="center" height="300"
- width="400" frameborder="1" scrolling="yes">
- </IFRAME>
- </BODY>
- </HTML>



- HTML nin en önemli araçlarından biri de bilgi girişinde kullanılan form kontrolleridir.
- Formlar, web sayfası tasarlayan kişi veya şirketlerle internet kullanıcıları arasındaki veri alışverişini sağlamak için ideal bir araçtır.
- Formlar iki yönlü olarak çalışır; web tarayıcı tarafından ekranda oluşturulan görüntü, kullanıcı tarafından doldurulduktan sonra gönderilir ve web tarayıcı bu bilgileri alarak sunucuda çalıştırılan bir programa iletir. Program bilgileri değerlendirdikten sonra gerekli işlemleri yapar.
- Formları bilgi girişinin haricinde bilgi gösterme maksadı ile de kullanabiliriz.

<FORM> Etiketi

- Bir form oluşturmak için <form>...</form> etiketleri arasına istenilen kontroller input etiketi sayesinde belirtilir. Kontrollerden toplanan bilgilerin nasıl değerlendirileceği form etiketinin içinde gösterilir.
- Genel kullanım;
- <FORM ACTION=url METHOD=get- post TARGET=pencere> </FORM>
- Özellikleri:

ÖZELLİK	AÇIKLAMA
ACTION	Formun onay buttonuna basıldığında form bilgilerinin gönderileceği URL yi belirtir.
METHOD	Formun bilgilerinin hangi metotla Server'a gönderileceğini tutar. Get ya da Post olabilir. Varsayılan Get tir. Çok fazla veri gönderileceği zaman çoğunlukla Post metodunu kullanırız. Get metodunda form ile gönderilen veriler URL nin sonuna eklenir.
TARGET	Form değerlendirici tarafından kullanıcıya geri gönderilen yanıtın hangi pencerede belireceğini gösterir.

- Kontrol Tipleri
- HTML nin form kontrolleri <INPUT>, <TEXTAREA> ve <SELECT> etiketleri ile oluşturulur.
- KONTROL YAZIM ŞEKLİ
- TextBox <INPUT TYPE="text">
- Kullanıcının tek satırdan oluşan string değer girmesinde kullanılır.



Adı: <input type="text" name="firstname">
Soyadı: <input type="text" name="lastname">

- Password <INPUT TYPE="password">
- Yukarıdaki gibi string ifade girişi yapılır. Bununla birlikte yazılan karakterler yerine "*" karakteri gösterilir.

Kullanıcı: Emel KAÇAR	Kullanıcı:
Parola:	<input name="username" type="text"/> Parola: <input name="password" type="password"/>

- Hidden <INPUT TYPE="hidden">
- Formla birlikte taşınan bu alandaki bilgi kullanıcıdan gizlenir.
- File <INPUT TYPE="file">
- Forma eklenen kullanıcının bir dosyasının ismini girmesine izin verir.

- CheckBox <INPUT TYPE="checkbox">
- Kullanıcının doğru/yanlış (true/false) şeklinde mantıksal değer girmesinde kullanılır.

☞ HTML □ PHP □ MySQL <HTML><body >
<FORM ACTION=' http://internet/form' METHOD=POST>
<input type="checkbox" name="kutu1" checked="on">HTML

<input type="checkbox" name="kutu2">PHP

<input type="checkbox" name="kutu3">MySQL
</FORM></body></HTML>

RadioButton <INPUT TYPE="radio">

 Özel bir seçenek listesinden saçim yapma imkanı sağlar. Radibuttonların name özelliğine aynı değerler verilerek sınıflandırılabilir.

O HTML O PHP O MySQL

<HTML><body >
<FORM ACTION=' http://internet/form' METHOD=POST>
<input type="radio" name="kutu" value="1">HTML

<input type="radio" name="kutu" value="2">PHP

<input type="radio" name="kutu" value="3">MySQL</FORM>
</body></HTML>

- Submit <INPUT TYPE="submit">
- <FORM> etiketinin ACTION özelliğinde belirtilen URL ye form bilgilerini göndermede kullanılır.
- Reset <INPUT TYPE="reset">
- Bütün form kontrollerini sıfırlamada kullanılır. Formun sayfa ilk yüklendiğindeki eski halini almasını sağlar.
- Image <INPUT TYPE="image">
- Tıklandığında Mouse imlecinin X ve Y koordinatlarını göndererek grafiksel onay buttonu yapmada kullanılır.

- Button <INPUT TYPE="button">
- İstemc-tarafı (Client-Side) scriptlerinden JavaScript ya da VBScript i tetikleyebilecek komut düğmesi oluşturur.

TextArea <TEXTAREA>

- Çok satırlı metinlerin girilmesine imkan sağlayan bir kontroldür.
- ComboBox <SELECT size="1">
- Kullanıcıya açılan bir listeden hazır seçenekler sunar.
- ListBox <SELECT size="n">
- Kullanıcıya liste şeklinde hazır seçenekler sunar. Buradaki "n" ListBox ın kaç satırlı olacağını belirtir.

<INPUT> Etiketi

 Bu HTML etiketi ile birden fazla kontrol çeşidi oluşturuyoruz. Her bir çeşidin hangi özelliklere sahip olduğu be bu özelliklerinin anlamları:

ÖZELLİK	KULLANAN KONTROL	AÇIKLAMA
TYPE	hepsi	Kontrolün çeşidini belirtir. Varsayılan Text tir.
NAME	Hepsi	Kontrolün adını belirtir. Kontrolün adı formun ACTION özelliğinde belirtilen URL ye kontrolde yazılı olan değeri taşır.
VALUE	Text, password, hidden, radio, submit, reset, button	Text, password ve hidden kontrolleri için birinci değeri, radio ve checkbox kontrolleri için seçildiğinde gidecek olan değeri, submit, reset ve button kontrollerinde ise kontrolün başlığını (caption) belirtir.
SIZE	Text, password, file	Kontrolün gösterebileceği karakter uzunluğu
MAXLENGTH	Text, password, file	Kullanıcının gireceği maksimum karakter sayısı
CHECKED	Checkbox, radio	Eğer yazılırsa başlangıçta kontrolün seçili olmasını sağlar.
SRC	image	Resmin kaynağını gösterir.
ALIGN	image	Resmin hizalanmasını belirtir. (left, right, top, middle ya da bottom olabilir.)

- <TEXTAREA> Etiketi
- Çok satırlı metin kutusu oluşturmada kullanılır.
- Kullanımı:
- <TEXTAREA name="adi" rows="m" cols="n">Sayfa yüklendiğinde kontrolde gözükecek olan metin
- </TEXTAREA>
- Özellikleri:

ÖZELLİK	AÇIKLAMA
NAME	Yukarıda yazmıştık
ROWS	Ekranda gösterilecek satır sayısı
COLS	Ekranda gösterilecek sütun sayısı (karakter sayısı olarak)

- <SELECT> ve <OPTION> Etiketleri
- Kontrolü oluşturmak için <SELECT> ve </SELECT> etiketi kullanılır. Bu
- kontrollerde gözükecek her bir seçenek <OPTION> etiketi ile oluşturulur.
- Kullanımı:
- SELECT name="adı" size="n">
- <OPTION value="değer1"> Gözüken Metin1</OPTION>
- <OPTION value="değer2"> Gözüken Metin2</OPTION>
- </SELECT>
- Eğer size>1 ise kontrol ListBox stilinde gözükür. Size=1 ise (varsayılan
- budur) ComboBox (Drop-Down Menu) stilinde gözükecektir.

Farklı Özellikler

- KAYAN, yuvarlanan, ters parende atarak ekrandan kaybolan yazılar insanların ilgisini daha çok çeker. Bu işlemler için Microsoft, <MARQUEE> tag'ını web tarayıcısının yorumlayıcısına eklemiş.
- Netscape ise Explorer 2.0'dan beri var olan bu tag'ı web tarayıcısına koymayı düşünmüyor. Yine de tag'ı kullandığınızda içerisinde var olan yazıların Netscape kullanıcıları tarafından 'normal yazı' olarak görüneceğini belirtelim.

<MARQUEE BEHAVIOR=tip BGCOLOR=renk DIRECTION=yön HEIGHT=n HSPACE=n LOOP=n SCROLLAMOUNT=n SCROLLDELAY=n VSPACE=n WIDTH=n>

- ARQUEE
- id= Eşsiz kimliğini belirtir.
- class= Sınıf adını belirtir.
- style= CSS kuralları tanımlar.
- title= Kısa bilgi belirtir.
- lang= İçerdiği metnin lisanını belirtir.
- dir= İçerdiği metnin yönünü belirtir.
- accesskey= Klavye ile erişim sağlanacak harfi belirtir.
- tabindex= Sekme (tab) sırasını belirtir.
- width= Genişliği belirtir.
- height= Yüksekliği belirtir.
- behavior= Metnin ekranda nasıl hareket edeceğini belirtir.
- direction= Yazının kayacağı yönü belirtir.
- loop= Kaydırma tur miktarını belirtir.
- scrollamount= Kayan yazının her hareketinde önceki yerleşimi ile yeni yerleşimi arasındaki mesafeyi belirtir.
- scrolldelay= Kayan yazının her hareketinde önceki yerleşimi ile yeni yerleşimi arasında geçecek zamanı belirtir. >

kayanYaziKaydırılacak metni belirler.

</MARQUEE>

- <MARQUEE BEHAVIOR=tip BGCOLOR=renk DIRECTION=yön HEIGHT=n HSPACE=n LOOP=n SCROLLAMOUNT=n SCROLLDELAY=n VSPACE=n WIDTH=n>
- BEHAVIOR=tip: Metnin ekranda nasıl hareket edeceğini belirten bu ek SCROLL, SLIDE, ALTERNATE değerlerinden birini alabilir.
- Ek kullanılmadığında SCROLL değeri varsayılan olarak kabul edilir ve yazı, sınırın bir köşesinden belirerek diğer köşeden kayarak kaybolur.
- SLIDE eki ise yazının bir köşeden başlayarak diğer köşede donmasını sağlar.
- Metnin iki köşe arasında gidip gelmesi ise ALTERNATE ekiyle gerçekleştirilebilir.

- BGCOLOR=renk: Kayan yazının arkasındaki rengi belirler. ' renk' değeri 16'lık sayı düzeninde bir değer olabileceği gibi ' red' ,' blue' gibi renk isimlerinden biri de olabilir.
- DIRECTION=yön: Yazının kayma yönünü belirler. DIRECTION eki, LEFT, up, down, RIGHT değerlerinden birini almalıdır. LEFT değeri yazının sola doğru kaymasını, RIGHT ise bunun tersini sağlar.
- HEIGHT=n: Kayan yazının sınırlarından ' yüksekliği' sağlar. N değeri sabit bir piksel sayısı olduğu gibi aynı zamanda bir pencere oranı olabilir. Oran değeri % işareti ile belirtilmelidir.
- HSPACE=n: Kayan yazının sınırlara ne oranda yaklaşacağını belirler. Bu ek özellikle arka plana bir renk verildiğinde daha da belirginleşir.

- LOOP=n: Yazıya verilen hareketin sınırlar içerisinde kaç kere tekrarlanacağını belirler. N değerine ' -1' veya ' INFINITE' verildiğinde hareket sınırsız olarak tekrarlanır. Tip olarak ' SLIDE' kullanıldığında bu işlem tek bir kez gerçekleşeceğinden LOOP ekini kullanmaya gerek yoktur.
- SCROLLAMOUNT, Kayan yazının bir defada kaç piksel hareket edeceğini, SCROLLDELAY ise yazının her bir hareketi arasında kaç milisaniye bekleyeceğini belirler.
- Yazının üst ve alt sınırdan ne kadar uzakta kalacağını VSPACE ile, yazının kayacağı toplam uzunluğu ise WIDTH ile ayarlayabilirsiniz. WIDTH değeri HEIGHT gibi bir yüzde alabilir.

• Örnek Kullanım:

<MARQUEE DIRECTION=RIGHT BEHAVIOR=SCROLL SCROLLAMOUNT=10 SCROLLDELAY=200>Bu kayan bir yazıdır... :) </MARQUEE>

- **<marquee>TEXT</marquee>**---Sola doğru kaydırma (direction= "left")
- <marquee direction="right">TEXT</marquee> --Sağa doğru kaydırma
- <marquee behavior="alternate">TEXT</marquee>--sağ-sol kaydırma
- **e** <marquee direction="up">TEXT</marquee> --yukarı kaydırma
- <marquee direction="down">TEXT</marquee>--aşağı kaydırma
- <marquee direction="up" behavior="alternate">TEXT</marquee>
- ----yukarı aşağı salınım
- <marquee behavior="alternate" direction="up" width="80%"><marquee direction="right">TEXT</marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></m
- Zig-zag salınım
- <marquee behavior="alternate" width="10%"> >> </marquee>TEXT<marquee</p>
 behavior="alternate" width="10%"> << </marquee>
- <marquee behavior="alternate" width="10%">>></marquee>TEXT<marquee</p>
 behavior="alternate" width="10%"><<</marquee>
- <marquee direction="left" style="background:BLUE">TEXT</marquee>
- <marquee style="border:BLUE 2px SOLID">TEXT</marquee>
- <MARQUEE behavior=alternate direction=left scrollAmount=3 width="4%">3</MARQUEE><MARQUEE scrollAmount=1 direction=left width="2%">| | |</MARQUEE>TEXT<MARQUEE scrollAmount=1 direction=right width="2%">| | |</MARQUEE>TEXT<MARQUEE scrollAmount=1 direction=right scrollAmount=3 width="4%">4</MARQUEE>/

- <MARQUEE behavior=alternate direction=left scrollAmount=3 width="4%">4</MARQUEE><MARQUEE scrollAmount=1 direction=left width="2%">| | |</MARQUEE>TEXT<MARQUEE scrollAmount=1 direction=right width="2%">| | |</MARQUEE scrollAmount=1 direction=right direction=right scrollAmount=3 width="4%">3</MARQUEE>
- <marquee behavior="alternate"><marquee width="150">TEXT</marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marquee></marque
- <marquee behavior=alternate direction=up scrollamount=2 scrolldelay=65 height=80 style="Text-align;filter:wave(add=0,phase=1, freq=1,strength=15,color=.FFFFF)"><center>TEXT
TEXT</center></marquee>

- HTML & JavaScript Olayları
- <MARQUEE</p>
- onclick= Elemente klavye veya fare ile tıklandığında tetiklenir.
- ondblclick= Elemente fare ile çift tıklandığında tetiklenir.
- onmousedown= Element üzerinde fare tuşuna basıldığında tetiklenir.
- onmouseup= Basılı olan fare tuşu bırakıldığında tetiklenir.
- onmouseover= Elementin kapladığı alana girildiğinde tetiklenir.
- onmousemove = Element üzerinde fare işaretçisi hareket ettirildiğinde tetiklenir.
- onmouseout = Elementin kapladığı alandan çıkıldığında tetiklenir.
- onkeypress= Element üzerindeyken klavyeden bir tuşa basıldığında tetiklenir.
- onkeydown= Element üzerindeyken klavyeden bir tuşa ilk basıldığında tetiklenir.
- onkeyup= Basılı olan klavye tuşu bırakıldığında tetiklenir.
- onfocus= Elemente odaklanıldığında tetiklenir.
- onblur= Element odaklanmasını yitirdiğinde tetiklenir. >
- </MARQUEE>

Örnek

- Aşağıdaki örnek, HTML MARQUEE etiketi kullanılarak bir metnin aşağıdan yukarı doğru kaydırılmasını ve iki farklı JavaScript fonksiyonu ile kaydırmanın durdurulup, devam ettirilmesini göstermektedir.
- <script type="text/javascript"> function kayDurdur()
- { document.getElementById("ornekMarq").stop(); }
- function kayDevam()
- { document.getElementById("ornekMarq").start(); }
- </script>
- <button type="button" onclick="kayDurdur()">Durdur</button>
- <button type="button" onclick="kayDevam()">Devam</button>
- <marquee id="ornekMarq" direction="up" width="200" height="200"> gmail.com
- </marquee>

MULTIMEDIA ÖĞELERI

- Seçiminize göre geri planda WAV, .AU veya MIDI ses dosyalarından birini çalmak için Internet Explorer'da BGSOUND tag'ını kullanabilirsiniz.
- Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Solution <->

 Soluti
- Deneme.Avi adlı bir video görüntüsünü ekrana yerleştirmek için aşağıdaki gibi bir komut kullanmalısınız. Deneme.Gif dosyası görüntü yüklenene kadar sayfada yer kaplayacaktır.
-

- Sayfanın HEAD alanında kullanılan meta etiketleri (tag) tarayıcılara, arama motorlarına, robotlara, link analiz vb. programların hepsine sayfa hakkında ihtiyaç duydukları bilgileri sağlar.
- Bu bilgiler, meta etiketinin name özelliğinin ismini, content özelliğinin değerini oluşturduğu isim-değer çifti olarak sağlanır.
- Tamamının isteğe bağlı olarak belirtildiği bu değerlerin popüler kullanıma sahip olan özellikler sonraki sayfalarda gösterilmektedir.

- author
- Güncel sayfanın yazarının ve/veya iletişim bilgilerinin belirtilmesini sağlar.
- <meta name="author" content="Resul T., support@gmail.com" />
- abstract
- Güncel sayfanın kısa özetinin belirtilmesini sağlar.
- <meta name="abstract" content="HTML meta etiketi tarayıcılara ve arama motorlarına sayfa hakkında bilgiler sağlar." />
- content-language
- Lisan kodları listesindeki değerlerden birini alarak güncel sayfanın içerik lisanını belirtir.
- Aşağıdaki örnek, sayfa lisanının Türkçe olduğunun belirtilmesini gösterir.
- <meta name="content-language" content="tr" />
- Lisan kodları ile birlikte ayrıca, Ülke kodları listesindeki değerlerden biri kullanılarak lisanın ait olduğu ülke de belirtilebilir.
- Aşağıdaki örnek, sayfa lisanının Türkiye Türkçesi olduğunun belirtilmesini gösterir.
- <meta name="content-language" content="tr-TR" />

- content-type
- Güncel sayfanın içerik tipini ve karakter setini belirtir.
- Aşağıdaki örnek, sayfa içerik tipinin *text/html* ve karakter setinin UTF-8 olarak belirtilmesini gösterir.
- <meta name="content-type" content="text/html charset=utf-8" />
- <meta http-equiv="content-type" content="text/html charset=utf-8" />
- copyright
- Güncel sayfanın kısa kopya/telif hakkı bilgisinin belirtilmesini sağlar.
- <meta name="copyright" content="(c) 2009 Resul T. (gmail.com) Tüm Hakları Saklıdır." />
- description
- Güncel sayfanın açıklamasının belirtilmesini sağlar.
- <meta name="description" content="Bu sayfada HTML meta etiketinin içerebileceği değerlerin isimleri açıklanmaktadır." />
- designer
- Güncel sayfanın (sitenin) tasarımcısının belirtilmesini sağlar.
- <meta name="designer" content="Resul T. gmail Sanat Galerisi" />

distribution

- Güncel sayfanın (sitenin) hitap ettiği bölgenin belirtilmesini sağlar. Aşağıdaki değerleri alabilir.
 - **Global** : Tüm internet ağına hitap ettiğini belirtir.
 - Local : Sitenin bulunduğu IP adres bloğunun ait olduğu ülkeye hitap ettiğini belirtir.
 - IU (Internal Use) : Kamuya hitap etmediğini belirtir.
- Aşağıdaki örnek, bu sayfanının sadece Türkiye'ye hitap ettiğinin bildirilmesini göstermektedir.
- <meta name="distribution" content="local" />
- expires
- Güncel sayfanın son geçerli olduğu tarihi GMT formatında belirtir. Sayfayı daimi olarak geçersiz kılmak için content özelliğine '0' değeri atanmalıdır.
- <meta name="expires" content="Thu, 31 Dec 2009 23:59:59 GMT" />
- generator
- Güncel sayfayı oluşturmak için kullanılan programın isminin ve sürümünün belirtilmesini sağlar.
- <meta name="generator" content="Notepad IDE v5.1" />

- keywords
- Güncel sayfanın içeriğini tanımlayacak virgül ile ayrılmış anahtar kelimeleri belirtir.
- <meta name="keywords" content="html, meta, name, keywords" />
- Ianguage
- İngilizce olmayan sayfalarda İngilizce sayfa lisan adının belirtilmesini sağlar.
- <meta name="language" content="Turkish" />
- publisher
- Güncel sayfayı yayınlayan uygulamanın isminin ve sürümünün belirtilmesini sağlar.
- <meta name="publisher" content="VS 2008" />
- reply-to
- Güncel sayfa hakkında e-posta adresi belirtilmesini sağlar.
- <meta name="reply-to" content="support@gmail.com" />
- revisit-after
- Arama motorlarına gün olarak sonraki ziyaretlerini gerçekleştirme zamanının bildirilmesini sağlar.
- e <meta name="revisit-after" content="7" /> <meta name="revisit-after" content="7 days" />

• refresh

- Belirtilen s
 üre sonunda belirtilen bir URL adresinin y
 üklenmesini sağlar. URL adresi belirtilmezse g
 üncel sayfa adresi tekrar y
 üklenir.
- Aşağıdaki örnek, her 10 saniyede bir güncel sayfanın tekrar yüklenmesini gösterir.
- <meta name="refresh" content="10" />
- Aşağıdaki örnek, sayfa yüklendikten 60 saniye sonra http://www.gmail.com adresinin yüklenmesini gösterir.
- <meta name="refresh" content="60;http://www.gmail.com" />

• title

- Arama motorunun dizininde gösterilecek başlığı belirler. title etiketinden bir farkı yoktur.
- <meta name="title" content="HTML Meta Etiket Değerleri" />

• robots

- Arama motorlarına sayfa ve sayfadaki linklerin takibi hakkındaki davranışlarının bildirilmesini sağlar. content özelliği aşağıdaki değerleri alabilir. Virgül ile ayrılarak birkaç değer birden belirtilebilir.
- **index** : Sayfanın dizine eklenmesini bildirir.
- follow : Sayfa içindeki bağlantıların/linklerin izlenmesini bildirir.
- **archive** : Sayfanın ön izleme için ekran görüntüsünün oluşturulmasını bildirir.
- **noindex** : Sayfanın dizine eklenmesini engeller.
- **nofollow** : Sayfa içindeki bağlantıların/linklerin izlenmesini engeller.
- noarchive : Sayfanın ön izleme için ekran görüntüsünün oluşturulmasını engeller.
 Daha önce oluşturulduysa silinmesini sağlar. (Önizlemenin silinmesi arama motorlarına göre farklılık gösterebilir.)

• robots

- all : Tüm olumlu bildirimleri uygular. Arama motorları index, follow ve archive bildirimlerinden farklı bildirimleri de kabul ediyor olabilir. Bu yüzden arama motorunun yardım/destek sayfasına bakınız.
- **none** : Sayfanın göz ardı edilmesini bildirir.
- noodp : Eğer sayfa DMOZ dizininde kayıtlı ise arama sonuçlarında DMOZ dizinindeki açıklamanın yerine sayfanın kendi açıklamasının kullanılmasını sağlar. Bu değer, özellikle MSNBOT (live.com) için geçerlidir.
- Bir arama motoru robotu aksi belirtilmediği sürece bir sayfayı dizine ekleyebilir, sayfadaki linkleri takip edebilir ve sayfanın ön izlemesini oluşturabilir. Bu yüzden robots özelliği ile all, index, follow ve archive değerlerinin kullanımının çok özel bir anlamı yoktur. Bunlar yerine sadece robotu engellemek amacıyla, engelleme amacına göre none, noindex, nofollow ve noarchive değerlerinin kullanımı daha doğru olacaktır.

- Aşağıdaki örnek, index, follow ve archive değerlerinin uygulamasını göstermektedir.
- <meta name="robots" content="all" />
- Aşağıdaki örnek, arama motoruna sayfayı dizinine eklememesini fakat, sayfadaki linkeri takip etmesinin bildirilmesini göstermektedir.
- <meta name="robots" content="noindex, follow" />
- Aşağıdaki örnek, arama motoruna sayfayı dizine eklemesini fakat, sayfadaki linkeri takip etmemesinin ve sayfanın ön izlemesinin oluşturulmamasının bildirilmesini göstermektedir.
- <meta name="robots" content="index, nofollow, noarchive" />

Uygulama

- Bu uygulama çalışmasında 4 çalışma yapılacak
 - Basit bir Web Sayfası yapımı
 - HTML biçimlendirme etiketlerinin kullanımı
 - Hyperlinkler oluşturma, tablolar
 - Web Sayfasına grafik ekleme
- Bu işlemleri yapmak için şimdiye kadar tüm öğrendiklerinizi kullanarak kendinizi tanıtan ve en az 6 sayfadan oluşan bir Web Sitesi hazırlayınız.

CSS (Cascading Style Sheets) Kullanımı

Hayrioğulu Fatih, (2006), "CSS(Stil Sablonları)"

Stil Sayfaları (CSS- Cascading Style Sheets)

- CSS, Web dokümanlarına stil eklemek için kullanılan basit ve güçlü bir dildir.
- Stil sayfaları web sayfası tasarımcılarının, sayfa görünümlerini daha hareketli hale getirmelerini sağlayan .css uzantılı dosyalardır.
- Css kullanırken kullanacağımız etiket <STYLE> dir. CSS'i HTML sayfalarımıza eklemenin 3 yöntemi vardır.
 - Satır içinde (In-line) style özelliği kullanarak
 - Sayfa Başında <style> etiketi kullanarak
 - Harici CSS dosyası kullanarak
 - <link rel=..> özelliği
 - @import özelliği

- Satır içinde (In-line) style özelliği kullanarak
- Tüm CSS komutlarını kodların içine direk uygulamak.
 Önerilen bir kodlama şekli değildir, ancak özel durumlarda kullanılabilir.
- Örnek:
 -
 - Yapılan işlemin normal HTML komutları ile yazılması;
 -

- Satır içinde (In-line) style özelliği kullanarak
- Örnek:
 - <div style="font:12pt Comic Sans MS; color:red";> bu bölüm diğer bölümlerden farklı görünecektir</div>
 - bu bölüm diğer bölümlerden farklı görünecektir
 - <P style="color:#12ddfa;">bu yazı mavidir.</P>
 - bu yazı mavidir.
 - Siyah zemin beyaz link
 - Siyah zemin beyaz link

Sayfa Başında <style> etiketi kullanarak

- <head> kısmında <style> elemanı içinde CSS kodu yazılarak uygulanır.
- Örnek
 - <head>
 - <title>CSS'i Uygulama</title>
 - <style type="text/css">
 - div { color : red; }
 - </style>
 - </head>

Örnek:

- <html> <head>
- <style>
- P { font-size:8pt; font-family: verdana; color: green; }
- H1 {
 font-size:14pt;
 font-family: times;

color: red;

- DIV {
 - font-size:10pt;
 - color: aqua;

- A {
 - font-size:9pt;
 - font-family: tahoma;
 - color: #5010dd;
 - </style> </head>
- <body>
- <h1>Yeni Sayfa başlığımız</h1>
- Bu bölümde kullanılan stil standart stildir.
- Paragraf kullanıyoruz
- Bu bölümde kullanılan stil standart stildir.
- <div> div tagında değişiklik yaptık. Fakat sadece font büyüklüğü ve renk değişti</div>

- <a href=
 "javascript:window.close()">
- Pencereyi kapat
- </body>
- </html>



Stil Sayfaları (CSS- Cascading Style Sheets)

Harici CSS dosyası kullanarak

 Bu metot da CSS kodlarımızı .css uzantılı bir dosyaya kaydedilir ve <head> etiketi içerisinde sayfaya eklenir.

• <link rel=..> özelliği :

• <link rel="stylesheet" type="text/css" href="ornek.css" />

• @import özelliği:

• <style type="text/css"> @import "ornek.css"; </style>

ornek.css dosyası

- p { color: red; }
- a { color: blue; }

- CSS'in yapısı, Seçiciler (Selector) ve Bildirim Bloğu (Declaration Block) olmak üzere iki ana kısımdan oluşmaktadır. Bildirim Bloğu ise Özellik (Property) ve Değer (Value) şeklinde iki parçadan meydana gelir.
- seçici { Özellik:Değer; } h1 { font-family:Arial; }
- p, h1, body...vb bütün HTML etiketleri CSS için birer seçicidirler. Bir başka ifade ile CSS bu seçicilere özellik ve değer atayarak tasarım işini ifa eder.
- Bildirim Bloğu: Bildirim bloğu süslü parantezle "{ }" açılır ve kapanır. Bildirimler arasında " ; " noktalı virgül kullanılır. Özellik ve değerler birbirinden " : " iki nokta üst üste ile ayrılır.
 - h1 {font: medium Arial;} Burada yazı tipinin boyut ve yazı tipinin ismi gösterilir.
 - h1 {font: 12px / 14px Arial;} Burada " / " Seçicinin yazı tipi boyutunu ve satır yüksekliğini gösterir.

- Gruplama: Birden fazla HTML elemanına de atama yapmaya gruplama denir.
 Gruplama yapılan seçicileri ayırmak için ", " virgül kullanılır. Sınırsız sayıda seçici gruplanabilir.
 - p, h3 {font-family: Arial; font-size: 10pt; font-weight: bold;}
- Burada p ve h3 etiketleri yazı tipi Arial yazı boyutu 2 ve kalın olarak belirtildi.
- **Dizgeler(Strings) :** Dizgeler çift ya da tek tırnak içinde yer alır.
 - div: before {content: "Dikkat"; color: red;}
 - Çıktı: Dikkat div içindeki ifadeden önce
- Dizge içinde çift tırnak kullanmak için bölüm(\) işareti ile birlikte kullanmalıdır.
 - div::after {content: "Dikkat \"burada\" köpek var "; color: red;}
 - Çıktı : div içindeki ifadeden sonra Dikkat "burada" köpek var
 - (before, after: etiketin önüne veya arkasına metin, resim (content=url(resim))ekleme için kullanılır.

- Anahtar sözcükler : Birçok CSS özelliği anahtar sözcük değeri alabilir. Anahtar sözcükleri tırnaksız olarak atanır.
 - p { color:red;} /*Doğru*/
 - p { color:"red";} /* Yanlış*/

• **Değerler :** Değerleri dört gruba ayırabiliriz.

- Uzunluk değerleri
- Yüzde değerleri
- Renk değerleri
- URL değerleri.

- Göreceli Uzunluklar : Bu değerler diğer uzunluk değerlerine göre görecelidir. Sayfalarımızın ölçeğinin değiştiği ve çıktısının alındığı durumlarda bu uzunluk değerlerini kullanmak avantaj sağlayacaktır. Her fontun karakterleri farklı boyutlarda olduğundan, her font için bu uzunluk birimleri farklı sonuçlar çıkaracaktır.
 - **em**: Elemanın yazıtipinin yüksekliğidir. font-size değeri 14 px olarak atanmış ise 1em 14 piksel eşit demektir.
 - ex: Elemanın "x" harfinin yüksekliğidir. Atanmış olan yazıtipinin küçük "x"değeri yüksekliğidir.
 - **px**: piksel değeridir.
 - Örn: #deneme {font-size: 22px;}

- Kesin Uzunluklar : Bu uzunluklar gerçek hayatta kullanılan birimlerdir.
 - in: inç. 1 inch=2,54 cm'dir. Yaklaşık olarak 1in=96 px dir.
 - Örn: line-height: 0.5in
 - cm: Santimetre. Yaklaşık olarak 1cm=38 px dir.
 - Örnek: margin:2cm
 - mm: Milimetre. Yaklaşık olarak 10mm=38px dir.
 - Örnek: letter-pacing:1mm
 - pt: Point. Standart baskı birimidir.100pt, 133pixele karşılık gelmektedir. (1pt = 1/72in)
 - Örnek: font-size:14pt
 - **pc**: Pika. Baskı ölçü birimidir.1pc, 16px'e karşılık gelmektedir.
 - Örnek: font-size:2pc

- Yüzde Değerleri : Yüzde değerleri göreceli değerlerdir. Yüzde değerleri uygulandığı nesnenin boyutuna bağlı hareket eder.
 - h4 { line-height: 150% }
 - /*elemanın satır yüksekliği değerinin yüzde 150 olduğunu gösterir.*/
 - p { text-indent: 10% }
 - /* Satır girintisinin yüzde 10 un içeriden başlayacağını gösterir. */

- Renk Değerleri : CSS'de renk değerleri isim olarak ve RGB degeri olarak tanımlanabilir. RGB tanımlaması da dört çeşittir.
- **Renk İsimleri :** Renklerin İngilizce isimleri kullanılarak tanımlama yapılabilir.
 - p {color: black} /* renk tanımı siyah olarak yapılıyor*/
 - h1{ background-color: red;} /* h1 başlığının zemin rengi kırmızı tanımlanıyor */
- RGB Renk Değerleri:CSS'de renk tanımlamak için genelde kullanılan yöntem budur. RGB tanımlamanın dört çeşidi vardır.
 - #RRGGBB : Burada, RR, GG, ve BB, sırasıyla kırmızı (red), yesil (green) ve mavi (blue) renklerin toplam renk içindeki yoğunluğunu gösteren hekzadesimal sayılardır.
 - Örnek: color:#ff0000; (kırmızı)

• RGB Renk Değerleri

• #RRGGBB

- #RGB: Kısa renk tanımı. Aynı değerleri taşıyan grup değerleri birleştirilir. Örnek: color: f00; (kırmızı)
- **RGB(R,G,B):** Bu RGB renklerinin ondalık sayma düzeninde 0 ile 255 arasında bir tanımı vardır. Buna göre tanımlama yapılabilir.
- Ornek: h1{color: rgb(191,127,127);}
- RGB(R%,G%,B%): Bu RGB renklerinin yüzdesel olarak %0 ile %100 arasında bir tanımı vardır. Buna göre tanımlama yapılır.
- Ornek: h1 {color: rgb(75%,0%,0%);}

•

URL Değerleri

- Herhangi bir metnin, belirli bir alanın, arka planına veya etrafına yerleştirilen resmin gösteriliş şeklidir.
- p { font-size: 12px;
- color: rgb(100%, 100%, 100%)
- background: url("/images/deneme.gif")

• }

CSS Stillerde Öncelik Hakkı

- Bazı durumlarda bir HTML öğesine uygulanmak üzere atanmış birden fazla stil söz konusu olabilir. Bu noktada Css'in ilk C'si yani Cascading kavramı devreye girer. Basamaklı anlamına gelen Cascading, CSS'in stiller arasındaki katmanlı yapısını temsil eder.
- Basamaklı yapıda öncelik hakkı hangi stilde olacaktır? İlk öncelik hakkı içinde **!important** özelliği barındıran stildir. !important özelliği barındıran bir stil yoksa en altta yer alan stil üsttekilere göre öncelikli olacaktır.

CSS Stillerde Öncelik Hakkı

- <style type="text/CSS">
- h1 {color: red !important;}
- h1{color:blue;}
- </style>

Yukarıdaki örneği denediğinizde metin renginin kırmızı olduğunu göreceksiniz. Eğer ilk stil tanımlaması içinden **!important** özelliğini silerseniz, bu sefer de metin rengi mavi olacaktır.

- <body>
- <h1> Stillerde Öncelik Hakkı</h1>
- </body>

Seçiciler (Selectors)

- Seçiciler bize oluşturduğumuz <h1>, <h2> ... gibi etiketlerin mevcut özelliklerini aynı bırakarak onlara yeni özellikler ekleme olanağı verdiği gibi istediğimiz bir kelimeye style özellikleri atayıp istediğimiz zaman çağırmamızı sağlar.
- Üç temel seçici vardır. Bunlar, etiket seçici, tekil (Id) seçici ve sınıf seçicisi.

Class Selectors (Sınıf Seçiciler)

- Aynı HTML elemanına farklı özellikler atamak için Sınıf Seçicisini kullanırız.
- Örneğin iki paragrafımızın olmasının istiyoruz ve bu paragraflardan biri sağa dayalı biri ise orta da olacak şekilde düzenliyoruz Böylelikle genel bazı özellikleri koruyarak farklı özellikleri değiştirebilirsiniz.
- p.sagadaya {text-align: right;}
- p.ortala {text-align: center;}

```
<html>
<head>
<title>Css</title>
<style type="text/css">
<!--
h2.yesil {color:green}
h2.gri {color:gray}
-->
</style>
</head>
<body>
<h2 class=yesil>YE$IL smf secicisi ile </h2>
<br>
<h2 class=gri>GRI smf secicisi ile </h2>
</body>
</html>
```

Sınıf Seçiciler

Çoklu sınıflar, birden fazla sınıfı bir HTML elemanına uygulamak için kullanılır.

- style>
- .stil1{
 - font-size:8pt;
 - font-family: verdana;
 - color: green;
 - }
- .stil2.uyari {
 - font-size:14pt;
 - color: blue;
 - }

</style>

- Yazılacak yazı
- <div class="stil2 uyari"> Yazılacak yazı
 </div>
- A class="stil1" href="#">Link
Tekil (Id) Seçiciler

- Tekil Seçicisi Id niteliğine eşit olan nesneleri seçmek için kullanılır.
- Tekil seçiciler tanımlanırken adının basına # (diyez) işareti konulur.
- HTML belgesinde kendi tanımlayıcı adlarına gönderme yaparak herhangi bir HTML etiketine stil vermekte kullanılır.

```
<html><head><title>Id seçicileri</title>
<METAcontent=text/html;CHARSET=iso-8859-9 http-
equiv=Content-Type>
<style type="text/css">
<!--
#idSecici
{
background:teal;
color:white;
font-weight:bold;
font-family:arial;
}
-->
</style></head><body>
<u id=idSecici>ID Seçiciler</u>
```

```
</body></html>
```

CSS Evrensel Seçiciler (Universal Selector)

- Genel Seçiciler bir elementi bütüne uygulamak için kullanılır bu işlem için
 * işareti kullanılır. Aşağıdaki kod tüm elementleri kırmızı yapacaktır. (Not: Html etiketlerine ait başka bir stil varsa (örneğin renk) o baskın olur, genel seçiciden etkilenmez)
 - <html> <head> <title>Genel Seçiciler</title> <style type="text/css">
 - * { color:red;}

</style>

</head> <body>
 <h1>Dijitalders</h1><h1>DijitalDers</h1>
 <h2>DijitalDers</h1>
 </body></html>

Dijitalders DijitalDers DijitalDers

- Özelliği tüm belgeye değil de bir seçiciye uygulayacağımızda ise kullanımı:
- p *{text-decoration:underline;} :şeklindeki kod, sadece belgedeki p seçicisinin içindeki <u>tüm öğelere</u> uygulanacaktır.

Torun Seçicileri (Descendant Selectors)

- Torun Seçicilere yapılan tanımlama sadece belirtilen elemanlara uygulanır bu kuralların haricinde kalanlara ise uygulanmaz.
 - Örnek; div em {color: gray} Bu kod dokümandaki atası div olan tüm em etiketine sahip torun elemanlarını gri yapacaktır.
 - <div>Yeni Sayfa başlığımız: CSS </div> /*uygulanır*/
 - <div>Yeni Sayfa başlığımız</div>/*uygulanmaz*/
 - <div> Yeni Sayfa başlığımız: CSS </div>/*uygulanır*/
 - Yeni Sayfa başlığımız: CSS
 - Yeni Sayfa başlığımız: CSS
 - Yeni Sayfa başlığımız: CSS
- Burada sınırsız sayıda eleman tanımlayabiliriz.
- Ancak tanımlanan tüm torunlar bulunmalı.
 - Örnek: ul li em{color: gray}



Çocuk Seçicileri(Direct Child Selectors)

 Bazı durumlarda keyfi olarak bir torun seçicisi kullanmaktansa daha ayrıntılı bir ayrım yapmak isteyebiliriz. Örnegin bir div elemanının altındaki em elemanlarından sadece Çocuk Elemanı için tanımlamalar yapmak istersek (torun elemanları hariç), Bunun için çocuk baglayıcısını kullanırız (>) büyüktür işareti.

BODY

em

div

P

em

H1

- <html><head> <style type="text/css">
- div em {color: gray}
- div > em { color: red } </style></head>
- <body><div>Bu koda uygulanacaktır.</div>
- <div>Bu koda uygulanacaktır.</div>
- <div>Bu koddaki bu kısım uygulama dışıdır. </div></body></html>
 Bu koda uygulanacaktır.
- Bu
- *koda* uygulanacaktır.
- Bu
- koddaki bu kısım uygulama dışıdır..

Bitişik Kardeş Seçiciler(Next Sibling Selector)

- Bitişik Kardeş Seçicileri biri biri ardına gelen aynı seviyedeki elemanlara stil tanımlaması için kullanılır. Örneğin bir sıralı gelen h2 elamanından sonraki h3 elemanına stil uygulamak istediğimiz durumlarda kullanılır.
 - <html> <head> <style> h2 + h3 { margin: 2em; color:red } </style> </head><body>
 - <h2>Baslık 2 metni</h2>
 - <h3>Baslık 3 metni</h3>
 - <h2>Baslık 2 metni</h2>
 - <h3>Baslık 3 metni</h3>
 - Buraya önemli ve iyi kod gir
 - <h3>Baslık 3 metni</h3> </body>
 - Baslık 2 metni Baslık 3 *metni*
 - Baslık 2 metni Baslık 3 metni
 - Buraya *önemli* ve **iyi** kod gir
 - Baslık 3 metni



- HTML, dokümanlarındaki seçilen etiketin özniteliklerine veya öznitelik değerlerine göre tanımlama yapmamızı sağlar.
 - Örnek: img [src="grafik.gif"] { border: 1px solid #000} resime kenarlık verilir.
- Basit Öznitelik Seçicisi : Bir etikete ait özniteliğe göre stil ataması yapmamızı sağlar. Örnek: img[alt] {border: 5px solid lime } Burada img etiketinin alt özniteliğindeki elemanın kenarlıkları 5px olarak belirlendi.
- seçici[öznitelik]{Özellik:Değer;}

Örnek <head> <title>CSS'in yapısı</title> <style type="text/css"> img[alt] { border: 5px solid lime } </style> </head> <body> </br> ></br> </body>

- Öznitelik Değeri Seçicisi: Bir etikete ait öznitelik değerine stil ataması yapmamızı sağlar.
 - seçici[öznitelik="öznitelik değeri"] {Özellik:Değer;}
- Değişik öznitelik değerlerine sahip bir uygulamada stil sadece bir öznitelik değerine uygulamak istenebilir. Örneğin input elemanı farklı öznitelik değerine (text, radio) sahiptir. radyo düğmelerine çerçeve değeri uygulamak istenmeyebilir. Bu yüzden önceki tür olan basit öznitelik seçicisi'ni burada kullanılamaz. Örnek:
 - <head> <style type="text/CSS">
 - input[type="text"] {border: 5px solid #000; width: 150px;}
 - </style></head> <body>
 - isminiz :<input type="text" name="ismi" />
 - Uzmanlık Alanınız: <input type="radio" name="uzman" value="Flash" /> Flash <input type="radio" name="uzman" value="Photoshop" /> Photoshop </body>

lsminiz :		
Uzmanlık Alanır	nz: O Flash O Photoshop	

- Kısmi Öznitelik Değeri Seçicisi: Bir etiketin öznitelik değerinin sadece bir parçasına göre stil tanımlaması yapmamızı sağlar.
 - seçici[öznitelik ~="öznitelik değerinin bir bölümü"]{Özellik:Değer;}
 - <head>
 - <style type="text/CSS">
 - h1 [title~="ilk"] {color: blue;}
 - </style>
 - </head>

Örnek uygulanırsa, sadece BAŞLIK 1'in renginin mavi olduğu görülür

- oddy
- <h1 title="ilk başlık">BAŞLIK 1</h1>
- <h1 title="ikinci başlık">BAŞLIK 2</h1>
- </body>
- Örnek: a[title~="Google"] { text-decoration:overline; }

- Ayrımlı Öznitelik Değeri Seçicisi: Bir etiketin öznitelik değerindeki kesik çizgi (-) ile ayrılmış kısımlarına göre tanımlama yapmamızı sağlar.
- seçici[öznitelik | ="öznitelik değerinin bir bölümü"]{Özellik:Değer;}
 - Örnek: *[lang|="en"] { color: red; }
 - <html><head>
 - <style type="text/css"> *[lang|="en"] { color: red; } </style>
 - </head>
 - ody>
 - <h1 lang="en">Hello!</h1>
 - Greetings!
 - <div lang="en-au">G'day!</div>
 - Bonjour!
 - <h4 lang="cy-en">Jrooana!</h4>
 - </body> </html>

Hello! Greetings! G'day! Bonjour! Jrooana!

Sözde(Pseudo) Sınıflar ve Sözde(Pseudo) Elemanlar

- Sözde denmesinin nedeni normal HTML hiyerarsisi içinde bu sınıf elemanların bulunmamasındandır. Bu sınıf ve elemanlar (X)HTML hiyerarşisi ile erişemediğimiz eleman ve sınıflara erişmemizi sağlar.
- Sözde sınıflar; bir elemanı farklı sınıflara böler (örn: link elemanını active, visited gibi sınıflarına böler). Kullanımı: seçici:pseudo-sınıfı {özellik: değer}
 - Örnek: a:visited {color: red}
 - Sözde sınıflarını ikiye ayırabiliriz. Sözde Link Sınıfları ve Sözde Dinamik Sınıfları
- Sözde elemanlar; bir elemanı alt kısımlara böler (örneğin bir paragrafın ilk harfi, bir paragrafın ilk satırı gibi.)
 - Örnek: p:first-line { font-weight: bold; }
- Normal sınıflar sözde sınıflar ve sözde elemanlar ile kullanılabilir.
 - Örnek: a.disariyalinkller:link, a.disariyalinkller:visited {color: maroon}
- Aynı şekilde tekil seçicileri ile birlikte de kullanılabilirler
 - Örnek: a#altkisim:link {font-weight: bold;}

Sözde(Pseudo) Sınıf Seçicileri

Sözde Dinamik Sınıf Seçicileri

- Sözde Dinamik Sınıfları sayfa görünümüne çok büyük katkılar yapabilir. Bu sınıflar genelde linklere uygulanır ancak birçok kullanım alanları vardır.
- :focus Odaklanan elemana stil tanımlaması yapmak için kullanılır Örn: <input> alanına içerik girerken metin giriş alanına odaklandığında renk değişimi yapılması gibi.
- :active Aktif olan elemana stil atamak için kullanılır.
- :hover Bir elemanın üzerine farenin imleci geldiğinde yapılacak tanımlama için kullanılır.
- :lang Lang sözde sınıfı içerikteki bir elemanı farklı bir dilde yazmamızı sağlar.

Sözde Sınıf Seçicileri

• Sözde Link Sınıfı

- Sözde Link sınıf stilleri oluşturulurken iki Sözde Link sınıfı vardır.
- :link Ziyaret edilmemiş linkin alacağı renk ve durum
- :visited Ziyaret edilmiş bir linkin alacağı renk ve durum
- Örnek: a:link {color: blue}
- a:visited {color: red}
- :hover Fare ile linkin üzerine gelindiğinde özelliklerinin işleme konulması. Örneğin *fare linkin üzerine geldiğinde* altı çizili olması isteniyorsa hover kısmında underline seçilmelidir. Üstü çizili olması için overline seçilmelidir. Hem altı hem üstü çizili olması için *iki tane* hover kullanılmalıdır.
- :active- Link tıklandığı anda alacağı renk ve durum
- color : Linklerin, ziyaret edilen linklerin.. renk özelliğinin ayarlanması burada yapılır.
- text-decoration : none seçilirse ilgili linkin altı çizili özelliği ortadan kalkar.
 underline seçilirse ilgili linkin altı çizili özelliği devam eder.

Sözde Link Sınıfı

- <html> <head>
- <style>
- a:link { color : Yellow; text-decoration :none }
- a:active { color : Yellow; text-decoration :none }
- a:visited { color : White; text-decoration :none }
- a:hover { color : Yellow; text-decoration :overline }
- </style>
- <title>Ornek.html</title> </head>
- <body bgcolor="#000080">
- Fırat Üniversitesi
- </body></html>



Sözde Sınıf Seçicileri

Sözde sınıfları birleştirmek

- CSS2.1 ile birlikte aynı seçiciye ait sözde sınıfları birleştirme özelliği de gelmiştir.
- Örneğin ziyaret edilmiş linklerin : hover özelliği ile ziyaret edilmemiş linklerin : hover özelliğini farklı atamak istersek.
- Örnek: a:link:hover {color: red;}
- a:visited:hover {color: pink;}

Sözde(Pseudo) Elemanlar seçicileri

- first-letter (ilk harf): Bir blok-level elemanın ilk harfine stil tanımlaması yapmak için kullanılır.
- first-line (ilk satır): Bir metnin ilk elemanına stil atamak için kullanılır.
- before ve after elemanları: Bir elemanın öncesine ve sonrasına bir içerik veya özellik eklemek için kullanılır.

Sözde Eleman

- <!DOCTYPE HTML>
 <html> <head>
 <style type="text/CSS">
 p:first-child
- {text-decoration:underline ;}
 </style>
 </head>
- ody>

<CSS harika bir teknolojidir.</p> CSS harika bir teknolojidir. Not: first-child sözde sınıfının IE ile çalışması için <!DOCTYPE> belge tanımlanması yapılmış olmalıdır. p:first-child , p etiketinin ilk geçtiği yerde altı çizili olacaktır.

 </body> </html>



CSS Liste (list) Biçimlendirme

- Css'de liste özelliklerinin biçimlendirilmesi amacıyla kullanılır.
- Liste imleme tipi (list style type)
- Burada kullanabileceğimiz unordered list (ul) ve ordered list(ol) olmak üzere iki tip liste seçeneği bulunmaktadır. unordered list (ul) ile kullanacağımız değerler; none, disc, circle ve square olmaktadır.
 - none: İmleme kullanılmayacak anlamına gelir.
 disc: İmleme olarak içi dolu daire kullanmak içindir.
 circle: Bu ise içi boş daire kullanacağımız anlamına gelir.
 square: Kare şeklinde imleme kullanmak için ise bu değeri seçeceğiz.

CSS Liste (list) Biçimlendirme

- ordered list (ol) için ise decimal, decimal-leading-zero, lowerlatin, lower-roman, upper-latin ve upper-roman kullanılmaktadır.
 - decimal: Onluk sayı sistemi kullanmak içindir.
 decimal-leading-zero: Her biri 0 ile başlayan (örn: 01.02.03...vd) onluk sayı sistemi.

lower-latin: Küçük latin harfleri (örn: a,b,c...vd) için. **lower-roman:** Küçük roman rakamları (örn: i, ii, iii, iv... vd) için. **upper-latin:** Büyük latin harfleri (örn: A, B, C, D...vd) için. **upper-roman:** Büyük roman rakamları (örn: I, II, III, IV... vd) için.

CSS Liste (list) Biçimlendirme

<head> <style type="text/CSS"> ul.circle {list-style-type:circle} ul.square {list-style-type:square} ol.upper-roman {list-style-type:upper-roman} ol.lower-latin {list-style-type:lower-latin} </style> </head> <body> Daire: XHTML CSS XML Kare: XHTML CSS XML Roman rakamlar1: XHTML CSS XML Küçük latin harfleri: XHTML CSS XML </01>

</body>

Daire:

o XHTML o CSS o XML

Kare:

- XHTML
- CSS
- XML

Roman rakamları:

I. XHTML II. CSS III. XML

Küçük latin harfleri:

- 1. XHTML
- 2. CSS
- 3. XML

CSS Kenar Çizgisi Stili (border-style)

 Border çizgisini hiç gösterilmeyeceği gibi (none), noktalı (dotted), kesik çizgili (dashed) normal (solid) çit çizgili (double) olarak da gösterebilir.

```
<head>
<style type="text/CSS">
p.normal {border-style:solid;}
p.normal {border-style:dotted;}
p.cizgi {border-style:dotted;}
p.cift {border-style:double;}
</style>
</head>
<body>
Bu cerceve genel cerceve yapısını ifade eder.
Bu cerceve genel cerceve yapısını ifade eder.
Bu cerceve genel cerceve yapısını ifade eder.
Bu cerceve genel cerceve yapısını ifade eder.
Bu cerceve genel cerceve yapısını ifade eder.
Bu cerceve genel cerceve yapısını ifade eder.
Bu cerceve genel cerceve yapısını ifade eder.
Bu cerceve genel cerceve yapısını ifade eder.

Su cerceve genel cerceve yapısını ifade eder.
Bu cerceve genel cerceve yapısını ifade eder.
```

Bu çerçeve noktalarla çizilmiştir. Bu çerçeve kesik çizgilerle çizilmiştir. Bu metnin etrafina çift çizgili border atanmıştır.

CSS Yazıtipi ve Metin ile Kullanımı

Yazıtipi (font) Biçimlendirme

 p { font:bold italic 1.5em Verdana, Helvetica, Arial, sansserif;}

Yazıtipi kalınlığı

font-weight=bold (kalın) | none (normal yazı);

Yazıtipi boyutu

font-size=px|em|pc|vs..;

Yazıtipi stili

font-style=italic|none;

Yazıtipi ailesi

font-family: yazı tipleri (arial, verdana...)

CSS Yazıtipi ve Metin ile Kullanımı

• Metin işaretleme

• text-decoration=underline(altı çizili)|overline (üstü çizili)|none;

Metin büyük-küçük harfe dönüşüm

- text-transform=uppercase|lowercase|none;
- Metin hizalama
 - text-align=left|right|center|justify;
- Metin gölge efekti
 - text-shadow: [yatay(x)koordinati mesafesi] [dikey(y)koordinati mesafesi] [bulanıklık (blur) değeri] [renk]
 - Burada x ve y koordinatlarında artı(+) ya da eksi(-) değer kullanılabilir. Yatayda(x) artı(+) gölgenin metne sağdan olan uzaklığı, eksi(-) soldan olan uzaklığıdır. Dikeyde (y) artı(+) gölgenin metne alttan olan uzaklığı, eksi (-) ise üstten olan uzaklığıdır.
 - p{ font:18px Arial, Helvetica, sans-serif; color: #000000;
 - text-shadow: 4px 5px 1px #C9C3C3;
- CSS harika bir teknolojidir.

- •
- text-shadow: -4px -5px 1px rgba(150, 150, 150, 0.82);

CSS harika bir teknolojidir.

Opaklık veya şeffaflık değeri

CSS Artalan (Background) Biçimlendirme

- background–color: Artalan rengi belirleme
 - p.artalan{background-color: #C0C0C0;}
 - background-color:transparent|renk;
- background image: Artalana resim atama
 - body {background-image:url('artalan.jpg')}
 - body {background-image:url(artalan.jpg)}
 - div {background-image:url("artalan.jpg")}

 background – attachment: Bu özellik; scroll değeriyle kullanıldığında artalanın sayfa ile beraber kaymasını; fixed değeri ile kullanıldığında ise artalanın sabit kalmasını sağlar.

- background-image:url('artalan.jpg');
- background-repeat:no-repeat;
- background-attachment:fixed|scroll;

CSS Artalan (Background) Biçimlendirme

- background repeat: Artalan resminin tekrarı ve tekrar yönü ile ilgili düzenlemeler için kullanılır. Bu özelliği kullanmadığımızda artalan resmi hem yatay hem de dikey olarak tekrarlar ve tüm artalanı kaplar. İşte bu özellikle bu standart kullanımdan farklı biçimlendirmeler yapılabilmektedir.
 - Aldığı değerler
 - repeat : Resim hem yatay hem de dikey olarak yinelenir.
 - repeat-x: Resim yalnız yatay düzlemde yinelenir.
 - repeat-y: Resim yalnız dikey düzlemde yinelenir.
 - no-repeat: Resim yinelenmez.
 - <style type="text/CSS">
 - { background-image:url('artalan.jpg');
 - background-repeat:repeat-x;
 - }
 - </style>

CSS Artalan (Background) Biçimlendirme

- background position: Artalan resminin konumunu belirlemek için kullanılır. Aldığı değerler;
 - Yatay hizalama için: left, center, right Dikey hizalama için: top, center, bottom
 - Bu değerler top left, top center, top right, center left, center center, center right, bottom left, bottom center, bottom right şeklinde birlikte veya piksel olarak, yüzde olarak ekranda konumlama yapılabilir.
 - <style type="text/CSS">
 - body{background-image:url('artalan.jpg');
 - background-repeat:no-repeat;
 - background-attachment:fixed;
 - background-position:center;
 - background-position:50% 50%;}
 - </style>
 - body { background: #00ff00 url('artalan.jpg') no-repeat fixed center; }

Stillerle Fontların Kullanımı

- Örnek:
- STYLE>
- <!---
- .stil{

}

- color: green;
- font-family: verdana;
- font-size:8pt;
- font-style:none;
- font-weight:bold;
- background:yellow;

- .stil1{
 - background:pink;
 - font-family: impact;
 - text-align:center;
 - --->

}

</style>

Stillerle Fontların Kullanımı

• Örnek:

- HTML>
- HEAD>
- <LINK rel=stylesheet href="stil.css" type="text/css">
- </HEAD>
- ody>
- Body tagının normal stili
- İlk paragrafımız . stil1</h1>
- <P class="stil ">ikinci paragraf . stil </P>
- Bu bölümde kullanılan stil standart stildir.
- <div class="stil"> Div tagında stil değişikliği yaptık. stil </div>
- Yine SPAN tagında değişiklik var. stil1
- </BODY> </HTML>



CSS line-height, word-spacing ve letter-spacing

- line-height: Satırlar arası yükseklik.
- <style type="text/CSS">
- p.dar {line-height: 80%}
- p.normal {line-height:normal}
- p.genis {line-height:200%}
- </style>
- word-spacing: Sözcükler arası mesafe
- p {word-spacing:30px;}
- letter-spacing: Harfler arası mesafe
- p{letter-spacing: 10px;}

Stillerle BODY Etiketinin Varsayılan Değerini Değiştirme

- Örnek:
- HTML> <HEAD>
- STYLE>
- <!—
- BODY{
 - background:#000000;
 - color: #ffffff;
 - font-family: verdana;
 - font-size:10pt;
 - font-weight:bold;
 - text-decoration=none;
 - margin-left:30px;
 - margin-right:50px;
 - margin-top:27px;
 - }
- \rightarrow </style> </head>

- Body etiketi için tanımlanan yeni değerler sayfada etkin olarak kullanılacaktır.
- Normal olarak sayfalar sol en üst köşeden itibaren yazmaya başlanır. Sayfaya kenarlıklarına boşluk vermek için margin ifadesi kullanılır.

Stillerle BODY Etiketinin Varsayılan Değerini Değiştirme

- <HTML> <head>
- STYLE>
- <!---
- BODY {background: white;}
- H1 {font: 24pt Arial bold}
- P {background: yellow;}
- A {text-decoration: none; color: blue}
- -->
- </STYLE> <title>Style Sheet Kullanımı</title> </head>
- BODY>
- <H1>Bu bir başlık! 24 punto, Arial ve Kalın.</H1>
- Üzeri sarı işaretleyici ile çizilmiş alelade bir yazı.
- Linklerin altının çizilmemesini bu yolla sağlayabilirsiniz.

🗿 Style Sheet Kullanımı - Microsoft Internet 🔳 🗖 🔀					
<u>D</u> osya	Dü <u>z</u> en	<u>G</u> örünüm	<u>S</u> ik Kullanılanlar	<u>A</u> raçlar <u>Y</u>	» 🥂
Bu bir başlık! 24 punto, Arial ve Kalın.					
Üzeri sarı işaretleyici ile çizilmiş alelade bir yazı.					
Link'lerin altının çizilmemesini bu yolla sağlayabilirsiniz. 🥃					
🙆 🤤 Bilgisayarım					

 etiketinin CSS ile kullanımı

- etiketi, isteğe bağlı verilerin gösterilmesi için yapısal ve görsel bir katman oluşturulmasını sağlar. SPAN, DIV etiketinden farklı olarak bir satır içi (*inline*) element olduğu için metnin içinde kullanılması durumunda ilgili metni kesmez.
- elementi ilk anda herhangi bir görsel özelliğe sahip olmadığı için Style ve/veya CSS özelliklerinin ayrıca belirtilmesi gerekir.
- HTML Sözdizimi ve Özellikler
- SPAN
- id=Eşsiz kimliğini belirtir.
- class=Sınıf adını belirtir.
- style=CSS kuralları tanımlar.
- title=Kısa bilgi belirtir.
- lang=İçerdiği metnin lisanını belirtir.
- dir=İçerdiği metnin yönünü belirtir. (Itr (Left-To-Right) : Soldan sağa. rtl (Right-To-Left)
- align=İçerdiği verinin yatay hizalamasını belirtir. >
- İcerik İsteğe bağlı HTML kodlarını belirler.

 etiketinin CSS ile kullanımı

- HTML & JavaScript Olayları
- <SPAN onclick= Elemente klavye veya fare ile tıklandığında tetiklenir.
- ondblclick= Elemente fare ile çift tıklandığında tetiklenir.
- onmousedown= Element üzerinde fare tuşuna basıldığında tetiklenir.
- onmouseup= Basılı olan fare tuşu bırakıldığında tetiklenir.
- onmouseover= Elementin kapladığı alana girildiğinde tetiklenir.
- onmousemove= Element üzerinde fare işaretçisi hareket ettirildiğinde tetiklenir.
- onmouseout= Elementin kapladığı alandan çıkıldığında tetiklenir.
- onkeypress= Element üzerindeyken klavyeden bir tuşa basıldığında tetiklenir.
- onkeydown= Element üzerindeyken klavyeden bir tuşa ilk basıldığında tetiklenir.
- Onkeyup= Basılı olan klavye tuşu bırakıldığında tetiklenir. >
- Örnek
- Aşağıdaki örnek, HTML SPAN etiketi kullanılarak bir paragraf içindeki bir kelimeye CSS özelliklerinin uygulanmasını göstermektedir.
- myNotlar != myNotlar
- Canlı Sonuç
- myNotlar != myNotlar

CSS Kırpma (clip) & Görünürlük (visibility) Özellikleri

- Bu özellik, Css'de içeriğin sadece bir bölümünün gösterileceği şekilde kırpılması için kullanılır. Kırpma olmayacağı anlamına gelen başlangıç varsayılan değeri ise **auto** olmaktadır.
- <head> <style type="text/css">
- .clip
- { position:absolute;
- clip:rect(50px,210px,190px,70px);
- }
- </style> </head>
- <body>

-
- </body>



CSS ile kutulara gölge efekti (box-shadow) uygulama

- Gölge efekti için CSS3 ile gelen box-shadow özelliğini kullanılır.
 - Genel Kullanımı: box-shadow: [yatay(x) koordinatı mesafesi] [dikey(y)koordinatı mesafesi] [bulanıklık (blur) değeri] [renk]
 - box-shadow: 2px 2px 2px #000;
 - Burada x ve y koordinatlarında artı(+) ya da eksi(-) değer kullanılabilir. Yatayda(x) artı(+) gölgenin metne sağdan olan uzaklığı, eksi(-) soldan olan uzaklığıdır. Dikeyde (y) artı(+) gölgenin metne alttan olan uzaklığı, eksi (-) ise üstten olan uzaklığıdır. Bu özellik Internet Explorer tarafından henüz desteklenmemektedir.
 - .kutu {
 - box-shadow: 3px 3px 4px #000;
 - width:400px;
 - padding:5px;
 - }

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi eleifend rutrum eros, eu aliquet lorem iaculis ac. Donec et metus nunc. Donec non nibh purus, sit amet sagittis augue. Mauris aliquam, sapien accumsan molestie tempor, dui felis feugiat purus, eget consectetur ante nunc at orci. Praesent sem diam, luctus tincidunt mollis at, porta vel augue. Donec volutpat porta turpis, vitae iaculis felis venenatis sit amet. Sed eget libero laoreet nibh facilisis tincidunt.

CSS ile Yuvarlatılmış (oval) Köşeler Oluşturmak

- Bu işlemi CSS3 ile gelen border-radius özelliği ile gerçekleştiriyoruz. Öncelikle bu özelliğin IE tarafından desteklenmediğini belirtelim. (IE 9 sürümünde desteklenecek.)
 - #kutu {
 - width: 400px;
 - border: 2px solid #000;
 - padding: 10px;
 - border-radius: 10px;
 - }

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi eleifend rutrum eros, eu aliquet lorem iaculis ac. Donec et metus nunc. Donec non nibh purus, sit amet sagittis augue. Mauris aliquam, sapien accumsan molestie tempor, dui felis feugiat purus, eget consectetur ante nunc at orci. Praesent sem diam, luctus tincidunt mollis at, porta vel augue. Donec volutpat porta turpis, vitae iaculis felis venenatis sit amet. Sed eget libero laoreet nibh facilisis tincidunt.

CSS ile Yuvarlatılmış (oval) Köşeler Oluşturmak

- Bu özelliği kullanırken tüm köşeleri oval yapmak zorunda değilsiniz. Aşağıdaki örnekte olduğu gibi border-radius özelliğini sadece istediğiniz köşelere uygulayabilirsiniz.
 - #kutu {
 - width: 400px;
 - border: 2px solid #000;
 - padding: 10px;
 - border-bottom-left-radius: 30px;
 - }

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi eleifend rutrum eros, eu aliquet lorem iaculis ac. Donec et metus nunc. Donec non nibh purus, sit amet sagittis augue. Mauris aliquam, sapien accumsan molestie tempor, dui felis feugiat purus, eget consectetur ante nunc at orci. Praesent sem diam, luctus tincidunt mollis at, porta vel augue. Donec volutpat porta turpis, vitae iaculis felis venenatis sit amet. Sed eget libero laoreet nibh facilisis tincidunt.
CSS Konum (position) Özelliği

- Sayfa tasarımı yaparken en çok sıkıntı çekilen konulardan biri bir elamanı sayfanın istenilen noktasına yerleştirememektir.
- CSS sayfaya noktasal olarak ulaşmaya izin verir. Örneğin Sayfanın alt kısımlarında iken ortaya yazı yazdırılabilir.

<P style="font:14pt times; position:absolute;top:150px; left:100px; width;200px">deneme</P>

CSS Konum (position) Özelliği

• position:absolute | relative | fixed;

- absolute: Sayfanın en başından geçerli noktayı alır.
- relative: Sayfanın üsten en son yazılan elamanının bulunduğu noktayı başlangıç olarak alır.
- fixed: Browser penceresine göre konumlama yapar.
- top: belirlenen başlangıç noktasından belirtilen pixel kadar aşağı yazar.
- left: başlangıç noktasında sağa doğru belirtilen pixel kadar uzaklığa yazar.
- width: yazılacak yazı için maksimum genişlik belirler ve bu genişlik aşıldığı zaman bir alt satıra geçer.

CSS z-dizin (z-index) özelliği

- Bir web sayfası hazırlanırken sayfamıza düzen verirken sayfayı yatay ve dikey olarak düşünüyoruz. Bu koordinat sisteminde x ve y olarak ifade edilen iki boyuta denk gelir.
- Üçüncü boyut ise derinliktir. Derinliğe koordinat sisteminde z ismi verilir. Verilen parametrelerden yararlanılarak üçüncü boyutun özellikleri web sayfalarında kullanılabilir. z-index parametresi derinlik olarak yeni stile bir değer verir. İki stil hazırladığımızda z-index değeri büyük olan üstte görünecektir. Aşağıdaki kodlar iki stilin birlikte kullanılmasını gösteriyor.



- HTML> <HEAD>
- STYLE>
- <!- -
- #bolum1{
 position:absolute;
 top:10px;
 left:20px;
 z-index:0;
 }
- #bolum2{

position:absolute; top:10px; left:20px; z-index:1; }

- </STYLE>
- </HEAD>
- BODY>
- SPAN id="bolum1">
- IMG src="cizgi1.gif" alt="cizgi1">
-
- SPAN id="bolum2">
- IMG src="cizgi2.gif" alt="cizgi2">
- </BODY></HTML>







- Şimde de bu örnek üzerinde z-index parametrelerini değiştirerek aynı örneği tekrar görelim
- <HTML> <HEAD> <STYLE>
- <!--

-->

- #bolum1{
 position:absolute;
 top:10px;
 left:20px;
 z-index:1;
 }
 #bolum2{
 position:absolute;
 - position:absolute; top:10px; left:20px; z-index:0; }

- </STYLE>
- </HEAD>
- BODY>
- SPAN id="bolum1">
- IMG src="cizgi1.gif" alt="cizgi1">
-
- SPAN id="bolum2">
- IMG src="cizgi2.gif" alt="cizgi2">
- </BODY></HTML>





- <html> <head> <style>
- div{ position:absolute; width:150px; height:150px;}
- div.bir { background: #FEB3BE;
- border:2px solid #CC8B94; top: 0; left: 0; z-index:2}
- div.iki { background: #E5ECF9;
- border:2px solid #BCCCEB; top: 10px; left: 10px; z-index:1}
- </style> <head>
- <body>
- <div class='bir'></div>
- </div></body>



İç ve dış kenar boşlukları

- Padding: İçeriğin elementin sınırlarına olan uzaklığını belirlemek için kullanılır. Örneğin "padding:4px;" şeklindeki bir deklerasyon içeriğin sınırlara (üst, sağ, alt ve sol) olan uzaklığının 4 pixel olacağını söyler.
 - padding:10px 5px 15px 20px; (üst, sağ, alt ve sol)
 - padding:10px 5px 15px; (üst, sol-sağ, alt)
 - padding:10px 5px; (üst-alt, sol-sağ)
- Margin: Elementin komşu elementlere olan uzaklığını belirtmek için kullanılır. Örneğin "margin:4px;" şeklindeki bir deklerasyon elementin komşularının sınırlarına olan uzaklığının 4 pixel olacağını söyler.

Nesnelerin yerleşimi

display :

- inline : Satır içi kutu .Kullandığımız nesneleri yanyana sıralar.
- block : Blok kutu. Kullandığımız nesneleri alt alta sıralar.
- none : Kullandığımız nesneleri göstermez. Genellikle JavaScript methodlarında kullanılır.
- list-item: Bu değer bir elemanın (HTML'deki gibi) bir blok kutu ve bir liste öğesi satır içi kutusu üretmek için kullanılır.



- **Overflow**: visible | hidden | scroll | auto | inherit
- inherit kalıtsallık üst elementin değerini alabilir
- visible kutu sınırları dışına taşan alanları göster
- hidden kutu sınırları dışına taşan alanları gizle.
- **scroll** kutu sınırlarını taşan kısmı kaydırma çubuğu ile ulaşmamızı sağlar.
- **auto** kutu sınırlarını aşması durumunda scroll ile aynı etkiyi yapar.
- div#solMenu {
- position: absolute;
- top: 0;
- left: 0;
- width: 200px;
- height: 120px;
- background: #BBB;
- overflow: visible;

- 🔹 <u>Anasayfa</u>
- 🔸 <u>Ürünler</u>
- 🔸 <u>Diğerleri</u>
- 🔸 <u>Diğerleri</u>
- 🔸 <u>Diğerleri</u>
- Diğerleri
- <u>Diğerleri</u>

•



• **Overflow**: visible | hidden | scroll | auto | inherit

Anasayfa

Ürünler Diğerleri Diğerleri Diğerleri Diğerleri

- div#solMenu {
- position: absolute;
- top: 0;
- left: 0;
- width: 200px;
- height: 120px;
- background: #BBB;
- overflow: scroll;
- }
- overflow: hidden;







- <HTML> <HEAD> <STYLE><!—
- div.ornek1 {position:absolute;
- top:30px;
 left:60px;
- width:50px; height:170px;
- background-color:#A5760A;
- overflow : hidden;z-index:1;
- padding:5px; border:1px solid black;}
- div.ornek2 {position:absolute;
- top:60px;
 left:30px;
- width:200px; height:50px;
- background-color:#A5760A;
- overflow : hidden;z-index:2;
- padding:5px; border:1px solid black;}

div.ornek3 {position:absolute; top:40px; left:150px; width:50px; height:160px; background-color:#FFD700; overflow : hidden; z-index:3; padding:5px;border:1px solid black;} div.ornek4{position:absolute;

> top:130px; left:130px; width:150px; height:50px; background-color: yellow; overflow : hidden;z-index:4;padding:5px; border:1px solid black;}

--->

</STYLE> </HEAD> <body>

<div class="ornek1">1</div> <div class="ornek2">2</div> <div class="ornek3">3</div> <div class="ornek4">4</div> </BODY> </HTML> CSS İle Menü Oluşturma ÖRNEK

Örnek 1: <html> <head>

- <style type="text/css">
- #coolmenu {

• }

- border: 1px solid black;
- border-bottom-width: 0;
- width: 170px; /*sütün genişliği*/
- background-color: #E6E6E6;

CSS Drive JavaScript Kit Free JavaScripts JavaScript Reference Coding Forums



- font: bold 13px Verdana;
- padding: 2px; padding-left: 4px;
- display: block; width: 100%; color: black;
- text-decoration: none;
- border-bottom: 1px solid black; }
- #coolmenu a:hover{
- background-color: black;
- color: white;
- } </style> </head>

CSS Drive
JavaScript Kit
Free JavaScripts
JavaScript Reference
Coding Forums

- <body>
- <div id="coolmenu">

- CSS Drive JavaScript Kit Free JavaScripts JavaScript Reference Coding Forums
- CSS Drive
- JavaScript Kit
- Free JavaScripts
- JavaScript
 Reference
- Coding Forums
- </div>
- </body> </html>

• Örnek 2:

<html> <head> <style type="text/css">

- #button {
- width: 150px;
- border-right: 1px solid #000;
- padding: 0 0 1em 0;
- margin-bottom: 1em;
- font-family: 'Trebuchet MS', 'Lucida Grande', Arial, sans-serif;
- font-size: 90%;
- background-color: #90bade;
- color: #333; }



- #button ul {
- list-style:none ;
- margin: 0;
- padding: 0;
- border: none; }
- #button li {
- border-bottom: 1px solid #90bade;
- margin: 0; }



- #button li a {
- display: block;
- padding: 5px 5px 5px 0.5em;
- border-left: 10px solid #1958b7;
- border-right: 10px solid #508fc4;
- background-color: #2175bc;
- color: #fff;
- text-decoration: none;
- width: 100%;



• }

- #button li a:hover {
- border-left: 10px solid #1c64d1;
- border-right: 10px solid #5ba3e0;
- background-color: #2586d7;
- color: #fff;
- }
- </style> </head>

e-dersler	
C/C++	
Pascal	
Java	
HTML	
×html	
×ml	

- <body> <div id="button">
- •
- e-dersler
- C/C++
- Pascal
- Java
- HTML
- xhtml
- xml </div> </body></html></html>



Örnek 3: <html><head> <style>

- #navcontainer ul{
- margin: 0;
- padding: 0;
- list-style-type: none;

• }

- •
- #navcontainer li {
- margin: 0 0 .2em 0;



- #navcontainer a {
 - display: block;
 - color: #ffffff;
 - background-color: #003366;
 - width: 9em;
 - padding: .2em .8em;
 - text-decoration: none;}
- #navcontainer a:hover{
 - background-color: rgb(51,102,153);
 - color: #ffffff; }



• #active a {

- display: block;
- color: #FFF;
- background-color: #600;
- width: 9em;
- padding: .2em .8em;
- text-decoration: none;
- }

</style>

</head>

CSS CODE HTML CO	DDE
Ankara	
İstanbul	
İzmir	
Adana	
Erzurum	

- <body> CSS CODE
- HTML CODE
- <div id="navcontainer">
- Ankara
- İstanbul
- İzmir
- Adana
- Erzurum
- </div></body></html>



dropdownlist menü yapımı

- html> <head>
- Ink href="style2.css" type="text/css" rel="stylesheet" />
- </head>
- div id="menu">
- >menu item
- menu item 1
- menu item 2
- class="item">menu item 3
 </div> </html>

dropdownlist menü yapımı

- Style sayfası (Bu sayfa Style2 adında kaydedilmelidir)
- #menu ul.item{display:none;}
- #menu ul:hover.item{display:block;}
- #menu {position:absolute;}
- Fare ile üzerine gelindiğinde menü açılır.



Sütun yapımı

- <html> <head> <style>
- .sollink {margin-right: 20px;font-size: 15px;}
- </style> <head>
- <body> <div>
- Link1
- Link2
- link3
- link3
- link4
- link5
- link6
- </div> </body> <html>



Resimli buton yapımı

- .outer a, .outer div, .outer li, .outer ul, .outer body { margin: 0; padding: 0;
- font-family: Verdana, Helvetica, Arial, sans-serif; }
- div.menu {width: 193px;}
- div.menu ul {list-style: none;text-indent: 0px;}
- div.menu li {margin-top: 3px;}
- div.menu a {font: 12px Verdana; padding: 4px; color: #646464;
- display:block; padding: 6px 30px; height: 20px;
- text-decoration: none;
- background: url('images/background.gif');}
- div.menu a:hover{background: url('images/highlight.gif') no-repeat right;
- padding: 6px 40px;}

	Firefox T	x
Recimit huton vanimi	CSS Örnekleri.com	+
Resinin buton yapını	< □ · ☆ マ С ▮ - Аи Я	3
	Ana Sayfa	
	Hakkımızda	
	Hizmetler	
	İletişim	
• <html> <head></head></html>	http://wwleri.com/	

- <link rel="stylesheet" type="text/css" href="style.css"> </head>
- ody>
- <div class="outer">
- <div class="menu">
- Ana Sayfa
- Hakkımızda
- Hizmetler
- İletişim
- •
- </div> </div> </body> </html>

Örnek

<pre>a.menu { font-family:Verdana; font-weight:bold; font-size:13px; color:#993333; text-decoration:none;</pre>	Burada "menü" adında link özellikleri ayarladık. Yazı tipi - verdana, kalın, 13px, bir renk, menünün altındaki çizginin çıkmaması için "text-deceration" ifadesini "none" yaptık, 3px boşluk ve 3px kenarlık ekledik.	
padding:3px; border:solid 3px } a.menu:hover { background:#000000;	Burada linkin "hover" özelliğinin değerlerini atatık Bilindiği gibi "hover" üzerine gelince gözüken linkti. Bir arkaplan, yazı renki, yazı kalınlığı ve 3px lik boşluk kullandık.	
font-weight:bold; padding:3px }	Burada menünün "active" özelliğin linke basıldığında karşımıza çıkan görüntüyü ifade etmektedir. Bir renk, ve altında çizgi çıkmamasını sağladık.	
a.menu:active { color:#00FFFF; text-decoration:non a.menu:visited { background-color:# color:#666666;	Burada menünün "visited" yani linke basıldıktan sonra karşımıza çıkan görüntüyü ifade etmektedir. Bir arkaplan, bir renk, ve altında çizgi	

- html>
- < head>
- <title>Css'in HTML içerisinde kullanım şekilleri</title>
- <link href="stil1.css" rel="stylesheet" type="text/css">
- </head> <body>
- Ana Sayfa |
- Hakkımızda |
- Referanslar |
- İletişim
- </body> </html>



Stillerle Yazıtipi(Font) Özelliklerinin Kullanımı

Komut	Değer	Açıklama
font-family	Tahoma, Comic Sans MS vs.	Yazı tipini belirler.(Arial , Verdana gibi.)
font-style	normal italic oblique inherit (kalıtsallık)	Yazının normal veya sağa eğik olmasını sağlar.
font-variant	normal small-caps inherit	Yazının, yazıldığı gibi veya tümünün büyük harflerden oluşmuş gibi görünmesini sağlar. İkinci durumda metnin ilk harfi de boyut olarak büyük gözükür.
font-weight	normal bold bolder lighter 100900	Yazının kalınlık-inceliğini belirler.
font-size	Kesin değerler(absolute-size) (xx-small x-small small medium large x-large xx-large) Göreceli değerler (relative-size) (larger smaller) px pt %	Yazının büyüklüğünü belirler.

Stillerle Metin(Text) Özelliklerinin Kullanımı

Komut	Değer	Açıklama
text-align	left right center justify inherit	Metin hizalama
vertical-align	baseline: Orta (metin içeriğine göre) sub: Altsimge super: Üstsimge top: Yukarı(elemana göre) text-top: Yukarı (Metin içeriğine göre) middle: Orta (Elemana göre) bottom: Alt (Elemana göre) text-bottom: Alt(Metin içeriğine göre)	Dikey hizalama
text-indent	Boyut Yüzde inherit	Paragrafların ilk cümlelerin soldan içeriye kaydırılmasını sağlar.
line-height	normal <sayı> <boyut> <yüzde> inherit</yüzde></boyut></sayı>	Satır yüksekliği
word-spacing	normal <boyut:px vb.=""> inherit</boyut:px>	Sözcükler arasındaki boşluk değerini belirler. Eksi değer alabilir.
letter-spacing	normal <boyut:px vb.=""> inherit</boyut:px>	Harfler arasındaki boşluk değerini belirler. Eksi değer alabilir.
text-transform	none capitalize(İlk harfbüyük) uppercase lowercase inherit	Metnin büyük-küçük harf çevirmek için kullanılır.
text-decoration	none underline overline line- through blink inherit	Satırın altını, üstünü, ortasını çizme ve yanıp sönme

Stillerle Metin(Text) Özelliklerinin Kullanımı

Komut	Değer	Açıklama
white-space	normal pre nowrap inherit normal: Birden fazla boşluğu tek boşluk gibi sayar. pre: Birden fazla boşluğu birlestirmez. nowrap : etiketi hariç alt satıra geçişe izin vermez.	Boşlukların korunması
direction	ltr rtl inherit	Metin yazma yönü. direction özelliğinin satıriçi seviyeden elemanlar üzerinde etkili olabilmesi için unicode-bidi özelliğinin değerinin embed veya override olması gerekir.
unicode-bidi	bidi-override embed normal inherit	Çok yönlülük algoritması . Bazı diller sağdan sola yazılırken bazı diller solda sağa doğru yazılır. Bazı dokümanlarımızda her ikisini birden kullanabiliriz buna çok yönlülük denir
Stillerle Artalan (Background) Özelliklerinin Kullanımı

Komut	Değer	Açıklama
color	Renk adı veya RGB değeri (red veya #FF0000 gibi), inherit	Renk tanımlaması
background	background-color> <background-image> <background-repeat> <background- attachment> <background-position></background-position></background- </background-repeat></background-image>	Bu özellik Artalan(background) ile ilgili tüm özelliklerin bir arada kullanımı sağlar.
background-color	Renk adı RGB değeri (red veya #FF0000 gibi) transparent inherit	Arka plan rengi
background-image	none url(image) inherit	Arka plana yerleştirilecek resmin adresi
background-repeat	repeat repeat-x repeat-y no-repeat inherit repeat [yinele]: Resim hem yatayda hem de düşeyde yinelenir. repeat-x: Resim sadece yatay olarak yinelenir. repeat-y: Resim sadece düşey olarak yinelenir. no-repeat [yineleme]: Resim yinelenmez; resmin sadece bir kopyası oluşturulur.	Arka plan resminin hangi yönde tekrarlanacağını belirler.
background-attachment	scroll fixed inherit	Sayfa kaydırma çubuğuyla hareket ettirildiğinde zeminin de hareket edip etmeyeceğini belirler.
background-position	yüzde boyut top center bottom left center right inherit	Artalan resmi konumu. Zemindeki resmin sayfa üzerinde istenilen yere yerleştirilmesini sağlar.

Stillerle Listeleme Özelliklerinin Kullanımı

Komut	Değer	Açıklama
list-style-type	disc circle square decimal lower-roman upper- roman lower-alpha upper-alpha none inherit	Listenin harflerden mi yoksa sayılardan mı oluşacağını belirler.
list-style-image	url none inherit (list-style-image: url(image))	Liste simgesi olarak resim kullanılmasını sağlar.
list-style-position	inside outside inherit	Listenin ilk satırının diğer satırlar aynı hizada olup olmayacağını belirler.
list-style	list-style-type list-style-position url (list-style: lower-roman inside gibi.)	list-style özelliği list-style-type, list- style-position ve list-style-image özelliklerinin kısa yoludur.

Stillerle Kutu Modeli Özelliklerinin Kullanımı

Komut	Değer	Açıklama
margin	Yüzde Sayısal değer. (10px gibi. Bu durumda nesnenin dört tarafında 10 piksel boşluk bırakılır.) margin-top margin-bottom margin-top margin-right ile her birine ayrı değerler de verilebilir.)	Kenar Dış Boşluğu. Nesnenin dışındaki nesnelere olan aralığını belirler.
padding	Yüzde Sayısal değer. (10px gibi. Bu durumda nesnenin sınırları ile içindeki nesneler arasında 10 piksel boşluk bırakılır.) padding-top, padding-right, padding-bottom, padding-left, ile her birine ayrı değerler de verilebilir.)	Kenar iç boşluğu.
border-width	thin medium thick uzunluk değeri inherit Sayısal değer. (2px gibi. border-top-width, border-bottom-width, border-left-width, border-right-width komutlarıyla her bir kenarı için faklı kalınlıklar verebiliriz.)	Kenarlık genişliği (kalınlığı)
border-color	Renk adı RGB değeri (red veya #FF0000 gibi. border-top-color, border-bottom-color, border-left-color, border-right- color komutlarıyla her bir kenarı için faklı renk de verebiliriz.)	Kenarlık rengi
border-style	none dotted dashed solid double groove ridge inset outset (border-top- style, border-bottom-style, border-left-style, border-right-style komutlarıyla her bir kenarı için faklı style belirleyebiliriz.)	Kenarlık stili
border	border-width border-style border-color (border: 2px double blue gibi.)	Kenarlık kalınlık, style ve rengi.
border-color	<renk> transparent inherit , border-[top,left,right,bottom]-color , ayrı ayrı verilebilir.</renk>	kenarlık rengini belirler.

Stillerle Boyut Tanımlama Özelliklerinin Kullanımı

Komut	Değer	Açıklama
width	boyut yüzde auto inherit	Blok seviyeli ve yerlestirilen elemanların(img, input, textarea vb.) tümü bir genislik(width) değeri alır.
height	boyut yüzde auto inherit	Yükseklik
min-width, min-height	boyut yüzde inherit	Asgari genişlik ve yükseklikler
max-width, max-height	boyut yüzde none inherit	Azami genişlik ve yükseklik

Stillerle Görsel Sonuç Özelliklerinin Kullanımı

Komut	Değer	Açıklama
overflow	 inherit - kalıtsallık üst elemanın değerini alabilir visible - kutu sınırları dışına tasan alanları göster hidden - kutu sınırları dışına tasan alanları gizle. scroll - kutu sınırlarını tasan kısmı kaydırma çubuğu ile ulaşmamızı sağlar. auto - kutu sınırlarını asması durumunda scroll ile aynı etkiyi yapar. 	Kutu elemanının içeriğinin sınırları taşması durumunda nasıl davranacağını bildiren bir özelliktir.
clip	rect(top, right, bottom, left) auto inherit	Kırpma. Clip'e atanan Değerler em cinsinden verilerek esnek sonuçlar elde edilebilir. Ayrıca negatif değerlerde alabilir. Negatif değer atandığında eleman kutusunun dışına doğru kırpma yapacaktır. Clip, elemanın içeriğinin belli bir bölümünün gösterilmesi sağlar.
visibility	visible hidden collapse inherit collapse tablo elemanlar için hidden etkisi yapar. Ancak burada bir fark var gizlenen elemanlar yerini korumaz.	Görünürlük(visibility) özelliği nesnelerin görünümünü belirler.

Stillerle Akış Özelliklerinin Kullanımı

Komut	Değer	Açıklama
float	left right none inherit	float özelliği bir elemanı(img, table, div vd.) konumunu belirlemek(sağa veya sola) için kullanılır. Siz bir elemana flaot:left değeri verirseniz diğer elemanlar o elemanın sağından akar
sağından akar Resim ve metin elemanları diger elen Clear özelliği floating uygulanmayan e uygulandığı elemanı floating elemanı sekilde saga atar, none özelliği elema none left right both değeride iki floating elemanın asagısına atar. clear özelliği CSS ile tablosuz web site metodunun en önemli ögelerinden b	Resim ve metin elemanları diger elemanlara göre floting eleman olarak tanımlanır. Clear özelliği floating uygulanmayan elemana köse tanımı yapar. left değeri uygulandıgı elemanı floating elemanın bir alt soluna atar, right özelliği benzer sekilde saga atar, none özelliği elemanın başlangıç değerini geri döndürür ve both değeride iki floating elemanın asagısına atar. clear özelliği CSS ile tablosuz web sitesi olusturma metodunun en önemli ögelerinden biridir.	
Normal Akış		Eğer sayfalardaki elemanlara herhangi bir konumlandırma yapılmamış ise veya float değeri verilmemiş ise sayfaya normal akış konumlandırması uygulanır.

Stillerle Konumlandırma Özelliklerinin Kullanımı

Komut	Değer	Açıklama
position	static relative absolute fixed inherit	Konumlandırma özelliği
top, right, bottom, left	<uzunluk> <yüzde> auto inherit</yüzde></uzunluk>	Üst, sağ, alt ve sol konum belirleme
z-index	<sayısal değer=""> auto inherit</sayısal>	z-index, konumlandırma elemanları uygulanmış katmanların z- eksenindeki konumu belirlemek için kullanılır. z-index tüm elemanlara uygulanamaz sadece konumlandırma değeri atanmış elemanlara uygulanır. Bunun anlamı eğer bir elemanı aynı z ekseni üzerinde üste veya alt sıraya atmak istiyorsanız ilk olarak konumlandırma değeri atamalısınız. z-index değeri dokümanımızdaki elemanların istiflenme sıralarını belirler. z-index elemanları z-ekseninde nerede duracağınız belirler, üstte veya altta. z-indeks değeri küçük olan elemanlar altta büyük olan elemanlar ise üstte görünecektir. z-index sadece position değeri relative, absolute ve fixed olan nesnelere uygulanır.

Stillerle Görünüm (display) Özelliklerinin Kullanımı

Komut	Değer	Açıklama
display	none inline block inline-block list-item runin table inline-table table-row- group table-header-group table-footer- group table-row table-column-group tablecolumn table-cell table-caption inherit	 block: Bu değer bir elemanın bir blok kutusu üretmek için kullanılır. inline: Bu değer bir elemanın bir veya daha fazla sayıda satır içi kutu üretmek için kullanılır. list-item: Bu değer bir elemanın (HTML'deki gibi) bir blok kutu ve bir liste öğesi satır içi kutusu üretmek için kullanılır. none: Bu değer bir elemanın biçimleme yapısında hiçbir kutu üretmemesine sebep olur (yani, elemanın yerleşime hiçbir etkisi olmaz; başka bir deyişle, gösterilmez). Ayrıca alt elemanlar da görünmez ve bu davranış alt elemanların display nitelikleriyle etkisiz kılınamaz. table, inline-table, table-row-group, table-column, table-columngroup, table-header-group, table-footer-group, table-row, tablecell ve table-caption: Bu değerler bir elemanın bir tablo elemanı gibi davranmasına sebep olur.

Stillerle Tablo Özelliklerinin Kullanımı

Komut	Değer	Açıklama
table-layout	auto fixed inherit	Tablo genişliği özelligi. table-layout:fixed değeri tablo genişliğimizi sabitlememizi sağlar. tablelayout: auto ise tablo genişliğini içeriğe göre arttırır.
caption-side	top bottom inherit	Tablonun başlığının(caption) nerede olacağını belirler.
border-collapse	collapse separate inherit	tablo hücrelerinin etrafındaki kenarlık ile olan mesafeyi azaltmamızı veya açmamızı saglar. separate ataması yapılmıs ise hücreler arasına bir kaç piksellik mesafe konur, HTML içinden bu boslukları etiketinin cellspacing özelligi ile kaldırmak istesek bile tarayıcılar bu seferde iki kenarlıgı bir bir üstü gösterecektir. Eger collapse değeri verilirse iki hücre arasındaki mesafe kalkacaktır ve tek kenarlık görünecektir.
border-spacing	<uzunluk> <uzunluk>? inherit</uzunluk></uzunluk>	border-spacing, normal uzunluk değerleri alır ve bordercollapse: separate ile birlikte kullanılır. Eğer tek değer verilirse bu tüm kenarlar için geçerlidir, iki değer atanırsa ilki yatayda boşluk için ikincisi dikey de boşluk vermek için tanımlanmıştır.
empty-cells	show hide inherit	Bir tabloya bazı değerler atadığımızda(örneğin kenarlık değeri) boş olan hücreler bu değeri görmez, bu durumu düzeltmek için bos olan hücreler için bos karakter() koyarız empty-cells özelliği ile bos kalan hücrelere nasıl davranması gerektiğini bildirebiliriz.

Stillerle Kullanıcı Arayüzü Özelliklerinin Kullanımı

Komut	Değer	Açıklama
outline-style	one dotted dashed solid double groove ridge inset outset inherit	Dış hattın biçemini belirlemek için kullanırız. none : Dış hat görünmesin anlamındadır. dotted : Dış hat belirli aralıklarla dizilmiş noktalardan oluşur. dashed : Dış hat belirli aralıklarla dizilmiş çizgi parçalarından(tire) oluşur. solid : Dış hat bir çizgi olara görünür double : iki çizgi seklinde görünürki çizginin kalınlıkları ile aralarındaki boşluğun genişliğinin toplamı outline-width değerine eşittir. groove : Dış hat tuvale oyulmuş gibi görünür. ridge : groove'un zıddı: Dış hat çizgisi kabartma gibi görünür. inset : Dış hat çizgisinin kutunun tamamını tuvale gömülü gibi görünmesine sebep olur. outset :Dış hat çizgisi kutunun tamamının kabartma gibi görünmesine sebep olur.
outline-width	thin medium thick length inherit	Atandığı elemanın dışına çizilen hattın kalınlığını belirlemek için kullanırız.
outline-color	color invert inherit	Dış hat çizgisi rengi
outline	<outline-color> <outline-style> <outlinewidth> inherit</outlinewidth></outline-style></outline-color>	Dış hat çizgisi(outline) özelligi yukarıda gördügümüz özelliklerin tek seferde kullanılmasını sağlayan bir kısa yoldur.





Kaynak: 1- <u>http://nefarkederki.com/html-dersleri</u> 2- HTML5, İBRAHİM ÇELİKBİLEK, KOD LAB

HTML 5.0

- HTML 5, World Wide Web Consortium (W3C) ve Web Hypertext Application Technology Working Group (WHATWG)'nin ortak bir çalışmasıdır.
 - Sadeleştirilmiş, düzeltilmiş HTML4 ve XHTML dillerinin son sürümüdür.
 - Daha fazla yapısal elemana, form elemanlarına ve yeni özelliklere sahiptir.
 - Gelişmiş multimedya desteği sunmaktadır.
 - İçerisinde yeni teknolojiler barındırır ya da önceden var olan teknolojilerin kullanımını sağlar.
 - JavaScript ve CSS3 teknolojilerini daha etkili bir şekilde kullanmamıza imkan tanır.

HTML5 yapısı

- Bir HTML5 belgesi her zaman DOCTYPE etiketi ile başlar ve ardından <html>, <head>, <body> etiketleri tanımlanır.
 - <!DOCTYPE html>
 - <html>
 - <head>
 - <meta charset="utf-8">
 - <title>Sayfa Başlığı</title>
 - </head>
 - <body>... </body>
 - </html>

HTML5 yapısı

<!DOCTYPE html>

- DOCTYPE tanımlaması yapmamış iseniz web sayfanız tarayıcının var olan eski kurallarına göre değerlendirilecek ve gösterilecektir. Sayfa görünümü ya da elemanların yerleşimi, kullanılan tarayıcıya bağlı olarak değişiklik gösterebilir.
- Örneğin HTML 5.0 da charset tanımlaması
 - e <meta charset="utf-8">
- Önceki HTML sürümlerinde charset tanımlaması;
 - <meta http-equiv="content-type" content="text/html;charset=utf-8" />

HTML5 DILI ETIKETLERI

- contenteditable :true | false
- Eleman içeriğinin düzenlenebilir olup olmayacağını ayarlamak için kullanılır.
- Bu özelliğe true değeri atanmışsa eleman içeriği kullanıcı tarafından düzenlenebilir, false değeri atanmışsa eleman içeriği kullanıcı tarafından düzenlenemez. Bu özelliği, değeri ile beraber kullanmak şarttır.

HTML5 DILI ETIKETLERI

• Örnek

- <!DOCTYPE html>
- <html> <head>
- <meta charset="utf-8" />
- <title> contenteditable özelliği</title>
- <style type="text/css">
- </style> </head>
- ody>
- <div contenteditable="true"> Düzenlenebilir İçerik</div>
- <div contenteditable="true">
- </div></div>
- </body></html>

HTML5 DILI ETIKETLERI

• Sayfanın normal (varsayılan) ekran çıktısı:



 Birinci div içerisinde bulunan metnin kullanıcı tarafından değiştirilmesi ve resmin küçültülmesi. Ekran çıktısı:



- Web sitelerini kodlarken belli yapısal kısımlara ayırırız. Üst kısım, içerik kısmı, sol-sağ kolonlar, menü alanı, alt kısım ve içerik alanı.
- HTML5 de bu alanların her biri için bir element tanımlanmıştır. Daha önce kodlama yaparken bu alanlar için genel <div> elementi kullanılırken, yeni elementlerle bu tekil element (<div>) yerine her kısım için bir element kullanma imkanı olmaktadır.

 Semantik olarak doğru ve daha iyi bir yapı sağlamak için HTML5 e eklenen yeni elemanlar:

Etiket	Tanım
section	Bir doküman içindeki bölüm için kullanılır.
header	Bu elemanı kullanarak sayfa hakkında bilgiler içeren başlık alanları, bölüm veya alt bölüm başlığı, yönlendirme alanları için başlık alanları oluşturabilirsiniz. Genelde h1, h2, h3 h6 elemanlarını ya da hgroup elemanını içerir.
hgroup	Başlıkların bir bölümüdür, <h1> den <h6> ya kadardır, en büyük olan bölümün başlığı olur, ve diğerleri de alt başlık olur.</h6></h1>
footer	Web sitemizde alt bilgi olarak belirlediğimiz kısmı tanımlamak için kullanabiliriz. Sayfa altlığı olarak kullanılır. Yazar hakkında bilgi, dokümanın tarihi, iletişim bilgisi ve telif hakları bilgilerini içermelidir.
nav	Navigasyon (<i>yönlendirme</i> , link) bölümü için kullanılır. Diğer sayfalara link verilen kısımlar, menü oluşturduğumuz kısımları tanımlamak için kullanabiliriz.
article	Sayfanın ana içerik alanında anlamsal bir bütünlüğü olan, yani farklı konu başlıklarına ya da davranışlarına göre ayrılmış alt bölümler oluşturmak için kullanılır.





- <section>: Sayfa içerisinde genel bölümler (ana içerik bölümü ya da diğer bölümler) oluşturmak için kullanılır.
 section { display: block; }
- <nav>: Navigasyon (yönlendirme, link) alanları oluşturmak için kullanılır.
 - nav elemanı link oluşturmak için kullandığımız a elemanının ya da ul ile ol elemanlarının yerlerine kullanılmaz. Sadece bu elemanları kapsar (başka elemanları da kapsayabilir) ve içerdiği eleman gurubunu bir navigasyon alanı olarak tanımlar.
 - nav { display: block; }

- <article>: Sayfanın ana içerik alanında anlamsal bir bütünlüğü olan, yani farklı konu başlıklarına ya da davranışlarına göre ayrılmış alt bölümler oluşturmak için kullanılır. Örneğin' bir blog sayfasında ana içerik bölümü içindeki farklı blog yazı alanlarını bu elemanla oluşturabilirsiniz.
 - article { display: block; }
- <footer>: Bu elemanı kullanarak sayfa ya da bölümler için alt bilgi alanları oluşturabilirsiniz. Alt bilgi alanları genelde içerik yazarı hakkında bilgi ve içeriğin oluşturulma tarihi yada sayfa sahibi ile ilgili iletişim bilgilerini içerir.
 - footer { display: block; }

HTML5 Yapısal Elementler Örnek:

- <div id="menu">
- •
- Birinci Link
- İkinci Link
- Dördüncü Link
- •
- </div>
- Şeklindeki bir kullanım yerine aşağıdaki kullanımı tercih edebilirsiniz.

- <nav>
- •
- Birinci Link
- İkinci Link</s>
- Üçüncü Link
- Dördüncü Link
- </nav>

- <html> <head>
- <title>HTML 4</title> </head>
- <body>
- <div class="header">
- <div class="nav"></div> </div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></ti></ti></ti>
- <div class="section">
- <div class="article">
- <div class="header"></div></div>
- <div class="figure"></div></div>
- <div class="footer"></div>
- <div class="dialog"></div></div>
- </div> </div></div></div></div>
- <div class="aside"></div></div>
- <div class="footer"></div></div>
- </body> </html>





- <!DOCTYPE html>
- <html> <head>
- <title>HTML 5</title> </head>
 <body>
- <header><nav></nav></header>
- <section>
- <article> <header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header></header
- <figure></figure><footer></footer>
- </article><dialog><dialog></section>
- <aside></aside> <footer></footer>
- </body>
- </html>

Örnek:

- <!DOCTYPE html>
- <html> <head>
- <meta charset="utf-8, tr" >
- <title>nav Elemanı</title>
- <style type="text/css">
- body { background-color: #E5E5E5; }
- nav
- { width: 238px; height: 300px;
- background: none repeat scroll 0 0 #FFFFF;
- border: 1px solid #E5E5E5;
- font-family: Helvetica, Arial, sansserif;

• nav h3 {

- border-bottom: 1px solid #DDDDDD;
- font-size: 12px;
- font-weight: 100;
- margin: 0;
- padding: 13px 15px;
- text-transform: uppercase;
- }
- nav #d1 { padding: 10px; }

Örnek:

- div#d1 .plist {
- border-bottom: 1px solid #E5E5E5;
- margin: 5px 0;
- padding: 0 0 5px 30px ;
- position: relative; font-size: 12px;
-]
- div#d1 .plist .title-numara a {
- color: #222222;
- display: block; font-size: 13px;
- margin-bottom: 2px;
- text-decoration: none;
- color: #c43f3f;

- div#d1 .plist .numara {
- background: url("number.png") no-repeat scroll left top transparent;
- color: #FFFFF;
- line-height: 20px !important;
- top: 3px; left: 0px; width: 20px;
- display: block;
- position: absolute;
- text-align: center;
- font-weight: bold;
- }

•

Örnek:

- </style> </head> <body>
- <nav> <h3>
- Duyurular & Haberler
- </h3>
- <div id="d1">
- 01
-
- HTML5 Kitabı
 Güncellendi!
- HTML5 Kitabının içeriği güncellendi.
 HTML diline yeni başlayacaklar için ve HTML5 getirdiği yenilikleri öğrenmek isteyenlere duyurulur.

- 02
- < a href="#">ibrahimcelikbilek.com
- programlama ve web tasarımı ile ilgili makaleler ve sorularınıza cevaplar bulabileceğiniz bir site
- 03
-
- Kodlab
- Bilişim Yayıncılığının yeni adresi!
- </div> </nav> </body> </html>

•

HTML5 Yapısal Elementler Örnek:



- <hgroup>: Başlık elemanlarını (h1...h6) gruplandırarak belge yada bölüm başlığı oluşturmak için kullanılır.
 - hgroup{ display: block; }
 - <hgroup>
 - <h1>Web Developers</h1> <h2>HTML5</h2>
 - </hgroup>
- <header>: Bu elemanı kullanarak sayfa hakkında bilgiler içeren başlık alanları, bölüm veya alt bölüm başlığı, yönlendirme alanları için başlık alanları oluşturabilirsiniz. Genelde h1, h2 ... h6 elemanlarını ya da hgroup elemanını içerir.

Örnek:

- <!DOCTYPE html>
- html>
- <head>
- e <meta charset="utf-8,tr" />
- <title>header elemani</title>
- <style type="text/css">
- header
- { font-family: Cavalero BT,serif;
- width:650px;
- height:54px;
- background-color:#f9f3f3;
- text-indent:5px;

Css float, html align özelliği aynı

- header img { float:right;}
- header h1 {margin:0px;}
- </style> </head> <body>
- <header>
-
- <h1>Her Yönüyle HTML5</h1>
- </header>
- </body> </html>



•

 section, nav, article, aside, hgroup, header, footer elemanları için tarayıcı desteği aşağıdaki tabloda görülmektedir. section, nav, article, aside, hgroup, header, footer elemanları blok seviyesinden elemanlardır. Tarayıcı blok seviyesi elemanından önce ve sonra bir satır sonu oluşturur.

	Tarayıcı Desteği			
Chrome Firefox Internet Explorer Opera				Safari
5	4	9.0	11.10	4.1

HTML5 Anlamlı Blok Elementler

 Yapısal elementlerden daha çok ayrıntıya inen elementlerde bulunmaktadır.

Etiket	Tanım
aside	Kendi içinde bir içeriği olan, tüm sayfadan bağımsız içerikler için kullanılır. Sahip olduğu içerik çevresel içerikle ilişkili olmalıdır. Blog ana sayfalarında yer alan bağımsız her bir gönderi için kullanılır.
figure	Genelde kullandığımız şekil ve açıklaması için eklenmiş güzel element. Özerk (stand-alone) bir içeriğin gruplanması için kullanılır (örn: video)
dialog	Diyalog elementi, karşıklık sohbet içeren sayfaları daha kolay kodlamamız için eklenmiştir. Bu element <dd> ve <dt> elementlerini alt element olarak kullanır.</dt></dd>

<aside></aside>	<figure id="fig2"></figure>	<dialog></dialog>
	<legend>resim açıklaması </legend>	<dt>Yönetici</dt>
		<dd></dd>
		edto Zivaretcie/dto
		<dd> </dd>
		<dt>Yönetici</dt>
		<dd></dd>
		edto Zivaratoie/dto
		edds cidd
		suosida

- <aside>: İçerik bölümleri ile alakalı ek bilgi alanları oluşturmak için kullanabileceğiniz bir elemandır. Sayfanın ana bölümlerinin dışında sayfa kenarlarında reklam alanları, yan açıklama ve ek bilgi alanları oluşturmak için de kullanılabilir.
 - article { display: block; }
- <figure>, <figcaption> (<legend>):figure etiketi; resim, fotoğraf alanları ya da media alanları tanımlamak için kullanılır. figure elemanı ile oluşturulan alanlara bir metin iliştirmek için figcaption elemanı kullanılır.
- <dialog>: Diyalog elementi, karşıklık sohbet içeren sayfaları kodlamayı sağlar. Bu element <dd> ve <dt> elementlerini alt element olarak kullanır.

- <!DOCTYPE html>
- <html> <head>
- <meta charset="utf-8,tr" />
- <title>figure ve figcaption elemani</title></head>
- ody>
- <figure>
- <img alt="html5Logo"
 src="html5.png" />
- <figcaption> HTML5 resim
 başlığı</figcaption>
- </figure> </body>
- </html>


Etiket	Tanım
<bdi></bdi>	parent elementin metin doğrultusu yönününde olmayan metinler için kullanılır. (arapça karakterleri gibi- <bdi> إيان </bdi>)
<command/>	Bir buton, radyo butonu ve bir onay kutusu tanımlamak için kullanılır.
<details></details>	Bir dokümanın bir parçasının yada tamamımın detaylarını açıklamak için kullanılır.
<summary></summary>	<details> elemanı içinde özet veya açıklama için kullanılır.</details>
<mark></mark>	Yazıları vurgulamak için kullanılır. Özellikle belirtilmek istenilen metinler (highligt) için kullanılır. (arka plan sarı)
<meter></meter>	Alt ve üst değerleri bilinen ölçümler için kullanılır.

Etiket	Tanım
<nav></nav>	Navigasyon bölümü için kullanılır.
<progress></progress>	İşin ilerleme durumunu göstermek için kullanılır. Flash'taki loading meselesi.
<ruby></ruby>	Uzakdoğu alfabelerinde kullanılan harflerin veya notların okunuşunu, açıklamasını göstermek için kullanılır. (Ruby için)
<rt></rt>	<ruby> elemanın bilgi notunu açıklamak için kullanılır.</ruby>
<rp></rp>	<ruby> elemanını desteklemeyen tarayıcıların göstermesi istenilen içerik için kullanılır.</ruby>
<time></time>	Saat, tarih veya her ikisinin de gösterimi için kullanılır.
<wbr/>	(WORD BREAK). Satır boşluğu bırakmak için kullanılır.

Ruby 'hızlı ve kolay', nesneye yönelik yazılım geliştirmeye yarayan yorumlanan bir betik dilidir.

- <!DOCTYPE html> <html> <head>
- e <meta charset="utf-8,tr" />
- <title>meter</title> </head>
- <body> <form> 1.Ölçüm
- <meter min="0" low="50" high="80"
 max="100" value="100"> </meter>
- 2.Ölçüm
- <meter min="0" low="50" high="80" max="100" value="70"> </meter>
- 3.Ölçüm
- <meter min="0" low="50" high="80" max="100" value="20"> </meter>
- •
- </form> </body> </html>

U Opera 📄 meter	×	÷
← → Ə G- (■Yere		
1.Ölçüm		
2.Ölçüm		
3.Ölçüm 📃		

 low –high değerleri ve arası yeşil diğer noktalarda sarı

- <!DOCTYPE html>
- <html> <head>
- <meta charset="utf-8,tr" />
- <title>progess</title>
- <script type="text/javascript">
- var prg = function () {
- var prg= document.querySelector('progress');
- var num=document.querySelector ('[type="number"]');
- prg.value=num.value;
- }
- </script> </head> <body> <form>
- <progress max="100"></progress>
</progress>

- <input type="number" min="10" max="100"
 step="10" onchange="prg();" />
- </form> </body> </html>



- number elemanı içerisinde saklanan değeri progress elemanının value özelliğine atandı.
- Bu durumda number elemanı içerisindeki değer her değiştiğinde prg() fonksiyonu çağrılacak ve ilerleme pozisyonu değişecektir.

Yeni medya elementleri

HTML5 yeni medya standartları sunmakta:

Etiket	Tanım					
<audio></audio>	Multimedya içerik, sesler, müzik ve diğer akışlar için					
<video></video>	Video içerikleri için, film gibi diğer medya akışları için					
<source/>	<audio> ve <video> elemanlarının içinde, kaynak adreslemek için kullanılır.</video></audio>					
<embed/>	Flash ve java gibi eklentileri sayfa içine yerleştirmek için kullanılır.					
<track/>	Medya oynatıcılarında içerik metnini göstermek için kullanılır.					

- Bugüne kadar web sayfalarında ses dosyası çalmak için bir standart bulunmuyordu ve genellikle ses oynatmak için flash gibi eklentiler kullanılıyordu.
- HTML5 ile gelen <audio> elemanı ile ses dosyası çalmak mümkün olmaktadır.
- <audio> Web sayfaları içerisine ses dosyaları eklemek için kullanılan elemandır. Ses parçalarının çalınmasını sağlar.

Audio Türleri

• Varsayılan olarak, audio elementine uygun 3 tür mevcut:

Format	IE 9	Firefox 3.5	Opera 10.5	Chrome 3.0	Safari 3.0
Ogg Vorbis	Hayır	Evet	Evet	Evet	Hayır
MP3	Evet	Hayır	Hayır	Evet	Evet
Wav	Hayır	Evet	Evet	Evet	Evet

audio elementinin özellikleri

Özellik (Attribute)	Değer	Açıklama		
autoplay	autoplay	Ses dosyası yüklenir yüklenmez çalınmasını sağlar		
controls	controls	Ses dosyası oynatıcısı kontrollerinin gösterilmesini sağlar		
width, height	pixel	Ses oynatıcısının genişliğini / yüksekliğini pixel cinsinden gösterir		
Іоор	Іоор	Ses dosyasının sonsuz sayıda tekrar tekrar oynatılmasını sağlar		
muted	muted	Ses dosyasını sessiz olarak başlatır.		
preload	auto metadata none	Sayfa yüklendiği zaman ses dosyasının nasıl yüklenmesi gerektiğini belirtir		
src	URL	Ses dosyasının adres bilgisidir		

- src: Eleman tarafından çalınacak ses dosyasını tanımlamak için kullanılır. Bu özelliğe ses dosyasının saklandığı bir URL adresi de yazılabilir.
 - <!DOCTYPE html>
 - <html> <head> <meta charset="utf-8,tr" />
 - <title>audio elemani</title> <style type="text/css">
 - audio { margin: 5px; }
 - </style> </head> <body>
 - <audio src="kalimba.ogg" controls autoplay > Tarayıcınız HTML5'in eklentisiz ses dosyası çalma özelliğini desteklememektedir.
 - </audio>
 - </body> </html>



 controls: Denetim çubuğunun (video kontrol düğmelerinin) gösterilip gösterilmeyecek ayarlamak için kullanılan boolean tipi bir özelliktir. Kontrol düğmelerinin gösterilmemesini sağlayıp, kendi oluşturduğunuz kontrol düğmelerini kullanabilir ya da autoplay özelliğini kullanıp, ses dosyasının sayfa yüklendiğinde otomatik olarak çalıştırılmasını sağlayabilirsiniz.



- preload: Sayfa yüklendiğinde, medya içeriğinin çalışmaya hazır olması için ön belleğe alınmasını sağlar. Eğer autoplay özelliği kullanılmışsa, bu işlem zaten otomatik olarak yapılır, none, metadata, auto değerlerinden birini alabilir.
- none değeri atanırsa, medya içeriği için preload (ön yükleme) işlemi yapılmaz, metadata değeri atanırsa performans açısından önemli olan medya içeriği ile ilgili çeşitli bilgiler (boyutu, toplam süre, kaynak listesi) için ön yükleme yapılır, auto değeri atanırsa, ön yükleme işlemi otomatik olarak tüm medya içeriği için yapılacaktır.

Web üzerinde video

- Günümüzde birçok web sayfası içinde, farklı yöntemlerle videoların oynatıldığı görülmektedir. Ancak web sayfalarında video gösterimi için, herhangi bir standart bulunmamaktadır.
- Günümüzde, flash gibi bir eklenti ile videolar gösterilmekte. Bununla beraber, tarayıcıların hepsi aynı tür eklentilere sahip değiller.
- HTML5, <video> elemanı ile bu gösterim için hem bir standart sunmakta, hem de dışa bağımlı eklentilerin kullanım ihtiyacını ortadan kaldırmaktadır.

Video Formatları

 Bugün için, <video> elemanı 3 farklı formatı desteklemektedir.

Format	IE	Firefox	Opera	Chrome	Safari
Ogg	No	3.5+	10.5+	5.0+	No
MPEG 4	9.0+	No	No	5.0+	3.0+
WebM	No	4.0+	10.6+	6.0+	No

Ogg = Ogg dosyaları Theora video codec'i ve Vorbis audio codec'i ile çalışır.

MPEG4 = MPEG 4 dosyaları H.264 video codec'i ve AAC audio codec'i ile çalışır.

WebM = WebM dosyaları VP8 video codec'i ve Vorbis audio codec'i ile çalışır.

Video elementinin özellikleri

Özellik (Attribute)	Değer	Açıklama
autoplay	autoplay	Video yüklenir yüklenmez oynatılmasını sağlar.
controls	controls	Video kontrollerinin gösterilmesini sağlar . Bu özellik play, pause, volume (oynat, durdur, ses) gibi kontrolleri video oynatıcıya eklemek için kullanılır. Her tarayıcının kendine özgü kontrolleri bulunmaktır.
width, height	pixel	Videonun genişliğini / yüksekliğini pixel cinsinden gösterir .
Іоор	Іоор	Videonun sonsuz sayıda tekrar tekrar oynatılmasını sağlar
muted	muted	Video dosyasını sessiz olarak başlatır.
poster	URL	Video yükleninceye kadar gösterilmesi istenilen görselin adresidir .

Video elementinin özellikleri

Özellik (Attribute)	Değer	Açıklama
preload	auto, metadata, none	Sayfa yüklendiği zaman videonun nasıl yüklenmesi gerektiğini belirtir
src	URL	Videonun adres bilgisidir
p	İfade	<video> elementini desteklemeyen tarayıcılar otomatik olarak bu elemanı ve <source/> elemanını yok sayacaktır. Bu da videoyu göstermeyeceği anlamına gelir. Kullanıcının bu durumda bilgilendirilmesi etiketi kullanılabilir. Böylece <video> elemanını desteklemeyen tarayıcılar elemanı içinde verdiğiniz uyarı mesajını göreceklerdir.</video></video>
source	Ogg, MPEG4, WebM	<video> elemanını destekleyen tarayıcının format desteğine göre video adreslerini göstermektedir. Tarayıcı yalnızca desteklediği formattaki videoyu gösterecektir.</video>

HTML5 Video

- HTML5'te video göstermek için, ihtiyacınız olan şey:
- video width="320" height="240" poster="moviePoster.png" controls>
- <source src="movie.mp4" type="video/mp4" />
- <source src="movie.ogg" type="video/ogg" />
- Tarayıcınız HTML5'in eklentisiz video oynatma özelliğini desteklememektedir.
- </video>

Video elementinin özellikleri

- <!DOCTYPE html>
- <html><head>
- <meta charset="utf-8,tr" />
- <title>source</title>
- <style type="text/css">
- body {font-family:Calibri;font-size:16px;}
- </style>
- <script type="text/javascript">
- var vd1=function () {
- document.getElementById("videoE1").src = document.getElementsByTagName ("source")[0].src; }

- var vd2=function () {
- document.getElementById("videoE1").src = document.getElementsByTagName ("source")[1].src; }</script>
- </head>
- ody>
- video id="videoE1" controls>
- <source src="cart1.ogg" />
- <source src="cart2.ogg" />
- </video>
 Seçiminizi Yapınız:
- <select id="s1">
- <option onclick="vd1();">Video1-Çizgi Film
 </option>
- <option onclick="vd2();">Video2-Belgesel </option>
- </select> </body> </html>

Video elementinin özellikleri



Canvas Elementi

- Canvas elementi bir web sayfası üzerinde çizim yapmak için kullanılır. Çizimler web sayfasında Javascript ile beraber HTML5 içinde kullanılarak elde edilir.
- <canvas> elemanı, her bir pixelini kontrol edebileceğiniz dörtgen bir alandır. Çember, daire, çokgen ve karakter çizmek için farklı metotları bulunmaktadır.

Etiket	Tanım
<canvas></canvas>	Script ile grafikler oluşturmak için kullanılır.

<canvas id="myCanvas" width="200" height="100"></canvas>

Canvas elementi oluşturmak

- <canvas id=" cizimalani" width="200"height="100"></canvas>
- <script type="text/javascript">
- var c=document.getElementById("cizimalani");
- var cxt=c.getContext("2d");
- cxt.fillStyle="#FF0000";
- cxt.fillRect(0,0,150,75);
- </script>
- bu örnekte olduğu gibi. Javascript canvas elementini bulmak için id'yi kullanmakta;
- var c=document.getElementById("cizimalanim");

Canvas'da koordinatları anlamak

 Yukarıdaki örnekte, fillRect metodunun parametreleri (0,0,150,75) olarak verilmektedir. Bu; 150x75 boyutlarında bir dikdörtgenin canvas elamanının 0,0 noktasından başlayarak çizilmesi anlamına gelir.



 Şekil 'de canvas elemanının 2 boyutu koordinat sistemi gösterilmektedir.

Canvas elemanı içinde örnek çizimler

- <!DOCTYPE html>
- <html> <body>
- <canvas id="myCanvas" width="360" height="210" style="border:1px solid #c3c3c3;"> Tarayıcınız CANVAS özelliğini desteklememektedir. Lütfen tarayıcınızı güncelleyiniz. </canvas>
- <script type="text/javascript">
- var c=document.getElementById("myCanvas");
- var cxt=c.getContext("2d");
- cxt.moveTo(0,0);
- cxt.lineTo(150,50);
- cxt.moveTo(0,0);
- cxt.lineTo(10,50);
- cxt.moveTo(360,210);
- cxt.lineTo(14,54);
- cxt.stroke();
- </script>
- </body></html>



Canvas elemanı içinde örnek çizimler

- <!DOCTYPE html> <html> <body>
- <canvas id="myCanvas" width="360" height="210" style="border:1px solid #c3c3c3;">
 Tarayıcınız CANVAS özelliğini desteklememektedir. Lütfen tarayıcınızı güncelleyiniz.</canvas>
- <script type="text/javascript">
- var c=document.getElementById("myCanvas");
- var cxt=c.getContext("2d");
- cxt.fillStyle="#FF0000";
- cxt.beginPath();
- cxt.arc(150,105,60,0,Math.PI*2,true);
- cxt.closePath();
- cxt.fill();
- var cxt2=c.getContext("2d");
- cxt2.fillStyle="#FFF";
- cxt2.beginPath();
- cxt2.arc(168,105,48,0,Math.PI*2,true);
- cxt2.closePath();
- cxt2.fill();
- </script> </body> </html>



Canvas elemanı içinde örnek çizimler

- <!DOCTYPE html> <html> <body>
- <canvas id="myCanvas" width="360" height="210" style="border:1px solid #c3c3c3;">
- Tarayıcınız CANVAS özelliğini desteklememektedir. Lütfen tarayıcınızı güncelleyiniz.</canvas>
- <script type="text/javascript">
- var c=document.getElementById("myCanvas");
- var cxt=c.getContext("2d");
- var grd=cxt.createLinearGradient(0,0,360,210);
- grd.addColorStop(0,"#FF0000");
 grd.addColorStop(1,"#EBD126");
- cxt.fillStyle=grd;
- cxt.fillRect(0,0,360,210);
- var kutu=c.getContext("2d");
- kutu.fillStyle="rgba(255, 255, 200, 0.4)";
- kutu.fillRect(50,50,250,110);
- var ciz=c.getContext("2d");
- ciz.moveTo(50,50);
 ciz.lineTo(300,160);
- ciz.lineTo(50,160);
 ciz.stroke();
- </script> </body> </html>

ciz.moveTo(300,50);



Web Depolama (Storage)

- HTML5, ziyaretçilerin bilgisayarlarında bilgi depolamak için iki yeni nesne önermektedir. Bunlar;
 - localStorage zaman sınırı olmaksızın bilgileri depolar
 - sessionStorage bir oturumluk depolama bir oturum süresince bilgi depolar
- HTML5'in bu iki yeni nesnesinden önce bu işlemler çerezler (cookies) kullanılarak yapılmaktaydı. Ancak çerezler, sunucuya her seferinde bir sorgu gittiğinde yeniden düzenlendiğinden verimliliği azaltarak sistemi yavaşlatmaları yüzünden fazla bilgi için kullanılamıyorlardı.
- HTML5, sunucuya her istek gittiğinde veri gönderilmediği için web sayfasının performansını olumsuz etkilemeden bilgilerin depolanmasına olanak sağlamaktadır.

Web Depolama (Storage)

- Bilgiler, farklı websayfaları için farklı alanlarda depolanır ve her web sayfası yalnızca kendi depolanan bilgilerine ulaşabilmektedir. Bilgileri depolamak ve yeniden erişmek için Javascript kullanılmaktadır.
- Çerezler ile saklanacak maksimum veri büyüklüğü 4 KB iken, Web Storage ile MB'lar seviyesinde veri saklanabilir.
- Aynı web sitesini iki farklı sekmede ya da pencerede çalıştırdığınızda kullanılacak çerez aynı olacağından, problemler yaşanabilmektedir. Bu sorun sessionStorage (oturum depolama) yapısı ile yaşanmaz. Çünkü bu yapıda veriler oturum bazında saklanır.

- IocalStorage Nesnesi :İstemci tarafında oturumdan bağımsız, kalıcı olarak veri saklamak için kullanılır. Yerel veri sadece kendisini oluşturan web sitesi tarafından kullanılabilir. Saklanacak veriler için bir zaman sınırlaması yoktur.
- Bilgi depolamak ve o bilgiye erişmek için;
 - <script type="text/javascript">

 localStorage.lastname="Smith";
 document.write(localStorage.lastname); </script>
- Yukarıda basit şekilde bir kullanım örneği verilmiştir.
 Aşağıdaki örnekte ise, ziyaretçinin sayfaya geliş sayısını depolayan ve ekrana yazan basit bir örnek bulunmaktadır.

- IocalStorage Nesnesi
- <script type="text/javascript">
- if (localStorage.pagecount)
- IocalStorage.pagecount=Number(localStorage.pagecount) +1; }
- else
- { localStorage.pagecount=1; }
- document.write("Visits "+ localStorage.pagecount + " time(s).");
- </script>

- Örnek:
- <!DOCTYPE html>
- <html><body> <div id="result"></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div>
- <script>
- else { document.getElementById("result").innerHTML="Sorry, your browser does not support web storage..."; }
- </script></body></html></script></body></html></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script

Örnek: Kullanıcının sayfa ziyaretini sayar;

```
<!DOCTYPE html>
<html>
<body>
<div id="result"></div>
<script>
if(typeof(Storage)!=="undefined")
 if (localStorage.pagecount)
    localStorage.pagecount=Number(localStorage.pagecount)
+1;
 else
   localStorage.pagecount=1;
 document.getElementById("result").innerHTML="You have
visited this page " + localStorage.pagecount + " time(s).";
else
 document.getElementById("result").innerHTML="Sorry, your
browser does not support web storage...";
</script>
Refresh the page to see the counter increase.
Close the browser window, and try again, and the
counter will continue.
</body>
</html>
```

You have visited this page 1 time(s).

Refresh the page to see the counter increase.

Close the browser window, and try again, and the counter will continue.

sesssionStorage Nesnesi

- Kullanım şekli localStorage nesnesi gibi olan bu nesne, bilgileri oturum süresince tutar ve ziyaretçi sayfayı kapattığı zaman bu bilgiler silinir.
- <script type="text/javascript"> sessionStorage.lastname="Smith"; document.write(sessionStorage.lastname); </script>

- Örnek: Kullanıcının sayfa ziyaretini sayar;
- <script type="text/javascript">
- if (sessionStorage.pagecount) {
 sessionStorage.pagecount=Number(sessionStorage.pagecount
) +1; }
- else { sessionStorage.pagecount=1; }
- document.write("Visits "+sessionStorage.pagecount+" time(s) this session.");
- </script>

Adresi:

http://www.w3schools.com/html5/tryit.asp?filename=tryhtml5_webstorage_sess ion_pagecount

```
<!DOCTYPE html>
<html> <head> <script>
function clickCounter()
if(typeof(Storage)!=="undefined")
  { if (sessionStorage.clickcount)
    sessionStorage.clickcount=Number
(sessionStorage.clickcount)+1;
  else
               sessionStorage.clickcount=1;
  document.getElementById("result").innerHTML="You have
clicked the button " + sessionStorage.clickcount + " time
(s) in this session.":
else {
  document.getElementById("result").innerHTML="Sorry, your
browser does not support web storage..."; }
</script>
</head>
<bodv>
<button onclick="clickCounter()" type="button">Click me!
</button>
<div id="result"></div>
Click the button to see the counter increase.
Close the browser tab (or window), and try again, and
the counter is reset.
</body> </html>
```

Click me!

You have clicked the button 2 time(s) in this session.

Click the button to see the counter increase.

Close the browser tab (or window), and try again, and the counter is reset.

Adresi:

http://www.w3schools.com/html5/tryit.asp?filename=tryhtml5_webstorage_sess ion

Yeni Input type özellikleri

 HTML5 daha zengin ve tarayıcı kontrollü formlar oluşturmak için yeni [input] tiplerine sahiptir.

Etiket	Tanım
tel	Telefon numarası girişi
search	Aranılacak metin ya da metin guruplarını yazmak için kullanılır
url	Bir URL adresi girmek için kullanılır.
email	bir veya daha fazla eposta adresi girişi
datetime	tarih veya zaman değeri
date	tarih değeri
month	Ay değeri
week	Hafta değeri
time	Zamanın tipine dair değer
datetime-local	Yerel zamana ait değer
number	Sayısal değer
range	Verilen aralık için sayısal değer
color	Hexadecimal kod değeri için giriş

 Bu tipler henüz tüm tarayıcılar ve/veya versiyonları tarafından desteklenmemektedir.

Input type özellikleri

Tarayıcı desteği

Input type	IE	Firefox	Opera	Chrome	Safari
email	Hayır	4.0	9.0	10.0	Hayır
url	Hayır	4.0	9.0	10.0	Hayır
number	Hayır	Hayır	9.0	7.0	Hayır
range	Hayır	Hayır	9.0	4.0	4.0
Tarih seçiciler (date,time vs.)	Hayır	Hayır	9.0	10.0	Hayır
search	Hayır	4.0	11.0	10.0	Hayır
color	Hayır	Hayır	11.0	12	Hayır

Input type özellikleri

- Tarayıcı desteklemiyorsa, kullanıldığınız bu [input] elemanını normal bir text input olarak gösterecektir. Ancak doğrulama (validation) için ayrıca kodlama yapmanız gerekecektir.
- email, url, tel, number gibi tanımlamaların doğrulama ve kontrol dışında en önemli özelliği mobil cihazların klavye diziliminin değişmesidir.


Input Type – tel

• Girdi değeri bir telefon numarası olduğu durumlarda kullanılır.

Input Type – email

- Girdi değeri eposta olduğu durumlarda kullanılır. Form onaylandığında Eposta alanının değeri otomatik olarak doğrulanır. Örnek :
- Anasayfa: <input type="email" name="user_email" />

Input Type – search

 Girdi alanı bir arama alanı olduğu durumlarda kullanılır. Destek veren tarayıcılarda kutunun sağ iç yanında, tüm metni silmek için (x) ikonu yer alacaktır.



Input type – URL

- Girdi değeri bir URL olduğu durumlarda kullanılır. Örnek;
- Homepage: <input type="url" name="user_url" />

Input Type – Color:

- Girdi değeri hexadecimal renk kodu olduğu durumlarda kullanılır.
- Opera color picker ile renk seçmenizi sağlarken chrome hexadecimal renk değerlerini kabul etmektedir.
- Örnek; Color: <input type="color" name="user_color" />

Input type – Number

- Sayısal değer girişlerini kolaylaştıran bir eleman oluşturur, min özelliği ile girilebilecek en düşük değer, max özelliği ile girilebilecek en yüksek sayısal değer ayarlanır.
- step özelliği; min, max değerleri arasında bu alana girilebilecek diğer sayıları tanımlamak için bir artış miktarı ayarlar. Elemana girilen değere value özelliği ile programatik olarak ulaşabilirsiniz.

- <!DOCTYPE html>
- <html> <head> <meta charset="utf-8,tr" /> <title>type="number"</title> </head>
- <body> <form action="kayit.aspx" method="post">
- <input type="number" min="10" max="100" step="10" name="num" />
- <input type="submit" />
- </form> </body> </html>





Input type – Range

- Girdi değeri sayı ile verilen bir aralık olduğu durumlarda kullanılır.
 Kaydırma çubuğuna sahip belirli sınırları olan numerik değer seçicisi oluşturur. Browser a göre farklı görüntü alır.
- Örnek;<input type="range" name="points" min="1" max="10" />



Özellik	Değer	Tanım
max	number	Maksimum değer limiti
min	number	Minimum değer limiti
step	number	Tanımlanan aralıkta sayılar artar ve azalır
value	number	Varsayılan değer

Input type – date : Tarih toplayıcıları

- date, month, week, time, datetime, datetime-local Hepsi kendine ait bir sınırlama getirmekte.
- Örnek; Date(Tarih): <input type="date" name="user_date" />
- Input type= "date":Görsel olarak tarih değeri girilebilecek form elemanı oluşturur. (Yıl/Ay/Gün)
- Input type ="month":Görsel olarak ay ve yıl değeri girilebilecek form elemanı oluşturur. (Yıl/Ay)
- Input type ="week":Görsel olarak hafta ve yıl değeri girilebilecek form elemanı oluşturur. (Yıl/Hafta)
- Input type= "time" :Saat ve dakika değeri girilebilecek form elemanı oluşturur.
- Input type= "datetime": Tarih (Yıl/Ay/Gün) ve saat (saat:dakika) değerleri girilebilecek form elemanı oluşturur. (UTC zaman standardına göre).
- Input type= "datetime-local": Girdi değeri kullanıcı bilgisayarındaki Tarih (Yıl/Ay/Gün) ve saat (saat:dakika) değerleri girilebilecek form elemanı oluşturur.

- <form action="DateTest.html">
- <label for="FromDate">Tarih Gir:</label>
- <input type="date" id="FromDate" name="FromDate"</p>
 - value="05/07/2013" /> <hr />
- <input type="submit" id="btnSubmit" name="btnSubmit"</p>
- value="Tarih Kaydet" />
- </form>

Tarih Gir: gg.aa.yyyy 🖨 🔻								
Tarih Kay	 ◄ Temmuz 2013 ▼ ▶ ▶ 							
	Pzt	Sal	Çar	Per	Cum	Cmt	Paz	
	24	25	26	27	28	29	30	
	1	2	3	4	5	6	7	
	8	9	10	11	12	13	14	
	15	16	17	18	19	20	21	
	22	23	24	25	26	27	28	
	29	30	31	1	2	3	4	
	Bugi	ün	Temi	zle				

- <!DOCTYPE html >
- <html > <head> <meta charset=utf-8" /><title>HTML5 ile yeni gelen input elementleri</title> </head>
- ody>
- <form>
- em>type="color":

- <input name="renk" type="color" value="#3300FF"/>
- <hr />
- type="date":

- <input name="tarih" type="date" />
- <hr />
- em>type="email":

- <input name="email" type="email" />
- <hr />
- type="number":

- <input name="sayi" type="number" />
- <hr />
- type="range":

- <input name="aralik" type="range" />
- <hr />
- type="search":

- <input name="arama" type="search" />
- <hr />
- em>type="time":

- <input name="saat" type="time" />
- <hr />
- <input name="Submit" type="submit" />
- <hr />
- </form> </body> </html>

type="color":	
type="date":	
gg.aa.yyyy	
type="email":	
avnibabaoglu@gmail.com	
type="number":	
53	
type="range":	
Ξ	
type="search":	
type="time":	



• HTML5 işlevi fazla olan ve daha fazla form elemanı sunmakta:

Etiket	Tanım
<datalist></datalist>	Girdi değerleri seçenekleri için bir liste oluşturmakta kullanılır.
<keygen/>	Kullanıcı girişleri için doğrulama ve yetkilendirme için kullanılır.
<output></output>	Farklı çıktıları göstermek için kullanılır. Örneğin bir script kullanılarak üretilen çıktı bu eleman ile gösterilir.

• Tarayıcı desteği;

Özellik	IE	Firefox	Opera	Chrome	Safari
datalist	Hayır	4.0	9.5	Hayır	Hayır
keygen	Hayır	4.0	10.5	3.0	Hayır
output	Hayır	Hayır	9.5	10.0	Hayır

• datalist

- Bu element, giriş alanı için ayarlar listesi belirler. Datalist'in içinde option elementi ile liste oluşturur. Datalist'e **id** vererek, listelerdeki veri girişlerinin değerlerini alabilirsiniz; Kullanıcı ilgili alanı doldururken girdiği her harf sonrasında listede yer alan uygun elemanlar metin alanın altında aşağı doğru açılır ve kelimenin tamamını yazmadan seçilerek metnin kutu içine yazılması sağlanır.
 - Webpage:
 - <input type="url" list="url_list" name="link" />
 - <datalist id="url_list">
 - <option label="W3Schools" value="http://www.w3schools.com"/>
 - <option label="Google" value="http://www.google.com" />
 - <option label="Microsoft" value="http://www.microsoft.com" />
 - </datalist>
- İpucu; Option elementi daima bir value taşımalı.

- list özelliği
- Metin alanına girilebilecek önceden tanımlı içerik var ise list özelliği kullanılır.
- Not; <input> types içindeki text, search, url, telephone, email, date pickers, number, range ve color ile beraber çalışır.
- Örnek;
 - <input type="text" id="ilListesi" list="iller" placeholder="Lütfen bir il seçiniz" >
 - <datalist id="iller">
 - <option value="İstanbul">
 - <option value="Ankara">
 - <option value="İzmir">
 - <option value="Eskişehir">
 - <option value="Erzurum">
 - <option value="Hakkari">
 - </datalist>

🕒 list özelliği 🛛 🗙	
← γ ♂ file:///C	:/Users/etanyildizi/Desktop/a.h 🏠 💽 🚳
Web Slice Galerisi Web	Cline College (Stan formalise
Lütfen bir il seçiniz	 list özelliği × + ← Y ♂ file:///C:/Users/etanyildizi/Des Web Slice Galerisi Web Slice Gallery istanbul izmir

• keygen Elementi

- Bu elementin amacı kullanıcıları doğrulamak için güvenli bir yol sunmaktır. Girilen verileri şifreleyerek iletilmesini sağlar. Keygen elementi anahtar grubu üretecidir. Bir form onaylandığında 2 anahtar üretilir; birisi özel birisi de halka açık. Özel anahtar istemcide depolanır ve halka açık anahtar da sunucuya gönderilir. Halka açık anahtar muhtemelen, gelecekte kullanıcıyı doğrulamak için bir istemci sertifikası üretecek.
- Internet Explorer dışındaki tüm tarayıcılar tarafından desteklenmektedir.
- <form>
- Kullanıcı adı: <input type="text" name="kullaniciAdi">
- Şifreleme: <keygen name="security">
- <input type="submit">
- </form>

Kullanıcı adı: Avni	Şifreleme: 2048 (Yükse 2048 (Yükse 1024 (Orta I	ek Düzey) ▼ Gönder <mark>ek Düzey)</mark> Düzey)
		Güvenlik Düzeyi
Sifreleme Sonucu :		

http://kod5.org/html5-formnesneleri-ve-kullanimi-2/

kullanıcıAdı=Avni&security=MIICQDCCASgwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwc

• Output Elementi

- Output elementi, hesaplama veya script çıktısı gibi çıktının farklı türleri için kullanılır.
- Örnek :<output id="result" onforminput="resCalc()"></output>
- Örnek: range değeri ile number değeri toplam sonucunun output gösterimi
- <html lang="tr"> <head> <meta charset=utf-8" /> </head> <body>

 100 ± 50

=130

- <form oninput="x.value=parseInt(a.value)+parseInt(b.value)">
- 0 <input type="range" id="a" value="50">100
- + <input type="number" id="b" value="50">
- = <output name="x" ></output>
- </form> </body> </html>

0

Etiket	Tanım
<fieldset> ve <legend></legend></fieldset>	fieldset: form üzerinde birbirleri ile alakalı etiketler/elementleri gruplandırır. (groups box) legend: fieldset' e başlık tanımlanmasını sağlar.
<select>, <option> ve <optiongroup></optiongroup></option></select>	<select> Açılır liste oluşturulmasını sağlar. <option> ise açılır liste elemanlarını (items) ifade eder. <optgroupt> ise birden çok açılır listenin, tek bir grup içinde gösterilebilmesini mümkün kılar.</option></select>

- <fieldset> ve <legend> Elementi
- <fieldset> elementi; form üzerinde birbirleri ile alakalı girişler(input), etiketler ve diğer kontrollerin gruplandırılmasını sağlar.
- <legend> elementi ise; <fieldset>'in yani etiketler/elementler grubuna başlık tanımlanmasını sağlar.
- <fieldset> <legend> Kullanıcı Girişi</legend>
- Ad:
 <input name="ad" type="text" placeholder="Adınız" />

- Soyad:
 <input name="soyad" type="text" placeholder="Soyadınız" />

- <input name="gonder" type="submit" value="GÖNDER" />

	Kullanıcı Girişi]
	Ad:	
	Adınız	
	Soyad:	
	Soyadınız	
	GÖNDER	

- <select>, <option> ve <optiongroup> Elementi
- <select> elementi açılır liste (drop-down list) oluşturulmasını sağlar. (size >1 büyük olursa liste kutusu (list box) oluşturmayı sağlar.)
- <option> ise açılır liste elemanlarını (items) ifade eder. <select> içinde tanımlanmak üzere her bir eleman için <option> kullanılması gerekmektedir.
- <optgroupt> ise birden çok listenin, tek bir grup içinde gösterilebilmesini mümkün kılar.
- Açılır Liste

Erkan TANYILDIZI
WEB Tasarımı ve Programlama HTML 5.0 eğitimi, Yeni form elamanı
Erkan TANYILDIZI
Yazılım Mühendisliği
Fırat Üniversitesi Teknoloji Fakültesi
İller
Elazığ
İstanbul
Ankara

Liste kutusu

WEB Tasarımı ve Programlama HTML 5.0 eğitimi, Yeni form elamanı	
Erkan TANYILDIZI	
Yazılım Mühendisliği	
Fırat Üniversitesi Teknoloji Fakültesi	
iller	
Elazığ	
İstanbul	
Ankara	
	Ŧ
	WEB Tasarımı ve Programlama HTML 5.0 eğitimi, Yeni form elamanı Erkan TANYILDIZI Yazılım Mühendisliği Fırat Üniversitesi Teknoloji Fakültesi İller Elazığ İstanbul Ankara

• Yeni form özellikleri;

- autocomplete,
- novalidate

• Yeni input özellikleri;

- autocomplete,
- autofocus,
- form,
- form overrides (formaction, formenctype, formmethod, formnovalidate, formtarget),
- height and width,
- list,
- min, max and step,
- multiple,
- pattern (regexp),
- placeholder,
- required,

• Tarayıcı desteği;

Attribute	IE	Firefox	Opera	Chrome	Safari
autocomplete	8.0	3.5	9.5	3.0	4.0
autofocus	Hayır	4.0	10.0	3.0	4.0
form	Hayır	4.0	9.5	10.0	Hayır
form overrides	Hayır	4.0	10.5	10.0	Hayır
height and width	8.0	3.5	9.5	3.0	4.0
list	Hayır	4.0	9.5	Hayır	Hayır
min, max and step	Hayır	Hayır	9.5	3.0	Hayır
multiple	Hayır	3.5	11.0	3.0	4.0
novalidate	Hayır	4.0	11.0	10.0	Hayır
pattern	Hayır	4.0	9.5	3.0	Hayır
placeholder	Hayır	4.0	11.0	3.0	3.0
required	Hayır	4.0	9.5	3.0	Hayır

autocomplete Özelliği

- Form ya da giriş alanının(input field) autocomplete ile otomatik tamamlama özelliğine sahip olmasını sağlar.
 - Not; autocomplete Form etiketi ile çalışır. Ayrıca; input->text, search, url, telephone, email, password, datepickers, range ve color ile de kullanılır.
 - Bu özelliğe sahip olan bir alanda yazmaya başlayan kullanıcı, tarayıcı alandaki verileri doldurabilecek şeylere dair ayarları göstermeye başlayacaktır.

• Örnek

- <form action="demo_form.asp" method="get" autocomplete="on">
- First name: <input type="text" name="fname" />

- Last name: <input type="text" name="lname" />

- E-mail: <input type="email" name="email" autocomplete="off" />
<input type="submit" /> </form>

autofocus Özelliği

- <input type="email" autofocus> Sayfa yüklendiği zaman otomatik olarak bu alan içinin aktif olmasını sağlar .
 - Not; Tüm input elemanları ile çalışır.
- Örnek ;
 - User name: <input type="text" name="user_name" autofocus="autofocus" />

• height ve width özellikleri

- input type girişindeki image için yükseklik ve genişlik atamasını sağlarlar.
 - Not; sadece input type içindeki image'da çalışır.
- Örnek;
 - <input type="image" src="img_submit.gif" width="24" height="24" />

form Özelliği

- Bu özellik input alanlarının kendisinin içinde olsa da olmasa da verilen id ile input alanlarının form elemanına aitmiş gibi davranmasını sağlar.
 - Not; tüm input türleri ile çalışır.
- Örnek;
 - <form action="demo_form.asp" method="get" id="user_form">
 - First name:<input type="text" name="fname" />
 - <input type="submit" /> </form>
 -
 - Last name: <input type="text" name="lname" form="user_form" />

form override özellikleri

- Bazı özellik ayarlarında üst üste bindirme yapmanızı sağlar. Yani form 'a ait özellik değerleri yerine bu değerler kullanılır. Bunun aldığı özellikler ise;
 - formaction form action özelliğinin üstüne biner
 - formenctype form enctype özelliğinin üstüne biner
 - formmethod form method özelliğinin üstüne biner
 - formnovalidate form novalidate özelliğinin üstüne biner
 - formtarget form target özelliğinin üstüne biner
- Örnek;
 - <form action="demo_form.asp" method="get" id="user_form">
 - E-mail: <input type="email" name="userid" />

 - <input type="submit" value="Submit" />

 - <input type="submit" formaction="demo_admin.asp" value="Submit as admin"/>

 - <input type="submit" formnovalidate="true" value="Submit without validation"/>
 </form>
 - Not; override özelliği input girişlerindeki submit ve image ile beraber kullanılır. Ve farklı onay butonları oluşturmada işe yarar.

E-mail:
Submit
Submit as admin
Submit without validation

min, max ve step özellikleri

- Input türlerinden tarih ve sayı girişlerine sınırlamalar getirmek için kullanılırlar.
- Max; girilebilecek maksimum değer,
- Min; izin verilen girilebilecek en düşük değer,
- step özelliği sayı artış ya da azalış aralığını belirler.
 - Not;<input type özelliklerinden tarih toplayıcıları, sayılar ve range ile kullanılır.
- Örnek ;
 - Points: <input type="number" name="points" min="0" max="10" step="3" />
 - Proje <progress min="0" max="100" value="60">60%</progress> tamamlandı.
 - <meter max="10"> 9 araba </meter>

- novalidate
- Bu özellik, onaya basıldığında verinin doğrulanmamasının gerektiğini belirtir. Require alanlar bile kontrol edilmez.
 - Not; <form> ve devamındaki <input> türlerinden text, search, url, telephone, email, password, date pickers, range ve color ile çalışır.
- Örnek;
 - <form action="demo_form.asp" novalidate="novalidate"> E-mail: <input type="email" name="user_email" /> <input type="submit" /> </form>
- pattern özelliği
- Posta Kodu: <input id="zip" name="postaKodu" pattern="[\d]{5}(-[\d]{4})">
- Girilecek olan alanın 5 haneli ve yalnızca rakam olduğunu göstermektedir. Okunaklılık konusunda HTML5'in yapısına biraz aykırı görünse de, programcılar için pattern tanıdık bir özelliktir. Tarayıcı tarafından bu kontrolün yapılması HTML5'in güzel bir özelliğidir.
 - Not; <input> türlerinden text, search, url, telephone, email, ve password ile çalışır.
- Örnek ;
 - Country code: <input type="text" name="country_code" pattern="[A-z]{3}" />
 - <input name="txt" type="text" pattern="[0—9]{4}—[0—9]{3}—[0—9]{2}—[0-9]{2}\D{5,}" />

J 🗋 pa	attern ö	zelliği			+
6-66/İş	telefo	nu] for	matino	la bir m	etin giriniz
6/iş	6	önder			
Johiğ					
n biçime	e uyun.				
	5-66/İş 6/iş n biçime	5-66/İş telefo 36/iş C	5-66/İş telefonu] for 6/iş Gönder n biçime uyun.	5-66/İş telefonu] formatınd 6/iş Gönder n biçime uyun.	5-66/İş telefonu] formatında bir m i6/iş Gönder n biçime uyun.

Placeholder Özelliği

- Veri giriş alanları için yer tutucu bir metin tanımlamaya yarar. Tanımlanan bu metin elemana girilecek veri için bir ipucu içerir. Kullanıcı belirtilen elemana odaklandığında tanımlanan bu değer kaybolur, value özelliği varsayılan bir değer tanımlaması yaparken, placeholder özelliği ise sadece bir ipucu tanımlaması yapar.
- HTML'den önce bu gibi bir kullanım için çeşitli scriptler kullanmak gerekirdi. Tarayıcının bu özelliğe destek vermemesi halinde, metin kutusunun içi boş olarak görünecektir.
 - Not; <input> türlerinden text, search, url, telephone, email, ve password ile çalışır. Alan boşken ipucu gözükür odaklanıldığında kaybolur.
- Örnek;
 - <form id="form1">
 - <input name="txt" type="text" placeholder="Sadece adınızı giriniz !" />

Sadece adınızı giriniz!

Gönder

- <input type="submit" value="Gönder" />
- </form>

required özelliği

- <input type="text" required> Form gönderildiği zaman bu alanın tarayıcı tarafından boş olup olmadığı kontrol edilir. HTML5'den önce bu doğrulama için çeşitli scriptler kullanmak gerekirdi. Not; Bu özellik <input> text, search, url, telephone, email, password, date pickers, number, checkbox, radio ve file ile kullanılır.
- Örnek ;
- <!DOCTYPE html> <html> <head> <meta charset="utf-8,tr" /> <title>required özelliği</title>
- <style type="text/css"> label { display: block; width: 300px; } </style> </head> <body>
- <h3>Kullanıcı Kaydı</h3> <form action="kayit.aspx" method="post"> <label for="email">
- E-mail adresi </label> <input id="email" type="email" required name="email" size="20">
- <label for="text1"> Adı Soyadı </label>
- <input id="text1" type="text" required name="txt" size="20" />
-

- <input type="submit" value="Kullanıcı Oluştur!">
- </form> </body> </html>

Firefox *	🔅 required özelliği
Kullanıcı K	Caydı
E-mail adresi	
tef@firat.edu.tr	
Adı Soyadı	
Lütfen bu al	anı doldurun.

multiple özelliği

- multiple özelliği bir input alanı için seçilebilen çoklu değerler sağlar. multiple özelliğini kullanarak arada virgül/noktalı virgül olmak kaydıyla birden fazla değer seçilebilir.
 - **Not**; input types içindeki; email ve file ile kullanılır.
- Örnek ;
 - <input name="email" type="email" multiple list="liste" />
 - <datalist id="liste">
 - <option value=zeynep@hotmail.com />
 - <option value="harun@hotmail.com" />
 - <option value="yigit@ornek.com" />
 - <option value="elif@ornek.com" />
 - </datalist>

Firefox T () type="e-mail"	+
e-posta adresi	
@hotmail.com;harun@ Gönder	

Etiketler	Tanımlar
	bir yorum
	belge tipi
<a>	bir hiperlink anchor yani çapa ile bağlantı sağlanır sayfalar arasında
<abbr></abbr>	bir kısaltma
<acronym></acronym>	Html5'te desteklenmemekte, yani kaldırılıyor.
<address></address>	bir adres elementi
<applet></applet>	Html5'te desteklenmemekte, yani kaldırılıyor.
<area/>	bir imaj haritasındaki bir alan
<article>Yeni</article>	bir makale yani yazı alanı
<aside>Yeni</aside>	sayfa içeriğinin haricindeki bir içerik
<audio>Yeni</audio>	ses içeriği
	kalın yazı
<base/>	bir sayfadaki tüm bağlantılar için temel url tanımı

Etiketler	Tanımlar
<basefont/>	Html5'te desteklenmemekte, yani kaldırılıyor.
<bdo></bdo>	yazı gösteriminin yönü
<big></big>	Html5'te desteklenmemekte, yani kaldırılıyor.
<blockquote></blockquote>	bir uzun alıntılama
<body></body>	body elementi
	bir tek, satır sonu
<button></button>	bir basılacak buton
<canvas>Yeni</canvas>	grafikler
<caption></caption>	bir tablo başlığı
<center></center>	Html5'te desteklenmemekte, yani kaldırılıyor.
<cite></cite>	minik alıntı
<code></code>	bilgisayar kod yazıları için
<col/>	tablo kolonları yani sütunları için özellik

Etiketler	Tanımlar
<col/>	tablo kolonları yani sütunları için özellik
<colgroup></colgroup>	tablo sütunları grubu
<command/> Yeni	bir emir butonu
<datalist>Yeni</datalist>	bir açılır liste
<dd></dd>	bir tanım açıklaması
	silinmiş yazı
<details>Yeni</details>	bir elementin detayları
<dfn></dfn>	bir tanım terimi
<dir></dir>	Html5'te desteklenmemekte, yani kaldırılıyor.
<div></div>	bir belgedeki bir bölüm
<dl></dl>	bir tanım listesi
<dt></dt>	bir tanım terimi
	vurgulanmış yazı

Etiketler	Tanımlar
<embed/> Yeni	içerik ya da eklenti gömmek için
<fieldset></fieldset>	bir ayarlı alan
<figcaption>Yeni</figcaption>	bir figure elementinin başlığı
<figure>Yeni</figure>	bir grup medya içeriği için ve onların başlıkları için
	Html5'te desteklenmemekte, yani kaldırılıyor.
<footer>Yeni</footer>	bir bölüm veya sayfadaki footer
<form></form>	bir form
<frame/>	Html5'te desteklenmemekte, yani kaldırılıyor.
<frameset></frameset>	Html5'te desteklenmemekte, yani kaldırılıyor.
<h1> to <h6></h6></h1>	1'den 6'ya başlıklar, 6 en küçük vurgulusu 1 en büyük vurgulusu
<head></head>	belgeye dair bilgiler
<header>Yeni</header>	bir bölge veya sayfanın header bölgesi
<hgroup>Yeni</hgroup>	bir belgedeki bölgenin bilgisi

Etiketler	Tanımlar
<hr/>	bir yatay çizgi
<html></html>	bir html belgesi
	yatık yazı
<iframe></iframe>	satır içi alt pencere
	bir resim
<input/>	bir giriş alanı
<ins></ins>	eklenen yazı
<keygen/> Yeni	bir form için üretilmiş anahtar
<kbd></kbd>	klavye yazısı
<label></label>	bir form kontrolü için etiket
<legend></legend>	bir fieldset için başlık
	bir liste elemanı
<link/>	bir referansın kaynağı

Etiketler	Tanımlar
<map></map>	bir imaj haritası
<mark>Yeni</mark>	işaretli yazı
<menu></menu>	bir menü listesi
<meta/>	meta bilgisi
<meter>Yeni</meter>	önceden ayarlanmış aralıktaki ölçü
<nav>Yeni</nav>	navigasyon linkleri
<noframes></noframes>	Html5'te desteklenmemekte, yani kaldırılıyor.
<noscript></noscript>	bir noscript bölgesi
<object></object>	bir gömülü nesne
<0 >	bir sıralı liste
<optgroup></optgroup>	bir seçenek grubu
<option></option>	bir açılır listedeki seçenek
<output>Yeni</output>	çıktıların bazı türlerini tanımlar

Etiketler	Tanımlar
	bir paragraf
<param/>	bir nesne için prametre
<pre></pre>	öntanımlı yazı
<progress>Yeni</progress>	bir görevin sürecine dair
<q></q>	kısa bir alıntı
<rp>Yeni</rp>	ruby kullanımında tarayıcı göstermediğinde gösterilecek yazı
<rt>Yeni</rt>	ruby bilgilendirmesi
<ruby>Yeni</ruby>	ruby ek açıklaması
<s></s>	yazının artık doğru olmadığını gösterir
<samp></samp>	örnek bilgisayar kodu
<script></script>	

Etiketler	Tanımlar
<small></small>	küçük yazı
<source/> Yeni	medya kaynakları
	belgedeki bir bölge
<strike></strike>	Html5'te desteklenmemekte, yani kaldırılıyor.
	vurgulu yazı
<style></style>	

Etiketler	Tanımlar
<tfoot></tfoot>	bir tablo footer'ı
>	bir tablo header'ı
<thead></thead>	bir tablo üst header'ı
<time>Yeni</time>	bir tarih/zaman
<title></title>	belge başlığı
	bir tablo satırı
<tt></tt>	Html5'te desteklenmemekte, yani kaldırılıyor.
<u></u>	Html5'te desteklenmemekte, yani kaldırılıyor.
	indislenmiş yazı
<summary>Yeni</summary>	detail elementinin header'ı
	üstsimge yazı
	bir tablo
	bir tablo gövdesi
	hir tahlo hücresi
HTML Etiketleri ve HTML5 etiketleri

Etiketler	Tanımlar
<tfoot></tfoot>	bir tablo footer'ı
>	bir tablo header'ı
<thead></thead>	bir tablo header'ı
<time>Yeni</time>	bir tarih/zaman
<title></title>	belge başlığı
	bir tablo satırı
<tt></tt>	Html5'te desteklenmemekte, yani kaldırılıyor.
<u></u>	Html5'te desteklenmemekte, yani kaldırılıyor.
	bir sırasız liste
<var></var>	bir değişken
<video>Yeni</video>	bir video
<wbr/> Yeni	bir satır sonu
<xmp></xmp>	Html5'te desteklenmemekte, yani kaldırılıyor.

HTML Özellikleri(attributes) ve HTML5'teki Yeni Özellikler

Özellik	Değer	Tanım
accesskey	character	Bir elemente erişmek için klavye kısayolu tanımlar.
class	classname	Bir element için class adı oluşturur (stil dosyası için).
contenteditable Yeni	true false	Kullanıcının içeriği düzenleyip düzenlememe izni verilip verilin verilin verilin verilimemesi – true, izinli – false, değil
contextmenu Yeni	menu_id	Bir element için kaynak menu oluşturulması ile ilgili.
dir	ltr rtl	Bir elementin içeriğindeki yazının yönünü belirler – ltr, soldan sağa – rtl , sağdan sola
draggable Yeni	true false auto	Kullanıcıya nesneyi sürükleme izni verip vermeme.
dropzone Yeni	copy move link	Sürüklenip bırakılan elementlere ne olacağını belirleme - copy, kopyalar – move, taşır – link, bağ kurar

HTML Özellikleri(attributes) ve HTML5'teki Yeni Özellikler

Özellik	Değer	Tanım
hidden Yeni	hidden	Elementin alakasız olduğunu tanımlar. Ve gösterilmezler.
id	id	Bir element için yegane kimlik belirlenir. class gibi ama bir defa kullanılırlar.
lang	language_c ode	Bir elementin içeriğine ait dil kodu.
spellcheck Yeni	true false	Eğer element için heceleme ve gramer koşulu aranıyorsa kullanılır.
style	style_defini tion	Bir element için satır içi stili belirler.
tabindex	number	Bir elementin tab tuşuna basım sırası, tarayıcı genelde kendisi atar ama tarayıcı işlemediğinde ya da kendimiz atamak istediğimizde kullanırız.
title	text	Bir elemente ait daha fazla bilgi.

- Genel Olay Özellikleri
- HTML 4; tarayıcıda, olaylara tetikleme izni vermiştir, tıpkı bir elemente tıklayınca çalışan Javascript gibi.
- Aşağıdaki genel olay özellikler HTML5 içerisinde olayların etkileşimde bulunmasını sağlamakta.

Window(pencere) Olay Özellikleri : Window nesnesi için olay tetikleyicileri. <body> tag'ına uygulanırlar:

Özellik	Değer	Tanım
onafterprint Yeni	script	Belge basıldıktan sonra çalışır
onbeforeprint Yeni	script	Belge basılmadan önce çalışır
onbeforeonload Yeni	script	Belge yüklenmeden önce çalışır
Onblur	script	Pencere odaklanmasını kaybedince çalışır
onerror Yeni	script	Bir hata gerçekleştiğinde çalışır
onfocus	script	Pencereye odaklandığında çalışır
onhaschange Yeni	script	Belge değiştirildiğinde çalışır
onload	script	Belge yüklendiğinde
onmessage Yeni	script	Mesaj tetiklendiğinde
onoffline Yeni	script	Belge çevrimdışı olduğunda
ononline Yeni	script	Belge çevrimiçi olduğunda
onpagehide Yeni	script	Pencere gizli olduğunda
onpageshow Yeni	script	Pencere görünür olduğunda
onpopstate Yeni	script	Pencerenin tarihi değiştiğinde ya da durumu
onredo Yeni	script	Belge bir tekrar yapma işlemine sahne olduğunda
onresize Yeni	script	Pencere yeniden boyutlandırıldığında
onstorage Yeni	script	Bir belge yüklendiğinde
onundo Yeni	script	Belge bir geri alma işlemine sahne olduğunda
onunload Yeni	script	Kullanıcı belgeden ayrılınca

Form Olayları

• Bir HTML formu içerisinde olay tetiklenir. Tüm HTML5 elementlerine uygulanır, fakat çoğu form elementlerine uygulanır:

Özellik	Değer	Tanım
onblur	script	bir element odaklanmasını yitirdiğinde
onchange	script	bir elment değiştiğinde
oncontextmenu Yen	iscript	bir kaynak menu tetiklendiğinde
onfocus	script	Bir elemente odaklanıldığında
onformchange Yeni	script	Bir form değiştiğinde
onforminput Yeni	script	Bir form kullanıcı veri girişi aldığında
oninput Yeni	script	Bir element veri girişi aldığında
oninvalid Yeni	script	Bir element doğru olmadığında (invalid)
onreset	script	Bir form resetlendiğinde HTML5 desteklemiyor
onselect	script	Bir elment seçildiğinde
onsubmit	script	Bir form onaylandığında

• Klavye Olayları

• Olaylar klavye ile tetiklenir. Tüm HTML5 elementlerine uygulanır.

Özellik	Değer	Tanım
onkeydown	script	Bir tuşa basıldığında
onkeypress	script	Basılı tutulduğunda
onkeyup	script	Tuşa basma serbest bırakıldığında

- Mouse(Fare) Olayları
- Fare ve benzeri kullanıcı davranışları ile olay tetiklenir: Tüm HTML5 etiketlerine uygulanır.

Özellik	Değer	Tanım
onclick	script	Bir mouse tıkında çalışır
ondblclick	script	Mouse çift tıklamasında çalışır
ondrag Yeni	script	Bir element sürüklendiğinde
ondragend Yeni	script	Sürükleme işleminin sonunda
ondragenter Yeni	script	Sürüklenen bir element uygun bir hedef bölgeye bırakıldığında
ondragleave Yeni	script	Bir element hedeflenmiş bırakma bölgesinden ayrıldığında
ondragover Yeni	script	Uygun bırakma noktasının üzerindeyken
ondragstar Yeni	script	Sürükleme işleminin başında
ondrop Yeni	script	Sürüklenen element bırakılmaya başlandığında
onmousedown	script	Bir fae butonu basıldığında
onmousemove	script	Fare işaretçisi hareket ettiğinde
onmouseout	script	Bir elementin üzerinden fare işaretçisi çıkınca
onmouseover	script	Fare işaretçisi bir elmentin üzerindeyken
onmouseup	script	Bir fare butonu bırakıldığında
onmousewheel Yeni	script	Farenin tekeri yani scrollu hareket ettiğinde
onscroll Yeni	script	Bir elementin kaydırma çubuğu hareket ettiğinde

 Media Olayları: Video, imaj ve ses gibi medyalarca tetiklenir. Tüm HTML5 elementlerine uygulanır, fakat çoğu medya için, ses gibi, gömülüler, imaj,nesne, ve video için:

Özellik	Değer	Tanım
onabort	script	Bir olaydan çıkışta çalışır.
oncanplay Yeni	script	Medya çalmaya başladığında çalışır fakat buffer(ön yükleme) için donabilir.
oncanplaythrough Yeni	script	Buffer'dan dolayı donmadan başından sonuna kadar çalınca çalışır.
ondurationchange Yeni	script	Medyanın uzunluğu değiştiğinde çalışır.
onemptied Yeni	script	Aniden bir medya elementinin kaynağı boş olduğunda çalışır. (network hataları, yüklemedeki hatalar gibi)
onended Yeni	script	Medya sona ulaştığında çalışır.
onerror Yeni	script	Bir elementin yüklenmesi sırasında hata gerçekleştiğinde.
onloadeddata Yeni	script	Medya verisi yüklendiğinde çalışır.
onloadedmetadata Yeni	script	Medya süreçleri ve verileri yüklendiğinde çalışır.
onloadstart Yeni	script	Medya verilerini tarayıcı yüklemeye başladığında çalışır.
onpause Yeni	script	Medya verisi durdurulduğunda çalışır.
onplay Yeni	script	Medya verileri çalmaya başladığında çalışır.
onplaying Yeni	script	Medya verileri çalmaya başlayıp devam ettiği süreçte çalışır.

Özellik	Değer	Tanım
onprogress Yeni	script	Tarayıcı medya verilerini çekerken çalışır.
onratechange Yeni	script	Medya verileri oynatma derecesi değiştiğinde çalışır.
onreadystatechange Yeni	script	Hazır olma durumu değiştiğinde çalışır.
onseeked Yeni	script	Bir medya elementinin özelliğini çıkarmaya çalışırken doğru olmaması durumunda durdurması
onseeking Yeni	script	Bir medya elementinin özelliğini çıkarmaya çalışırken doğru olması durumunda ortaya çıkarmaya devam etmesi.
onstalled Yeni	script	Mdya verisini çekerken yüklenme hatasında durdurma eylemini gerçekleştirmesi.
onsuspend Yeni	script	Tarayıcı medya verilerini çekmeye başladığında çalışır fakat daha öncesinde tümü çekilmiştir.
ontimeupdate Yeni	script	Medyanın kendisinin çalma pozisyonu değiştirildiğinde çalışır.
onvolumechange Yeni	script	Medyanın ses ayarı değiştiğinde hatta susturulduğunda çalışır.
onwaiting Yeni	script	Medya çalışırken durdurulduğunda çalışır fakat tekrar çalıştırılmak üzere beklenir.

Web Programlama-C#-ASP.NET

Derleme Ders Notları

Web Programlamaya Giriş

- Web uygulamalarını zenginleştiren birçok programlama dili bulunur. Web sayfalarını geliştirmek için HTML dilinin kullanılması gerekir. Sadece HTML kullanımı statik (sabit) sayfalar geliştirmek için yeterlidir.
- Ancak içeriği kolayca şekillendirmeyi sağlayan CSS dili, istemci taraflı çalışan kodların yazılması için Javascript – VBScript dilleri de Web uygulamalarını zenginleştirir.

Web Programlamaya Giriş

- İstemci ve sunucu tarafında çalışan çeşitli teknolojiler mevcuttur.
- Bir Web sayfası tasarlamak için kullanılan ortak dil HTML dilidir. Ancak HTML sayfaları içinde, sunucu tarafında ve kullanıcı tarafında çalışabilen ayrı web sunarken, CGI, ASP, Php, Perl, ASP.NET gibi programlama dilleri kullanılabilir. Örneğin JavaScript, kullanıcı tarafında kodlama imkanı, web programlama dilleri, sunucu tarafında kod yazma imkanı sunar.

Web Programlamaya Giriş

- Web programlamada temel olarak aşağıdaki konuların bilinmesi gerekmektedir.
 - HTML
 - CSS
 - Script (JavaScript, JQuery VbScript)
 - C#
 - ASP.NET

JavaScript

http://www.belgeler.com/blg/b9r/java-script-ders-notlari

http://www.cizgi-tagem.org/resource/vfiles/tagem/dms_file/3620/javascript_ders_notu.pdf http://www.mynotlar.com/javascript/rehber/operatorler/mantiksal_operatorler.aspx http://www.mynotlar.com/javascript/rehber/operatorler/karsilastirma_operatorleri.aspx http://www.serdarsert.com/Makaleler/javascript-nesne-olusturmak.aspx http://www.banadersanlat.com/index.php/java-script-dersleri/785-javascript-nesneler-.html http://bid.ankara.edu.tr/yardim/www/javasturk/javascr7.html http://www.bilgisayardershanesi.com/bilgisayar_dersleri/java-script-form-nesnesi.html http://www.mynotlar.com/javascript/rehber/cerezler.aspx http://mfyz.com/dokuman/87/temel-javascript-4---zamanlayicilarla-calismak

- JavaScript dili, Netscape firması tarafından oluşturulmuştur.
- Yazım biçimi olarak C dili esas alınarak tasarlanmıştır. Amaç olarak HTML'in yetmediği yerlere script'ler ile destek vermesi düşünülmüştür.
- Web programcılığına dinamik bir yapı kazandıran JavaScript, istemci tarafında çalışır. Kullanımı giderek yaygınlaşan JavaScript, daha sonra Microsoft firmasının Internet Explorer web tarayıcısında da kullanılabilir hale geldi. Günümüzde tüm tarayıcıların desteklediği bir script dilidir.

- JavaScript kodları yazmak için Notepad gibi bir metin editörü yeterlidir. Kodlar
- <script>...</script> etiketleri arasında yazılmalıdır.
- Bu kod alanı içinde yorum satırları için // ve /* ... */ ifadeleri kullanılabilir.
- Sadece bir satır yorum satırı yapılacaksa // ifadesi kullanılır.
- // bu satır yorum satırıdır.
- Birden fazla satır yorum satırı yapılacaksa, satırların başladığı yere /*, bittiği yere */ ifadeleri yerleştirilir.
- /* yoruma alınan 1. satır
- 2. satır
 - ... */

- JavaScript kodları HTML sayfaları içine <head> etiketlerine gömülü olarak veya .js uzantılı dosyalara referans gönderilerek HTML içinden çağrılabilir.
 - <head></head> etiketi arasına yazılan JS çağrıldığı zaman çalışır.
 - <body></body> etiketi arasına yazılan JS sayfa yüklenince çalışır.
- Her iki etiket arasına istediğiniz kadar kod öbeği yazabilirsiniz fakat harici .js dosyalarını kullanmanız HTML kodlamada sayfanıza daha sade bir görünüm kazandıracak ve hata yapma ihtimalinizi daha aza indirgeyecektir. Bu harici dosyaları da <head> yada <body> arasında kullanabilirsiniz.

```
Örnek;
  <html> <head>
   <script src='harici.js'></script>----\rightarrow harici .js dosyası
   <script type='text/javascript'> veya
   <script language="javascript">
  //kodlar
   </script>
   </head>
   <body>
   <script src='harici.js'></script>----→ harici .js dosyası
   <script type='text/javascript'> veya
   <script language="javascript">
  //kodlar
   </script>
   </body> <html>
```

- JavaScript dilinde nesneler, nesnelere uygulanan olaylar ve olaylara ilişkin görevler vardır. Bir nesneyi tıklamak, üzerine gelmek, üzerinde dolaşmak gibi işlemler, sayfa ile kullanıcının etkileşimli olarak çalışmasını sağlar.
- JavaScript, aynı bir programlama dilinde olduğu gibi değişkenlere, klavyeden bilgi alma, ekrana çıktı verme işlemlerine, koşul ve döngü yapılarına, fonksiyon, nesne ve olay kavramlarına sahiptir.

- <html> <head><title>onClick</title>
- <script language="javascript">
- function merhaba() { alert ("beni tikladiniz"); }
- </script> </head> <body>
- <input type="button" name="tikla" value="tikla"</p>
- onClick=merhaba()>
- </body> </html>
- script etiketinin language attribute (dil özelliği) değeri ile kullanılacak script dili belirtilir.
- Javascript kullanılması için burada language="javascript" bildirimi yapılır.

🐔 onClick	- Microsof	t Internet	Explore	۲.
Dosya Düa	en Görün	üm Sik Kul	lanılanlar	Araçlar
Geri -	•	× 2	6	🔎 Ara 🦄
Adres 餐 C:	\Documents	and Settings	:\etanyildiz	:i.FIRAT\De
tikla	Microsoft	t Internet	Explore	r 🔀
	♪	beni tikladin	iz	
		Tamam		

- <html> <head> <title>JavaScript Örneği</title> </head>
- <body>
 Bu yazı html ile yazıldı.

- <script language="JavaScript">
- document.write("İşte bu ise JavaScript ile yazıldı!")
- </script>
-
 dr> Bu yazı yine HTML ile yazıldı.
- </body> </html>
- Buradaki script ifadesi head etiketleri arasında bir fonksiyon olarak değil, body etiketleri arasında satır halinde kullanılmıştır.

🕘 JavaScript Örneği - Microsoft Inte
Dosya Düzen Görünüm Sık Kullanılanlar
Ġ Geri 🝷 🕥 👻 📓 🏠
Adres 🗿 C:\Documents and Settings\etanyild
Bu yazı html ile yazıldı. İşte bu ise JavaScript ile yazıldı! Bu yazı yine HTML ile yazıldı.

- Örnek <HEAD> Etiketleri Arasına ve <BODY> Etiketleri İçinde Farklı Bölgelere Javascript Kodu Yazma
- <head>
 <script language="javascript" type="text/javascript" >
 var a=5;
 var b=10;
 var adi="harun";
 </script> </head>
- ody>

<script language="javascript" type="text/javascript" > document.write(**a**+**b**+"
"); document.write(**adi**); </script> </body>

Değişkenler

- var anahtar sözcüğü ile yeni bir değişken oluşturulur. Tür bilgisi saklanmaz.
- Sayısal değerler verildiğinde işlem yapma yeteneğine sahip olurlar.
- Çift veya tek tırnak içerisinde değer verildiğinde ise metin ifadesi olarak anlaşılır.
- Dikkat edilmesi gereken nokta değişkenlerin küçük büyük harf duyarlı olmasıdır.

- Değişkenler
- var deger1;
- var deger2=20;
- var deger3=30;
- var ay="Kasım";
- var yil='2012';
- var degerToplam=deger2+deger3;
- var tarih=ay+yil;
- Satırın sonunda sonlandırma karakteri olarak ; kullanılır.
- degerToplam isimli değişkende 20 ve 30 değerleri toplanarak elde edilen 50 değeri tutulurken, tarih isimli değişkende, ay ve yil değişkenlerinden gelen metin ifadeleri birleştirilir ve "Kasım2012" değeri oluşturulur.

Operatörler

- JavaScript operatörleri, Visual Basic .NET dilinde kullanılan operatörlerden biraz farklıdır. Örneğin mod almak için Mod anahtar sözcüğü yerine % mod alma operatörü kullanılır.
- Atama Operatörü (=)
- Değişkenlere değer atamak için = karekteri kullanılır.
- Aritmetik Operatörler
- Değişkenler üzerinde aritmetik işlemler yapmak için tanımlanmış operatörlerdir.

Operatör Açıklama

- + Sayısal değişkenleri toplar. String değişkenlerini birbirine ekler.
- Sayısal değişkenlerde çıkarma işlemi yapar.
- * Sayısal değişkenlerde çarpma işlemi yapar.
- / Sayısal değişkenlerde bölme işlemi yapar.
- % Sayısal değişkenlerde mod alma işlemini yapar.
- ++ Sayısal değişkenlerde artma işlemini yapar.
- -- Sayısal değişkenlerde azalma işlemini yapar.

- Karşılaştırma Operatörleri
- Operatör Açıklama
- == Eşit midir? operatörü. İki değer de birbirine eşit ise true sonucu verir.
- != Eşit değil midir? operatörü. İki değer birbirine eşit değilse true sonucunu verir.
- Küçük operatörü. Sol taraf değeri, sağ taraf değerinden küçükse true sonucunu verir.
- Büyüktür operatörü. Sol taraf değeri, sağ taraf değerinden büyük ise true sonucunu verir.
- <= Küçük eşittir operatörü.</p>
- >= Büyük eşittir operatörü.

- Karşılaştırma Operatörleri
- İki değerin eşitliğinin karşılaştırılması için == operatörü kullanılır.
- if (a == b) {
- document.write("a ile b değişkeni eşit")
- }
- İki değerin eşitsizliğinin karşılaştırılması için != operatörü kullanılır.
- if (a != b) {
- document.write("a ile b değişkeni eşit değildir")
- }

- Mantıksal Operatörler
- Operatör Açıklama
- && And (ve) operatörü. İki tarafta belirtilen ifadeler true ise, sonuç olarak true değerini döndürür.
- Or (veya) operatörü. İki tarafta verilen ifadelerden en az birinin doğru olması durumunda true değerini döndürür.
- I Not operatörü: Koşulun yanlış olması durumunda true değerini verir.

Koşul Operatörü

- [koşul ifadesi] ? koşul_doğru_ise : koşul_yanlış_ise
- Değişken tanımlarken aritmetik, karşılaştırma ve mantıksal operatörler kullanılabilir. Bunlara ek olarak C dilinden gelen koşul operatörleri kullanılabilir. Bir if deyiminin tek satırda yazılmış haline benzeyen bu operatörün genel kullanım biçimi ise
- [koşul ifadesi] ? koşul_doğru_ise : koşul_yanlış_ise
- şeklindedir.

Koşul Operatörünün Kullanımı

- Bu kodda, (a + b < c) ifadesi ile elde edilen sonuca göre, e değerine d veya a + b değerleri atanır.
- a + b işleminin sonucu olan 12 değeri, c değerinden küçük olduğu için ifade doğru olarak sonuçlanır. Bu durumda d değeri, e değişkenine atanır ve e değişkeni 23 değerini taşır.
- Bu koşulu if deyimi ile yazılabilir.
- if (a + b < c)
- e = d;
- else
- e = a+ b;

Koşul Operatörünün Kullanımı

var a=5; var b=7; var c=14; var d=23; var e; e = (a + b < c) ? d : a+b ;

Dizi yapısı

- Dizi, Array veya [] tanımlanır. Javascripte dizi boyutu dinamiktir. Dizinin boyutunu belirledikten sonra eğer isterseniz eleman girerek boyutunu değiştirebilirsiniz.
 - var bilgi=new Array(boyut);
 - var bilgi= new Array("bilgisayar",3,4, true); //1.yöntem var bilgi=["bilgisayar", 3, 4, true]; //2. yöntem Örnek:
 - <script language="javascript" ttype="text/javascript">
 var deger=new Array(3); deger[0]="a"; deger[1]=5;
 deger[2]=true; document.write(deger.length+"
");//sonuç: 3
 deger[3]=false; document.write(deger.length); // sonuç: 4
 </script>

- Dizi içerisindeki elemanlara ulaşmak için [] operatörü kullanılır.
 Bu operatör içerisine indeks numarası yazılarak erişilir.
 - var veri=new Array("javascript",true,false,4,3.4); document.write(veri[3]);//4

• Örnek:

- <script language="javascript" type="text/javascript">
- var veri=new Array(1,2,3,4,5);
- document.write("Dizinin Boyutu="+veri.length+"
");
- veri[5]="efe";
- document.write("Dizinin Boyutu="+veri.length);
- </script>
Çok Boyutlu Diziler

- Javascript ile çok boyutlu diziler oluşturabilirsiniz. Aslında çok boyutlu diziden kasıt bir dizi elemanı olarak başka bir dizi değişkeninin adının atanmasıdır. Yani bir dizinin başka biz dizi içerisinde eleman olarak bulunmasıdır.
- Kullanımı:
- var a=new Array("css", "ders", 78, 1, new Array(true, false));

Örnek

 <html> <head> <title>Sıralama</title> <script language="JavaScript"> function sirala() {

Sırala

```
var kelime = new Array(9)
```

```
var i, j, k, m, ara
for (i=0; i <=9; i++) {
```

ara =kelime[k]

```
101 (1=0; 1 <=9; 1++) {
```

```
kelime[i] = prompt("Metinsel bir ifade girin lütfen...","Buraya,")
```

```
//Sayıları sıralıyoruz.
for (j=0; j <=9; j++) {
for (k=0; k <=8; k++) {
if (kelime[k] > kelime[k+1]) {
```

kelime[k] = kelime[k+1]

kelime[k+1] = ara } }

ifade girin lütfen","Buraya	ı,") }
Metinsel bir ifade girin lütfen	1. kelime = 0 2. kelime = 2
Buraya, Tamam Vazgeç	3. kelime = 4
	4. kelime = 4
	5. kelime = 5
	6. kelime = 7
	7. kelime = 7
	8. kelime = 7
	9. kelime = 8
	10. kelime = 9

Düğmeye basarak 10 tane yazıyı alfabetik sıraya sokabilirsiniz.

```
for (m=0; m <= 9; m++) document.write(m + 1 + ". kelime = " + kelime[m] + "<br>")
} </script> </head><body> Düğmeye basarak 10 tane yazıyı alfabetik sıraya
sokabilirsiniz </br> <input type="button" name="tikla" value="Sırala"
onClick=sirala()> </body></html>
```

- push(): Bu metot dizinin en sonuna değer eklemek için kullanılır. Diziye birden fazla veya tek tek değer eklenebilir.
 - diziadi.push("deger1","deger2");
 - Örnek
 - <script language="javascript" type="text/javascript" >
 - var veri=new Array(1,2,3,4,5);
 - veri.push("zeynep");veri.push("emir");
 - veri.push ("ali",6);
 - document.write(veri);
 - </script>
 - Çıktı:1,2,3,4,5,zeynep,emir,ali,6

- unshift(): Bu metot dizinin en başına değer eklemek için kullanılır. Diziye birden fazla veya tek tek değer eklenebilir.
 - diziadi.unshift("değer1", "değer2");
 - Örnek
 - <script language="javascript" type="text/javascript" >
 - var veri=new Array(1,2,3,4,5);
 - veri.unshift("zeynep");
 - veri.unshift("emir"); veri.unshift("ali",6);
 - document.write("veri);
 - </script>
 - Çıktı:ali,6, emir,zeynep,1,2,3,4,5

- splice(): Dizi eleman içerisine eleman eklemek ve çıkartmak için kullanılan metottur.
 - Elaman Ekleme:
 - diziadi.splice (parametre1, **0**, "değer1", "değer2");
 - parametre1: Değer atamaya hangi indeks numarasından başlanacağı belirtilir. (başlangıç indisi)
 - O: Diziden eleman çıkarmak için buraya 0 dışında bir sayı eklemeniz gerekecektir. Fakat 0 eklerseniz hiçbir eleman çıkarılmayacaktır.

Elaman Çıkarma:

- diziadi.splice (parametre1, parametre2);
 - parametre1: Dizi değişkeni içerisinde çıkarma işlemine başlanacak elemanın indeks numarasıdır.
 - parametre2: Çıkarılacak eleman sayısıdır.

- splice()
 - Aynı Anda Elaman Ekleme ve Çıkarma(Silme):
 - diziadi.splice(parametre1, parametre2, "değer1", "değer2");
 - parametre1: Dizi değişkeni içerisinde çıkarma işlemine başlanacak elemanın indeks numarasıdır.
 - parametre2: Dizi değişkeni içerisinde indeksten itibaren kaç tane eleman çıkarılacağıdır. değer1, değer2... Çıkarma işleminden sonra parametre1 ile belirlenen indeks numarasına göre veriler eklenecektir.
 - var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits.splice(2,1,"Lemon","Kiwi");
 - Banana,Orange,Lemon,Kiwi,Mango

splice()

- <script language="javascript" type="text/javascript">
- var myarray=new Array(2,1,"c#","asp.net");
- var my_dizi=myarray.splice(1,3); for(var i=0; i<my_dizi.length; i++)
- { document.write(my_dizi[i]); document.write("
"); }
 </script>
- //myarray dizisinden çıkarılan elamanlar my_dizi içine eklendi
- sonuç:
- 1

c#

asp.net

- concat(): Birden fazla dizi değişkenini tek bir dizi içerisinde toplamaya yarayan metottur.
 - <html><head>
 - <script type="text/javascript">
 - function aile(){
 - var h = ["Mehmet", "Saniye"]; var s = ["Rabia", "Seda", "Yağmur"];
 - var k = ["Resul"]; var c = s.concat(k); var f = h.concat(s,k);
 - document.write("Aile: "+f+"
");
 - document.write("Çocuklar: "+c+"
");}
 - </script></head>
 - Aile: Mehmet,Saniye,Rabia,Seda,Yağmur,Resul
 <body>
 Çocuklar: Rabia,Seda,Yağmur,Resul
 - <input type="button" value="Göster" onClick="aile()"/>

Çıktı:

</body> </html>

- pop(): Dizi içindeki son elemanın alınma işlemi Array.pop() metodu ile sağlanmaktadır. Bu metod, herhangi bir parametre almadan, dizi içindeki son elemanı silip diziden çıkartarak geri döndürür.
 - <input type="button" value="Array.pop Uygula" onClick="sonSil()" />
 - <input type="button" value="Örneği Sıfırla" onClick="ornekSifirla()" /> <div id="ornekCerceve"> </div>
 - <script type="text/javascript">
 - var ornekArray; var ornekKap;
 - function diziGoster(alinan)
 - { var sonuc = "<hr/>Alınan : " + alinan + "
>";
 - var sonHal = "Kalan : " + ornekArray.join(" + ") + "
>";
 - ornekKap.innerHTML = ornekKap.innerHTML + sonuc + sonHal; }
 - function sonSil() { silinen = ornekArray.pop(); diziGoster(silinen); }
 - function ornekSifirla()
 - { ornekArray = ["eins", "zwei", "drei", "vier"];
 - ornekKap = document.getElementById("ornekCerceve");
 - ornekKap.innerHTML = ornekArray.join(" + ") + "
>";
 - }
 - ornekSifirla(); // Başlama noktası
 - </script>

Array.pop Uygula	Örneği Sıfırla
eins + zwei + drei +	· vier
Alınan : vier Kalan : eins + zwei	+ drei

- shift(): pop fonksiyonun tersidir. Bu metod, herhangi bir parametre almadan, dizi içindeki ilk elemanı silip diziden çıkartarak geri döndürür.
 - <script type="text/javascript">
 - var myArray = ['bir', 'iki', 'üç', 'dört'];
 - document.write(myArray + "
>");
 - document.write(myArray.shift() + "
>");
 - document.write(myArray + "
>");
 - </script>

bir,iki,üç,dört	
bir	
iki,üç,dört	

- delete oparatörü: Bir nesneyi, nesne özelliğini ya da bir dizi içerisinde sırası belirtilmiş bir öğeyi silmek için kullanılır.
- delete nesneAdı delete nesneAdı.özellik delete nesneAdı[öğe-indeksi] delete özellik
- uyeler = new Array("sahin", "axion", "man", "sessiztetik");
 delete uyeler[2]; // "man" öğesi diziden kaldırılır.
- Not: Delete operatörüyle kaldırılan dizi öğeleri, dizi içinde yer kaplamazlar.

- sort (): Array dizisinin elemanlarını sözlük dizilimine göre sıralar. dizi.sort();
- reverse(): Array dizisinin elemanlarını ters çevirir. dizi.reverse();
- indexof(): string içinde bir kelime veya karakter aramasını baştan yaparak bulunduğu konumun öğrenilmesini sağlar. Bulduğunda indis numarasını, bulamadığında -1 değerini döndürür.
 - var sonuc = katar.indexOf(aranacak);
- IastIndexOf (): indexof()'un tersidir. Aramaya sondan başlar
 - lastIndexOf(aranacak, [basKonum])
 - document.write(myString.lastIndexOf("m") + "
>");
 - document.write(myString.lastIndexOf("m",4) + "
>");

Koşul ve Döngü Yapıları

- Programlamanın akışını yönlendiren koşul yapıları ve döngülerdir. Döngüler birden fazla gerçekleştirilecek işlemlerin blok halinde yazılmasını sağlar.
- if koşul ifadesinin genel yapısı
 - if (koşul)
 - // koşul doğru ise çalışacak ifade
 - // koşul yanlış ise akışın devam edeceği alan

```
if (koşul) { //koşul doğru ise }
else { //koşul yanlış ise }
```

• Örnek:

}

 var Bugun = 'Bugünü bilen bir değişken'; if (Bugun == 'Pazartesi') { document.write('Bugün haftanın ilk günüdür.') document.location = 'Pazartesi.asp'; } else if (Bugun == 'Salı') { document.write('Bugün haftanın ikinci günüdür.') document.location = 'Sali.asp'; } else {

document.write('Bugün haftanın diğer günlerinden biridir.')
document.location = 'nereye gitmek isteniyorsa...';

- Koşul ve Döngü Yapıları
- for döngü yapısı
- Tekrarlanan belirli bir işlemi yaptırmak için kullanılan döngülerin JavaScript dilindeki kullanımı tamamen C dilinin yapısına göre tasarlanmıştır.
- for döngüsünün genel kullanım biçimi;
- for(başlangıç_değeri; döngü_ifadesi; değişecek_değişken_adı)

• •

- //yapılacak işlemler
- }
- şeklindedir.

- Koşul ve Döngü Yapıları
- Örnek:For Döngüsünün Kullanımı
- var a;
- var b = 10;
- for (a = 1; a <= b; a++) {</pre>
- document.write(a , ". sayı", "
"); }

- Koşul ve Döngü Yapıları
- for .. in döngü yapısı
- for in ifadesi dizi elemanlarının yinelemeleri (iteration) için kullanılır.
- Bu kod dizi içindeki elemanları teker teker alıp sırasıyla yazdıracaktır.

```
var Dizi = new Array();
Dizi[0]= 'Deg1';
Dizi[1]= 'Deg2';
Dizi[2]= 'Deg3';
for (var Yaz in Dizi) {
    document.write([Yaz] + '<br>');
```

- Koşul ve Döngü Yapıları
- while-do..while döngüsünün yapısı
- while (döngü_koşul_ifadesi)
- { //şart doğruysa yapılacak işlemler }
- //şart doğru değilse yapılacak işlemler
- •
- do
- { //şart doğruysa yapılacak işlemler } while(döngü_koşul_ifadesi)
- //şart doğru değilse yapılacak işlemler

Koşul ve Döngü Yapıları

- Döngüleri durdurmak ve devam ettirmek için ayrıca iki ifade vardır.
- break
- Belirli bir şart vuku bulduğunda break döngüyü durdurur.
- for(i = 1; i < 5; i++) {
- if(i == 3) break;
- document.write(i);
- }

Koşul ve Döngü Yapıları

Continue

}

* break'ın durdurduğu döngüyü devam ettirir.

-Bu kodda herhangi bir iş yapmaz, çünkü duran bir döngü yoktur.

```
for (i = 1; i < 5; i++) {
```

```
if(i == 3) continue;
```

```
document.write(i);
```

- Koşul ve Döngü Yapıları
- Örnek: while döngüsünün yapısı

```
    var i = 1;
while (i < 5) {
document.write(i + '<br>');
i = i + 1; //i'yi bir artırarak devam et, i++ şeklinde de
yazılabilir
}
```

- Koşul ve Döngü Yapıları
- switch-case yapısı:
- Genel kullanımı:
- switch (parametre) {
- case "ifade1":
- // ifade1 koşulu doğru ise yapılması istenenler
- break; //break ile diğer koşulların da çalışması
- //engellenir ve döngüden çıkılır.
- case "ifade2:"
- //ifade2 koşulu doğru ise yapılması istenenler
- break; }

- Koşul ve Döngü Yapıları
- Örnek:
- var Deg = 5; switch (Deg) {
 - case 5:

document.write('Değerimiz 5') break;

case 6:

document.write('Değerimiz 6') break;

case 0:

document.write('Değerimiz 0') break;

default:

document.write('Deg değişkenin değeri değiştikçe betiğin yazacağı değerde değişir') }

- Klavyeden Bilgi Alma ve Ekrana Çıktı Verme
- JavaScript dilinde kullanıcıdan bilgi almak için formların dışında promt komutu kullanılır. promt komutu ile kullanıcıdan bilgi alırken ayrı bir pencere açılır.
- prompt("soru", "cevap için rehber ifade");
- Örnek:Prompt ile kullanıcıdan değer almak
 - var sehir;
 - sehir=prompt("Yaşadığınız yer",
 - "İstanbul için 34, Ankara için 6 gibi");

ires 🖉 C:\Documents and Settings\etanyildizi.FIRAT\Desktop\jsornek.html		
Explorer Kullamcı Komut İstemi		
Kod Komut İstemi: Yaşadığınız yer	Tamam İptal	
İstanbul için 34, Ankara için 6 gibi		

🔁 🎧 🔎 Ara 🥂 Sik Kullanilanlar

- Klavyeden Bilgi Alma ve Ekrana Çıktı Verme
- JavaScript dilinde HTML sayfasına yazı yazdırmak için write komutu kullanılır.
- document.write ("Yazılmak istenen değişkene ilişkin açıklama", degisken);
- Görüldüğü gibi write komutu document fonksiyonuyla birlikte kullanılır.

Popup Kutuları

- prompt() : İstem Kutusu prompt('İstem iletisi.', 'Varsayılan değer');
- alert() : Uyarı Kutusu
- alert('Uyarı iletisi.');
 veya
- var Deg = 'Uyarı iletisi.';
- alert(Deg);



(üyelik işlemlerinde boş bırakılan bir yer olduğunda uyarı verilebilir)

Popup Kutuları

 confirm() : Onay Kutusu confirm('Onay iletisi.'); veya var Deg = 'Onay iletisi.'; confirm(Deg);



- Seçilen mesaj(lar)ı silmek istediğinizden emin misiniz?" Onay iletisidir. Geriye true veya false değeri döner.
- var cevap confirm('Veriyi silmek istiyor musunuz?')

- <html> <head>
- <script type="text/javascript">
- <!-- function dogrulama()</pre>
- var cevap = confirm("Sayfadan cikmak istiyormusun?")
- if (cevap)
- { alert("Güle Güle!")
- window.location = "http://www.google.com/";
- }
- else { alert("Teşekkürler kaldığınız için") }
- }
- //-->
- </script>
- </head> <body> <form> <input type="button" onclick="dogrulama()" value="Sayfadan ayrıl"> </form> </body> </html>

Fonksiyonlar

- JavaScript dilinde, kodların yeniden kullanılabilmesi için kullanılır. Genel kullanımı:
- •
- function fonksiyon_ismi(parametre1, parametre2)
- {
- //yapılacak işlemler
- }
- •
- Fonksiyon içinde hesaplanan değer, return ifadesi ile geri döndürülür.

- Örnek: JavaScript ile Toplama
- function topla(deger1, deger2)
- {

• }

- var sonuc= deger1+deger2;
- return sonuc;

• Örnek:

```
    //1. ve en yaygın olan yazım şekli

  function Islev1() {
   var Deg = "Islev1'in değişkeni";
   alert(Deg);
  //2. yazım
  Islev1 = function () {
   var Deg = "Islev1'in değişkeni";
   alert(Deg);
  //çağırma
  <input type="button" value="İşlevi çağır" onclick="Islev1()">
```

 Fonksiyonlar(işlevler) durağan komutlar olduğundan bazen sayfa yüklenmesiyle veya bunun gibi herhangi bir müdahele olmadan çalışmasını istediğimiz durumlarda aşağıdaki yazımları kullanırız.

```
    //1. yazım
        window.onload = function() {
            var Deg = "Islev1'in değişkeni";
            alert(Deg);
        }
        //2. yazım
        function Islev1() {
            var Deg = "Islev1'in değişkeni";
            alert(Deg);
        window.onload = Islev1(); //Islev1 şeklinde de yazılabilir
```

 İşlevlerimiz bağımsız değişkenlere (argument) de tabi olarak çalıştırılabilir ve bir işlev birden fazla bağımsız değişken alabilir.

```
    function Islev(Ileti)
```

```
{
alert(lleti);
}
```

<input type="button" value="1. ileti!" onclick="Islev('1. (değişken) ileti!')">

<input type="button" value="2. ileti!" onclick="Islev('2. (değişken) ileti!')">

* İşlevler sayısal sonuçlar elde etmek amacıyla da kullanılabilir.

```
function Carp(Deg1, Deg2) {
  Sonuc = Deg1 * Deg2;
  return Sonuc;
}
```

<input type="button" value="Çarp" onclick="alert(Carp(2, 2))">

Hazır Fonksiyonlar

- eval (): Bir argümanı değerlendiren veya çalıştıran (Evaluate/Execute) fonksiyondur.
- var x = 10; var y = 20; var a = eval("x*y") + "
"; var b = eval("2+2") + "
"; var c = eval("x+17") + "
"; var res = a + b + c;
- var toplamTutar = "((siparisMiktari * fiyatTL) * vergi)"
- if (vergiYok) {

}

- toplamTutar = "(siparisMiktarı * fiyatTL)"
- •
- Programın herhangi bir yerinde, eval() fonksiyonunu kullanarak aynı kodları yazmadan değişkenlerin değerlerini bulmaya yardımcı olacaktır.
- document.write(eval(toplamTutar))
- eval içeriği tırnaksız kullanılırsa değişkenlerde tırnak olacak

Hazır Fonksiyonlar

- isNaN (): Gelen veri ya da herhangi bir değişkenin sadece rakamlardan oluştuğunu anlamak için kullanılır. Rakamlardan oluşuyor ise false, değilse true.
 - var a = '123';
 isNaN(a); ==> false
 - var a = 'a123';
 isNaN(a); ==> true
- parsInt ("string"), ParseFloat("string"): string veri tiplerini tamsayı ve ondalık sayı tipine dönüştürmek için kullanılır.
 - document.write(parseFloat("3.14"))
Olaylar

- Bir Web sayfası üzerinde kullanıcının her türlü hareketi kontrol edilebilir. Bir kontrolün üzerine gelmesi, dolaşması ve üzerinden ayrılması gibi hareklere olay denir.
- Bu olaylar ise onClick, onMuoseOver, onMouseOut, onSubmit,onReset, onChange, onLoad, onUnLoad, onError, onAbort, onFocus, onBlur olarak belirtilebilir.

onClick

- Internet sitelerinin çoğunda en sık kullanılan JavaScript olayıdır. Sayfa üzerinde bir nesnenin fare ile tıklanıp bırakılması sonucunda gerçekleşen olaydır. Link, button ve resim nesneleri tıklanarak onClick olayı tetiklenebilir.
- Nesnelerin etiketlerinde ise onClick olaylarını tetikleyen fonksiyonların ismi bildirilmelidir.
- Ayrıca onDblClick, çift tıklanma olayını tetikler.

- Örnek: onClick Olayı
- function tikla()
- {
- alert("Tıklama işlemi gerçekleşti...");
- }
- <input type="button" name="tikla" value="tikla" onClick=tikla()>

onMouseOver, onMouseOut

- Fare nesnenin üzerindeyken **onMouseOver**, fare nesne üzerinden ayrılınca **onMouseOut** olayları devreye girer.
- function nesneUzerinde() {
- window.status="Şu anda nesne üzerindesiniz."; }
- function nesneDisinda() {
- window.status="nesnenin dışına çıktınız."; }
- <a href="http://www.google.com"</p>
- onMouseOver = nesneUzerinde()
- onMouseOut = nesneDisinda()> Google



onSubmit

- Web safyalarında ziyaretçinin forma bilgi girip sunucuya göndermesi durumlarında onSubmit devreye girer.
 Gönderilecek forma girilen verilerin uygunluğunun kontrolü bu olayın tetiklediği fonksiyonlara yaptırılabilir.
- function dogrula()
- { confirm ('Formu doldurduysanız OK'i tıklayınız');
- }
- <form action="mail.pl" method="post" onSubmit="dogrula()">
- confirm komutu, kullanıcıya Ok ve Cancel butonlarından oluşan bir diyalog penceresi açar.

onReset

- Form içinde kullanılan tüm metin alanlarının temizlenmesini sağlar. Doldurulan formda yanlışlık olduğunda bu olay tetiklenir. Kullanıcıya onay penceresi çıkartmak için de kullanılabilir.
- function sil()
- { return confirm('Silmek istediginize emin misiniz?'); }
- <form onReset="sil()">
- <input type="text" name="mail">
- <input type="reset" value="sil">
- </form>

onChange

- Bilgi girişi yapılan alanlarda, değişikliğin gerçekleştiği bilgisi onChange olayı ile tetiklenir.
 - <html> <head> <title>onChange</title>
 - <script language="JavaScript1.2">
 - function yaz(secnesnesi){
 - return secnesnesi.options[secnesnesi.selectedIndex].value}
 - </script> </head> <body> <form name="form1">
 - <select name="mesaj" size="3" onchange="alert(yaz(this.form.mesaj))">
 - <option name="mesaj" value="Bu">mesaj1
 - <option name="mesaj" value="program">mesaj2
 - <option name="mesaj" value="onChange komutu ile yapılmıştır">mesaj3
 - </select> </form> </body> </html>



onLoad, onUnLoad

- onLoad olayı sayfaya giriş yapıldığında gerçekleşir.
 onUnLoad olayı sayfadan çıkıldığında gerçekleşir.
- function giris()
- { alert("Sayfaya Giriş Yaptınız!"); }
- function cikis()
- { alert("Sayfadan çıktınız.."); }
- <body onLoad="giris()" onUnload="cikis()">
- </body>

onError, onAbort

- Ziyaret edilen sayfadaki nesneler, çeşitli nedenlerden dolayı tam olarak yüklenememiş olabilir. Genellikle resim nesnelerinin yüklenmesinde problem çıkabilir. Bu tür durumları ziyaretçiye bildirmek için onError veya onAbort olayları kullanılır.
- <img src="resim.gif" onError="alert('Resim dosyası</p>
- yüklenemedi')">

onFocus, onBlur

- onFocus olayı kullanıcı kontrollerine giriş yapılırken gerçekleşir. OnBlur olayı ise ve kullanıcı kontrollerinden çıkış yapılırken gerçekleşir.
- function dogru()
- { document.form1.mesaj.value="Lütfen hata yapmayın!"; }
- function sor()
- { document.form1.mesaj.value="isminiz alındı"; }
- <form name="form1" method="post">
- <h3>Lütfen isminizi yazınız!</h3>
- <input type="text" size="20" name="isim"</p>
- onfocus="dogru()" onblur="sor()">
- <input type="text" name="mesaj"> </form>

JavaScript - Hata Denetimi

try { // Hata oluşabilecek kodlar}
 catch (err) { // Hata oluştuğunda çalışacak kodlar }
 finally { alert(" bu bloktaki kodlar her durumda çalışacaktır");}

 try komut bloğunda ister hata oluşsun ister oluşmasın finally bloğundaki kodlar çalışacaktır. Bu anahtar sözcüğünün kullanılması isteğe bağlıdır. Bu anahtar sözcük kullanılırsa mutlaka catch bloğundan sonra yazılmalıdır.

throw

- Bu anahtar sözcüğünü kullanarak try bloğu içinde kendimiz özel istisnai durumları oluşturabilirsiniz. Bu anahtar sözcük try blokları içerisinde kullanılmalıdır.
 - Örnek:

<script type="text/javascript"> try

- { for var i=0; i<10; i++)</p>
- { if("i==4"); { throw ("i=4 hatası"); alert("Hata sonrası kodlar"); } }
- } catch(err) { alert(err) }
 </script>
- Throw içerisine yazdığımız ifade catch (err) bloğundaki err parametresine aktarılır.

- Tarayıcı Nesneleri-Window Nesnesi
- Genel pencere özelliklerini tutan nesnedir. Pencere açma ve kapama işlemleri için bu nesne kullanılır.
- Genel kullanımı:
- window.open(" url ","pencere_ismi","pencere_ozellikleri");
- window.close();
- Open komutu ile yeni bir pencere açılırken, close komutu ile pencere kapatılır.
- Yeni bir pencere açmak için open komutuna ilk parametrenin girilmesi zorunludur. Pencere_ismi, birden fazla pencere ile işlem yapıldığı durumlarda kullanılabilir.

• Window Nesnesi

Pencereye ait özellikler ;

 Özellik 	Varsayılan Değer	Açıklama
• width	auto	Açılan pencerenin pixel genişliğini belirtir.
• height	auto	Açılan pencerenin pixel yüksekliğini belirtir.
• left	auto	Ekranın sol noktasına ile pencere arasındaki uzaklığı verir.
• top	auto	Ekranın üst noktasına ile pencere arasındaki uzaklığı verir.
• menubar	no	Menu çubuğunun görüntülenmesini sağlar.
• Toolbar	no	Araç çubuğun görüntülenmesini sağlar.
Location	no	Adres çubuğunun görüntülenmesini sağlar.
Status	no	Durum çubuğunun görüntülenmesini sağlar.
Scrollbars	no	Kaydırma çubuklarının görüntülenmesini sağlar
Resizable	no	Penceresinin boyutlandırılmasını sağlar.

• Window Nesnesi

- Örnek: Yeni bir pencere açma
- window.open('http://www.bilgeadam.com',
- 'bilgeadam', 'menubar=no, toolbar=no, scrollbars=yes, location=yes,width=300, heigt=300')
- veya
- Click here for specific popup window



- Window Nesnesi
- history
- go(number|url):
- Internet tarayıcısı ile daha önce ziyaret edilmiş sayfalara tekrar ulaşabilmek için window.history.go(-1) komutu kullanılabilir. -1 ifadesi ile bir önceki sayfaya gidilir. Sayı arttırılarak daha önceki sayfalara da gidilebilir. Pozitif değerler kullanılarak ileri sayfalara gidilebilir.
- Bir geri git
 İki geri git
- İki ileri git

• Window Nesnesi

 back() : geçmiş belgeyi döndürür (1 geri gider), geçmişte bir belge bulamaz ise null sonucunu döndürür yani hiç bir sayfayı döndürmez.

Geri git

- forward() : gelecek belgeyi döndürür (1 ileri gider), gelecekte bir belge bulamaz ise null sonucunu döndürür yani hiç bir sayfayı döndürmez.
- İleri git
- Internet tarayıcısının en alt kısmında bulunan status penceresine erişmek için window.status komutu kullanılır.
- window.status ="JavaScript öğreniyoruz!";

 Javascriptler başlı başına konu olup verilen özellikler dışında daha birçok özelliği vardır. Burada sadece temel bilgi verilmiştir.



- VbScript, Microsoft tarafından geliştirilmiştir. Ancak, VbScript tüm tarayıcılar tarafından desteklenmez. Bu nedenle tüm tarayıcılarda çalışacak script yazılmak isteniyorsa, JavaScript kullanılmalıdır.
- Vbscript dilinin kullanılabilmesi için script etiketi içerisindeki language özniteliğine vbscript değeri atanır.
- <script language="vbscript">...</script>

- İlk VbScript Örneği
- html>
- ody>
- <script language="vbscript">
- document.write("İlk VBScript!")
- </script>
- </body>
- </html>



- Değişken tanımlamaları dim anahtar sözcüğü ile yapılır.
- html>
- ody>
- <script language="vbscript">
- dim isim
- isim="Bilge Adam"
- document.write(isim)
- </script>
- </body>
- </html>

- Metot tanımlaması Visual Basic.NET diline benzer. Geriye sonuç döndürmeyen metotlar Sub, geriye sonuç döndüren metotlar ise function anahtar sözcüğü ile tanımlanır.
- Bu tanımlanmış metotlar "MetotAdı()" anahtar sözcüğü ile çağrılabilirler.

Function Tanımlaması

- <html> <head>
- <script language="vbscript">
- function sehir()
- sehir = "Trabzon"
- end function
- </script> </head> <body>
- <script language="vbscript">
- document.write("En sevdiğim sehir: " & sehir()) </script>
- function procedure geri dönüş değerine sahiptir.
- </body> </html>



http://jquery.com/ Çavuş Musa, "Kendinize Güvenerek JQuery ",Kasım(2010),Almanya http://www.cemdemir.net/jquery/jquery-nedir-nasil-kullanilir-nasil-ogrenilir-465.html <u>http://www.phpkodlari.com/javascript-dersleri/jquery-nedir/</u> <u>http://www.eburhan.com/jquery-dunyasina-adim-atiyoruz/</u> <u>http://www.codekodu.com/jquery.php</u>

Web Programlama-C#-ASP.NET

Derleme Ders Notları

Web Programlamaya Giriş

- Web uygulamalarını zenginleştiren birçok programlama dili bulunur. Web sayfalarını geliştirmek için HTML dilinin kullanılması gerekir. Sadece HTML kullanımı statik (sabit) sayfalar geliştirmek için yeterlidir.
- Ancak içeriği kolayca şekillendirmeyi sağlayan CSS dili, istemci taraflı çalışan kodların yazılması için Javascript – VBScript dilleri de Web uygulamalarını zenginleştirir.

Web Programlamaya Giriş

- İstemci ve sunucu tarafında çalışan çeşitli teknolojiler mevcuttur.
- Bir Web sayfası tasarlamak için kullanılan ortak dil HTML dilidir. Ancak HTML sayfaları içinde, sunucu tarafında ve kullanıcı tarafında çalışabilen ayrı web sunarken, CGI, ASP, Php, Perl, ASP.NET gibi programlama dilleri kullanılabilir. Örneğin JavaScript, kullanıcı tarafında kodlama imkanı, web programlama dilleri, sunucu tarafında kod yazma imkanı sunar.

Web Programlamaya Giriş

- Web programlamada temel olarak aşağıdaki konuların bilinmesi gerekmektedir.
 - HTML
 - CSS
 - Script (JavaScript, JQuery VbScript)
 - C#
 - ASP.NET

JavaScript

http://www.belgeler.com/blg/b9r/java-script-ders-notlari

http://www.cizgi-tagem.org/resource/vfiles/tagem/dms_file/3620/javascript_ders_notu.pdf http://www.mynotlar.com/javascript/rehber/operatorler/mantiksal_operatorler.aspx http://www.mynotlar.com/javascript/rehber/operatorler/karsilastirma_operatorleri.aspx http://www.serdarsert.com/Makaleler/javascript-nesne-olusturmak.aspx http://www.banadersanlat.com/index.php/java-script-dersleri/785-javascript-nesneler-.html http://bid.ankara.edu.tr/yardim/www/javasturk/javascr7.html http://www.bilgisayardershanesi.com/bilgisayar_dersleri/java-script-form-nesnesi.html http://www.mynotlar.com/javascript/rehber/cerezler.aspx http://mfyz.com/dokuman/87/temel-javascript-4---zamanlayicilarla-calismak

- JavaScript dili, Netscape firması tarafından oluşturulmuştur.
- Yazım biçimi olarak C dili esas alınarak tasarlanmıştır. Amaç olarak HTML'in yetmediği yerlere script'ler ile destek vermesi düşünülmüştür.
- Web programcılığına dinamik bir yapı kazandıran JavaScript, istemci tarafında çalışır. Kullanımı giderek yaygınlaşan JavaScript, daha sonra Microsoft firmasının Internet Explorer web tarayıcısında da kullanılabilir hale geldi. Günümüzde tüm tarayıcıların desteklediği bir script dilidir.

- JavaScript kodları yazmak için Notepad gibi bir metin editörü yeterlidir. Kodlar
- <script>...</script> etiketleri arasında yazılmalıdır.
- Bu kod alanı içinde yorum satırları için // ve /* ... */ ifadeleri kullanılabilir.
- Sadece bir satır yorum satırı yapılacaksa // ifadesi kullanılır.
- // bu satır yorum satırıdır.
- Birden fazla satır yorum satırı yapılacaksa, satırların başladığı yere /*, bittiği yere */ ifadeleri yerleştirilir.
- /* yoruma alınan 1. satır
- 2. satır
 - ... */

- JavaScript kodları HTML sayfaları içine <head> etiketlerine gömülü olarak veya .js uzantılı dosyalara referans gönderilerek HTML içinden çağrılabilir.
 - <head></head> etiketi arasına yazılan JS çağrıldığı zaman çalışır.
 - <body></body> etiketi arasına yazılan JS sayfa yüklenince çalışır.
- Her iki etiket arasına istediğiniz kadar kod öbeği yazabilirsiniz fakat harici .js dosyalarını kullanmanız HTML kodlamada sayfanıza daha sade bir görünüm kazandıracak ve hata yapma ihtimalinizi daha aza indirgeyecektir. Bu harici dosyaları da <head> yada <body> arasında kullanabilirsiniz.

```
Örnek;
  <html> <head>
   <script src='harici.js'></script>----\rightarrow harici .js dosyası
   <script type='text/javascript'> veya
   <script language="javascript">
  //kodlar
   </script>
   </head>
   <body>
   <script src='harici.js'></script>----→ harici .js dosyası
   <script type='text/javascript'> veya
   <script language="javascript">
  //kodlar
   </script>
   </body> <html>
```

- JavaScript dilinde nesneler, nesnelere uygulanan olaylar ve olaylara ilişkin görevler vardır. Bir nesneyi tıklamak, üzerine gelmek, üzerinde dolaşmak gibi işlemler, sayfa ile kullanıcının etkileşimli olarak çalışmasını sağlar.
- JavaScript, aynı bir programlama dilinde olduğu gibi değişkenlere, klavyeden bilgi alma, ekrana çıktı verme işlemlerine, koşul ve döngü yapılarına, fonksiyon, nesne ve olay kavramlarına sahiptir.

Örnek JavaScript

- <html> <head><title>onClick</title>
- <script language="javascript">
- function merhaba() { alert ("beni tikladiniz"); }
- </script> </head> <body>
- <input type="button" name="tikla" value="tikla"</p>
- onClick=merhaba()>
- </body> </html>
- script etiketinin language attribute (dil özelliği) değeri ile kullanılacak script dili belirtilir.
- Javascript kullanılması için burada language="javascript" bildirimi yapılır.
Örnek JavaScript

🐔 onClick	- Microsof	t Internet	Explore	۲.
Dosya Düa	en Görün	üm Sik Kul	lanılanlar	Araçlar
Geri -	•	× 2	6	🔎 Ara 🦄
Adres 餐 C:	\Documents	and Settings	:\etanyildiz	:i.FIRAT\De
tikla	Microsoft	t Internet	Explore	r 🔀
	♪	beni tikladin	iz	
		Tamam		

Örnek JavaScript

- <html> <head> <title>JavaScript Örneği</title> </head>
- <body>
 Bu yazı html ile yazıldı.

- <script language="JavaScript">
- document.write("İşte bu ise JavaScript ile yazıldı!")
- </script>
-
 dr> Bu yazı yine HTML ile yazıldı.
- </body> </html>
- Buradaki script ifadesi head etiketleri arasında bir fonksiyon olarak değil, body etiketleri arasında satır halinde kullanılmıştır.

Örnek JavaScript

🕘 JavaScript Örneği - Microsoft Inte
Dosya Düzen Görünüm Sık Kullanılanlar
Ġ Geri 🝷 🕥 👻 📓 🏠
Adres 🗿 C:\Documents and Settings\etanyild
Bu yazı html ile yazıldı. İşte bu ise JavaScript ile yazıldı! Bu yazı yine HTML ile yazıldı.

- Örnek <HEAD> Etiketleri Arasına ve <BODY> Etiketleri İçinde Farklı Bölgelere Javascript Kodu Yazma
- <head>
 <script language="javascript" type="text/javascript" >
 var a=5;
 var b=10;
 var adi="harun";
 </script> </head>
- ody>

<script language="javascript" type="text/javascript" > document.write(**a**+**b**+"
"); document.write(**adi**); </script> </body>

Değişkenler

- var anahtar sözcüğü ile yeni bir değişken oluşturulur. Tür bilgisi saklanmaz.
- Sayısal değerler verildiğinde işlem yapma yeteneğine sahip olurlar.
- Çift veya tek tırnak içerisinde değer verildiğinde ise metin ifadesi olarak anlaşılır.
- Dikkat edilmesi gereken nokta değişkenlerin küçük büyük harf duyarlı olmasıdır.

- Değişkenler
- var deger1;
- var deger2=20;
- var deger3=30;
- var ay="Kasım";
- var yil='2012';
- var degerToplam=deger2+deger3;
- var tarih=ay+yil;
- Satırın sonunda sonlandırma karakteri olarak ; kullanılır.
- degerToplam isimli değişkende 20 ve 30 değerleri toplanarak elde edilen 50 değeri tutulurken, tarih isimli değişkende, ay ve yil değişkenlerinden gelen metin ifadeleri birleştirilir ve "Kasım2012" değeri oluşturulur.

Operatörler

- JavaScript operatörleri, Visual Basic .NET dilinde kullanılan operatörlerden biraz farklıdır. Örneğin mod almak için Mod anahtar sözcüğü yerine % mod alma operatörü kullanılır.
- Atama Operatörü (=)
- Değişkenlere değer atamak için = karekteri kullanılır.
- Aritmetik Operatörler
- Değişkenler üzerinde aritmetik işlemler yapmak için tanımlanmış operatörlerdir.

Operatör Açıklama

- + Sayısal değişkenleri toplar. String değişkenlerini birbirine ekler.
- Sayısal değişkenlerde çıkarma işlemi yapar.
- * Sayısal değişkenlerde çarpma işlemi yapar.
- / Sayısal değişkenlerde bölme işlemi yapar.
- % Sayısal değişkenlerde mod alma işlemini yapar.
- ++ Sayısal değişkenlerde artma işlemini yapar.
- -- Sayısal değişkenlerde azalma işlemini yapar.

- Karşılaştırma Operatörleri
- Operatör Açıklama
- == Eşit midir? operatörü. İki değer de birbirine eşit ise true sonucu verir.
- != Eşit değil midir? operatörü. İki değer birbirine eşit değilse true sonucunu verir.
- Küçük operatörü. Sol taraf değeri, sağ taraf değerinden küçükse true sonucunu verir.
- Büyüktür operatörü. Sol taraf değeri, sağ taraf değerinden büyük ise true sonucunu verir.
- <= Küçük eşittir operatörü.</p>
- >= Büyük eşittir operatörü.

- Karşılaştırma Operatörleri
- İki değerin eşitliğinin karşılaştırılması için == operatörü kullanılır.
- if (a == b) {
- document.write("a ile b değişkeni eşit")
- }
- İki değerin eşitsizliğinin karşılaştırılması için != operatörü kullanılır.
- if (a != b) {
- document.write("a ile b değişkeni eşit değildir")
- }

- Mantıksal Operatörler
- Operatör Açıklama
- && And (ve) operatörü. İki tarafta belirtilen ifadeler true ise, sonuç olarak true değerini döndürür.
- Or (veya) operatörü. İki tarafta verilen ifadelerden en az birinin doğru olması durumunda true değerini döndürür.
- I Not operatörü: Koşulun yanlış olması durumunda true değerini verir.

Koşul Operatörü

- [koşul ifadesi] ? koşul_doğru_ise : koşul_yanlış_ise
- Değişken tanımlarken aritmetik, karşılaştırma ve mantıksal operatörler kullanılabilir. Bunlara ek olarak C dilinden gelen koşul operatörleri kullanılabilir. Bir if deyiminin tek satırda yazılmış haline benzeyen bu operatörün genel kullanım biçimi ise
- [koşul ifadesi] ? koşul_doğru_ise : koşul_yanlış_ise
- şeklindedir.

Koşul Operatörünün Kullanımı

- Bu kodda, (a + b < c) ifadesi ile elde edilen sonuca göre, e değerine d veya a + b değerleri atanır.
- a + b işleminin sonucu olan 12 değeri, c değerinden küçük olduğu için ifade doğru olarak sonuçlanır. Bu durumda d değeri, e değişkenine atanır ve e değişkeni 23 değerini taşır.
- Bu koşulu if deyimi ile yazılabilir.
- if (a + b < c)
- e = d;
- else
- e = a+ b;

Koşul Operatörünün Kullanımı

var a=5; var b=7; var c=14; var d=23; var e; e = (a + b < c) ? d : a+b ;

Dizi yapısı

- Dizi, Array veya [] tanımlanır. Javascripte dizi boyutu dinamiktir. Dizinin boyutunu belirledikten sonra eğer isterseniz eleman girerek boyutunu değiştirebilirsiniz.
 - var bilgi=new Array(boyut);
 - var bilgi= new Array("bilgisayar",3,4, true); //1.yöntem var bilgi=["bilgisayar", 3, 4, true]; //2. yöntem Örnek:
 - <script language="javascript" ttype="text/javascript">
 var deger=new Array(3); deger[0]="a"; deger[1]=5;
 deger[2]=true; document.write(deger.length+"
");//sonuç: 3
 deger[3]=false; document.write(deger.length); // sonuç: 4
 </script>

- Dizi içerisindeki elemanlara ulaşmak için [] operatörü kullanılır.
 Bu operatör içerisine indeks numarası yazılarak erişilir.
 - var veri=new Array("javascript",true,false,4,3.4); document.write(veri[3]);//4

• Örnek:

- <script language="javascript" type="text/javascript">
- var veri=new Array(1,2,3,4,5);
- document.write("Dizinin Boyutu="+veri.length+"
");
- veri[5]="efe";
- document.write("Dizinin Boyutu="+veri.length);
- </script>

Çok Boyutlu Diziler

- Javascript ile çok boyutlu diziler oluşturabilirsiniz. Aslında çok boyutlu diziden kasıt bir dizi elemanı olarak başka bir dizi değişkeninin adının atanmasıdır. Yani bir dizinin başka biz dizi içerisinde eleman olarak bulunmasıdır.
- Kullanımı:
- var a=new Array("css", "ders", 78, 1, new Array(true, false));

Örnek

 <html> <head> <title>Sıralama</title> <script language="JavaScript"> function sirala() {

Sırala

```
var kelime = new Array(9)
```

```
var i, j, k, m, ara
for (i=0; i <=9; i++) {
```

```
IOT (I=0; I <=9; I++) {
```

```
kelime[i] = prompt("Metinsel bir ifade girin lütfen...","Buraya,") }
```

```
//Sayıları sıralıyoruz.
for (j=0; j <=9; j++) {
  for (k=0; k <=8; k++) {
    if (kelime[k] > kelime[k+1]) {
      ara =kelime[k]
```

kelime[k] = kelime[k+1]

kelime[k+1] = ara } }

Metinsel bir ifac	le girin lütfen	
uraya,		
10		
Tamam	Vazgeç	

Düğmeye basarak 10 tane yazıyı alfabetik sıraya sokabilirsiniz.

1.	kelime =	0
2.	kelime =	2
3.	kelime =	4
4.	kelime =	4
5.	kelime =	5
6.	kelime =	7
7.	kelime =	7
8.	kelime =	7
9.	kelime =	8

for (m=0; m <= 9; m++) document.write(m + 1 + ". kelime = " + kelime[m] + "
")
} </script> </head><body> Düğmeye basarak 10 tane yazıyı alfabetik sıraya
sokabilirsiniz <wbr> <input type="button" name="tikla" value="Sırala"
onClick=sirala()> </body></html>

- push(): Bu metot dizinin en sonuna değer eklemek için kullanılır. Diziye birden fazla veya tek tek değer eklenebilir.
 - diziadi.push("deger1","deger2");
 - Örnek
 - <script language="javascript" type="text/javascript" >
 - var veri=new Array(1,2,3,4,5);
 - veri.push("zeynep");veri.push("emir");
 - veri.push ("ali",6);
 - document.write(veri);
 - </script>
 - Çıktı:1,2,3,4,5,zeynep,emir,ali,6

- unshift(): Bu metot dizinin en başına değer eklemek için kullanılır. Diziye birden fazla veya tek tek değer eklenebilir.
 - diziadi.unshift("değer1", "değer2");
 - Örnek
 - <script language="javascript" type="text/javascript" >
 - var veri=new Array(1,2,3,4,5);
 - veri.unshift("zeynep");
 - veri.unshift("emir"); veri.unshift("ali",6);
 - document.write("veri);
 - </script>
 - Çıktı:ali,6, emir,zeynep,1,2,3,4,5

- splice(): Dizi eleman içerisine eleman eklemek ve çıkartmak için kullanılan metottur.
 - Elaman Ekleme:
 - diziadi.splice (parametre1, **0**, "değer1", "değer2");
 - parametre1: Değer atamaya hangi indeks numarasından başlanacağı belirtilir. (başlangıç indisi)
 - O: Diziden eleman çıkarmak için buraya 0 dışında bir sayı eklemeniz gerekecektir. Fakat 0 eklerseniz hiçbir eleman çıkarılmayacaktır.

Elaman Çıkarma:

- diziadi.splice (parametre1, parametre2);
 - parametre1: Dizi değişkeni içerisinde çıkarma işlemine başlanacak elemanın indeks numarasıdır.
 - parametre2: Çıkarılacak eleman sayısıdır.

- splice()
 - Aynı Anda Elaman Ekleme ve Çıkarma(Silme):
 - diziadi.splice(parametre1, parametre2, "değer1", "değer2");
 - parametre1: Dizi değişkeni içerisinde çıkarma işlemine başlanacak elemanın indeks numarasıdır.
 - parametre2: Dizi değişkeni içerisinde indeksten itibaren kaç tane eleman çıkarılacağıdır. değer1, değer2... Çıkarma işleminden sonra parametre1 ile belirlenen indeks numarasına göre veriler eklenecektir.
 - var fruits = ["Banana", "Orange", "Apple", "Mango"]; fruits.splice(2,1,"Lemon","Kiwi");
 - Banana,Orange,Lemon,Kiwi,Mango

splice()

- <script language="javascript" type="text/javascript">
- var myarray=new Array(2,1,"c#","asp.net");
- var my_dizi=myarray.splice(1,3); for(var i=0; i<my_dizi.length; i++)
- { document.write(my_dizi[i]); document.write("
"); }
 </script>
- //myarray dizisinden çıkarılan elamanlar my_dizi içine eklendi
- sonuç:
- 1

c#

asp.net

- concat(): Birden fazla dizi değişkenini tek bir dizi içerisinde toplamaya yarayan metottur.
 - <html><head>
 - <script type="text/javascript">
 - function aile(){
 - var h = ["Mehmet", "Saniye"]; var s = ["Rabia", "Seda", "Yağmur"];
 - var k = ["Resul"]; var c = s.concat(k); var f = h.concat(s,k);
 - document.write("Aile: "+f+"
");
 - document.write("Çocuklar: "+c+"
");}
 - </script></head>
 - Aile: Mehmet,Saniye,Rabia,Seda,Yağmur,Resul
 <body>
 Çocuklar: Rabia,Seda,Yağmur,Resul
 - <input type="button" value="Göster" onClick="aile()"/>

Çıktı:

</body> </html>

- pop(): Dizi içindeki son elemanın alınma işlemi Array.pop() metodu ile sağlanmaktadır. Bu metod, herhangi bir parametre almadan, dizi içindeki son elemanı silip diziden çıkartarak geri döndürür.
 - <input type="button" value="Array.pop Uygula" onClick="sonSil()" />
 - <input type="button" value="Örneği Sıfırla" onClick="ornekSifirla()" /> <div id="ornekCerceve"> </div>
 - <script type="text/javascript">
 - var ornekArray; var ornekKap;
 - function diziGoster(alinan)
 - { var sonuc = "<hr/>Alınan : " + alinan + "
>";
 - var sonHal = "Kalan : " + ornekArray.join(" + ") + "
>";
 - ornekKap.innerHTML = ornekKap.innerHTML + sonuc + sonHal; }
 - function sonSil() { silinen = ornekArray.pop(); diziGoster(silinen); }
 - function ornekSifirla()
 - { ornekArray = ["eins", "zwei", "drei", "vier"];
 - ornekKap = document.getElementById("ornekCerceve");
 - ornekKap.innerHTML = ornekArray.join(" + ") + "
>";
 - }
 - ornekSifirla(); // Başlama noktası
 - </script>

Array.pop Uygula	Örneği Sıfırla
eins + zwei + drei +	· vier
Alınan : vier Kalan : eins + zwei	+ drei

- shift(): pop fonksiyonun tersidir. Bu metod, herhangi bir parametre almadan, dizi içindeki ilk elemanı silip diziden çıkartarak geri döndürür.
 - <script type="text/javascript">
 - var myArray = ['bir', 'iki', 'üç', 'dört'];
 - document.write(myArray + "
>");
 - document.write(myArray.shift() + "
>");
 - document.write(myArray + "
>");
 - </script>

bir,iki,üç,dört	
bir	
iki,üç,dört	

- delete oparatörü: Bir nesneyi, nesne özelliğini ya da bir dizi içerisinde sırası belirtilmiş bir öğeyi silmek için kullanılır.
- delete nesneAdı delete nesneAdı.özellik delete nesneAdı[öğe-indeksi] delete özellik
- uyeler = new Array("sahin", "axion", "man", "sessiztetik");
 delete uyeler[2]; // "man" öğesi diziden kaldırılır.
- Not: Delete operatörüyle kaldırılan dizi öğeleri, dizi içinde yer kaplamazlar.

- sort (): Array dizisinin elemanlarını sözlük dizilimine göre sıralar. dizi.sort();
- reverse(): Array dizisinin elemanlarını ters çevirir. dizi.reverse();
- indexof(): string içinde bir kelime veya karakter aramasını baştan yaparak bulunduğu konumun öğrenilmesini sağlar. Bulduğunda indis numarasını, bulamadığında -1 değerini döndürür.
 - var sonuc = katar.indexOf(aranacak);
- IastIndexOf (): indexof()'un tersidir. Aramaya sondan başlar
 - lastIndexOf(aranacak, [basKonum])
 - document.write(myString.lastIndexOf("m") + "
>");
 - document.write(myString.lastIndexOf("m",4) + "
>");

Koşul ve Döngü Yapıları

- Programlamanın akışını yönlendiren koşul yapıları ve döngülerdir. Döngüler birden fazla gerçekleştirilecek işlemlerin blok halinde yazılmasını sağlar.
- if koşul ifadesinin genel yapısı
 - if (koşul)
 - // koşul doğru ise çalışacak ifade
 - // koşul yanlış ise akışın devam edeceği alan

```
if (koşul) { //koşul doğru ise }
else { //koşul yanlış ise }
```

• Örnek:

}

 var Bugun = 'Bugünü bilen bir değişken'; if (Bugun == 'Pazartesi') { document.write('Bugün haftanın ilk günüdür.') document.location = 'Pazartesi.asp'; } else if (Bugun == 'Salı') { document.write('Bugün haftanın ikinci günüdür.') document.location = 'Sali.asp'; } else {

document.write('Bugün haftanın diğer günlerinden biridir.')
document.location = 'nereye gitmek isteniyorsa...';

- Koşul ve Döngü Yapıları
- for döngü yapısı
- Tekrarlanan belirli bir işlemi yaptırmak için kullanılan döngülerin JavaScript dilindeki kullanımı tamamen C dilinin yapısına göre tasarlanmıştır.
- for döngüsünün genel kullanım biçimi;
- for(başlangıç_değeri; döngü_ifadesi; değişecek_değişken_adı)

• •

- //yapılacak işlemler
- }
- şeklindedir.

- Koşul ve Döngü Yapıları
- Örnek:For Döngüsünün Kullanımı
- var a;
- var b = 10;
- for (a = 1; a <= b; a++) {</pre>
- document.write(a , ". sayı", "
"); }

- Koşul ve Döngü Yapıları
- for .. in döngü yapısı
- for in ifadesi dizi elemanlarının yinelemeleri (iteration) için kullanılır.
- Bu kod dizi içindeki elemanları teker teker alıp sırasıyla yazdıracaktır.

```
var Dizi = new Array();
Dizi[0]= 'Deg1';
Dizi[1]= 'Deg2';
Dizi[2]= 'Deg3';
for (var Yaz in Dizi) {
    document.write([Yaz] + '<br>');
```

- Koşul ve Döngü Yapıları
- while-do..while döngüsünün yapısı
- while (döngü_koşul_ifadesi)
- { //şart doğruysa yapılacak işlemler }
- //şart doğru değilse yapılacak işlemler
- •
- do
- { //şart doğruysa yapılacak işlemler } while(döngü_koşul_ifadesi)
- //şart doğru değilse yapılacak işlemler

Koşul ve Döngü Yapıları

- Döngüleri durdurmak ve devam ettirmek için ayrıca iki ifade vardır.
- break
- Belirli bir şart vuku bulduğunda break döngüyü durdurur.
- for(i = 1; i < 5; i++) {
- if(i == 3) break;
- document.write(i);
- }

Koşul ve Döngü Yapıları

Continue

}

* break'ın durdurduğu döngüyü devam ettirir.

-Bu kodda herhangi bir iş yapmaz, çünkü duran bir döngü yoktur.

```
for (i = 1; i < 5; i++) {
```

```
if(i == 3) continue;
```

```
document.write(i);
```

- Koşul ve Döngü Yapıları
- Örnek: while döngüsünün yapısı

```
    var i = 1;
while (i < 5) {
document.write(i + '<br>');
i = i + 1; //i'yi bir artırarak devam et, i++ şeklinde de
yazılabilir
}
```
- Koşul ve Döngü Yapıları
- switch-case yapısı:
- Genel kullanımı:
- switch (parametre) {
- case "ifade1":
- // ifade1 koşulu doğru ise yapılması istenenler
- break; //break ile diğer koşulların da çalışması
- //engellenir ve döngüden çıkılır.
- case "ifade2:"
- //ifade2 koşulu doğru ise yapılması istenenler
- break; }

- Koşul ve Döngü Yapıları
- Örnek:
- var Deg = 5; switch (Deg) {
 - case 5:

document.write('Değerimiz 5') break;

case 6:

document.write('Değerimiz 6') break;

case 0:

document.write('Değerimiz 0') break;

default:

document.write('Deg değişkenin değeri değiştikçe betiğin yazacağı değerde değişir') }

- Klavyeden Bilgi Alma ve Ekrana Çıktı Verme
- JavaScript dilinde kullanıcıdan bilgi almak için formların dışında promt komutu kullanılır. promt komutu ile kullanıcıdan bilgi alırken ayrı bir pencere açılır.
- prompt("soru", "cevap için rehber ifade");
- Örnek:Prompt ile kullanıcıdan değer almak
 - var sehir;
 - sehir=prompt("Yaşadığınız yer",
 - "İstanbul için 34, Ankara için 6 gibi");

dres 🙋 C:\Documents and Settings\etanyildizi.FIRAT\Desktop\jsornek.html		
Explorer Kullamcı Komut İstemi		
Kod Komut İstemi: Yaşadığınız yer	Tamam İptal	
İstanbul için 34, Ankara için 6 gibi		

🔁 🎧 🔎 Ara 🥂 Sik Kullanilanlar

- Klavyeden Bilgi Alma ve Ekrana Çıktı Verme
- JavaScript dilinde HTML sayfasına yazı yazdırmak için write komutu kullanılır.
- document.write ("Yazılmak istenen değişkene ilişkin açıklama", degisken);
- Görüldüğü gibi write komutu document fonksiyonuyla birlikte kullanılır.

Popup Kutuları

- prompt() : İstem Kutusu prompt('İstem iletisi.', 'Varsayılan değer');
- alert() : Uyarı Kutusu
- alert('Uyarı iletisi.');
 veya
- var Deg = 'Uyarı iletisi.';
- alert(Deg);



(üyelik işlemlerinde boş bırakılan bir yer olduğunda uyarı verilebilir)

Popup Kutuları

 confirm() : Onay Kutusu confirm('Onay iletisi.'); veya var Deg = 'Onay iletisi.'; confirm(Deg);



- Seçilen mesaj(lar)ı silmek istediğinizden emin misiniz?" Onay iletisidir. Geriye true veya false değeri döner.
- var cevap confirm('Veriyi silmek istiyor musunuz?')

- <html> <head>
- <script type="text/javascript">
- <!-- function dogrulama()</pre>
- var cevap = confirm("Sayfadan cikmak istiyormusun?")
- if (cevap)
- { alert("Güle Güle!")
- window.location = "http://www.google.com/";
- }
- else { alert("Teşekkürler kaldığınız için") }
- }
- //-->
- </script>
- </head> <body> <form> <input type="button" onclick="dogrulama()" value="Sayfadan ayrıl"> </form> </body> </html>

Fonksiyonlar

- JavaScript dilinde, kodların yeniden kullanılabilmesi için kullanılır. Genel kullanımı:
- •
- function fonksiyon_ismi(parametre1, parametre2)
- {
- //yapılacak işlemler
- }
- •
- Fonksiyon içinde hesaplanan değer, return ifadesi ile geri döndürülür.

- Örnek: JavaScript ile Toplama
- function topla(deger1, deger2)
- {

• }

- var sonuc= deger1+deger2;
- return sonuc;

• Örnek:

```
    //1. ve en yaygın olan yazım şekli

  function Islev1() {
   var Deg = "Islev1'in değişkeni";
   alert(Deg);
  //2. yazım
  Islev1 = function () {
   var Deg = "Islev1'in değişkeni";
   alert(Deg);
  //çağırma
  <input type="button" value="İşlevi çağır" onclick="Islev1()">
```

 Fonksiyonlar(işlevler) durağan komutlar olduğundan bazen sayfa yüklenmesiyle veya bunun gibi herhangi bir müdahele olmadan çalışmasını istediğimiz durumlarda aşağıdaki yazımları kullanırız.

```
    //1. yazım
        window.onload = function() {
            var Deg = "Islev1'in değişkeni";
            alert(Deg);
        }
        //2. yazım
        function Islev1() {
            var Deg = "Islev1'in değişkeni";
            alert(Deg);
        window.onload = Islev1(); //Islev1 şeklinde de yazılabilir
```

 İşlevlerimiz bağımsız değişkenlere (argument) de tabi olarak çalıştırılabilir ve bir işlev birden fazla bağımsız değişken alabilir.

```
    function Islev(Ileti)
```

```
{
alert(lleti);
}
```

<input type="button" value="1. ileti!" onclick="Islev('1. (değişken) ileti!')">

<input type="button" value="2. ileti!" onclick="Islev('2. (değişken) ileti!')">

* İşlevler sayısal sonuçlar elde etmek amacıyla da kullanılabilir.

```
function Carp(Deg1, Deg2) {
  Sonuc = Deg1 * Deg2;
  return Sonuc;
}
```

<input type="button" value="Çarp" onclick="alert(Carp(2, 2))">

Hazır Fonksiyonlar

- eval (): Bir argümanı değerlendiren veya çalıştıran (Evaluate/Execute) fonksiyondur.
- var x = 10; var y = 20; var a = eval("x*y") + "
"; var b = eval("2+2") + "
"; var c = eval("x+17") + "
"; var res = a + b + c;
- var toplamTutar = "((siparisMiktari * fiyatTL) * vergi)"
- if (vergiYok) {

}

- toplamTutar = "(siparisMiktarı * fiyatTL)"
- •
- Programın herhangi bir yerinde, eval() fonksiyonunu kullanarak aynı kodları yazmadan değişkenlerin değerlerini bulmaya yardımcı olacaktır.
- document.write(eval(toplamTutar))
- eval içeriği tırnaksız kullanılırsa değişkenlerde tırnak olacak

Hazır Fonksiyonlar

- isNaN (): Gelen veri ya da herhangi bir değişkenin sadece rakamlardan oluştuğunu anlamak için kullanılır. Rakamlardan oluşuyor ise false, değilse true.
 - var a = '123';
 isNaN(a); ==> false
 - var a = 'a123';
 isNaN(a); ==> true
- parsInt ("string"), ParseFloat("string"): string veri tiplerini tamsayı ve ondalık sayı tipine dönüştürmek için kullanılır.
 - document.write(parseFloat("3.14"))

Olaylar

- Bir Web sayfası üzerinde kullanıcının her türlü hareketi kontrol edilebilir. Bir kontrolün üzerine gelmesi, dolaşması ve üzerinden ayrılması gibi hareklere olay denir.
- Bu olaylar ise onClick, onMuoseOver, onMouseOut, onSubmit,onReset, onChange, onLoad, onUnLoad, onError, onAbort, onFocus, onBlur olarak belirtilebilir.

onClick

- Internet sitelerinin çoğunda en sık kullanılan JavaScript olayıdır. Sayfa üzerinde bir nesnenin fare ile tıklanıp bırakılması sonucunda gerçekleşen olaydır. Link, button ve resim nesneleri tıklanarak onClick olayı tetiklenebilir.
- Nesnelerin etiketlerinde ise onClick olaylarını tetikleyen fonksiyonların ismi bildirilmelidir.
- Ayrıca onDblClick, çift tıklanma olayını tetikler.

- Örnek: onClick Olayı
- function tikla()
- {
- alert("Tıklama işlemi gerçekleşti...");
- }
- <input type="button" name="tikla" value="tikla" onClick=tikla()>

onMouseOver, onMouseOut

- Fare nesnenin üzerindeyken **onMouseOver**, fare nesne üzerinden ayrılınca **onMouseOut** olayları devreye girer.
- function nesneUzerinde() {
- window.status="Şu anda nesne üzerindesiniz."; }
- function nesneDisinda() {
- window.status="nesnenin dışına çıktınız."; }
- <a href="http://www.google.com"</p>
- onMouseOver = nesneUzerinde()
- onMouseOut = nesneDisinda()> Google



onSubmit

- Web safyalarında ziyaretçinin forma bilgi girip sunucuya göndermesi durumlarında onSubmit devreye girer.
 Gönderilecek forma girilen verilerin uygunluğunun kontrolü bu olayın tetiklediği fonksiyonlara yaptırılabilir.
- function dogrula()
- { confirm ('Formu doldurduysanız OK'i tıklayınız');
- }
- <form action="mail.pl" method="post" onSubmit="dogrula()">
- confirm komutu, kullanıcıya Ok ve Cancel butonlarından oluşan bir diyalog penceresi açar.

onReset

- Form içinde kullanılan tüm metin alanlarının temizlenmesini sağlar. Doldurulan formda yanlışlık olduğunda bu olay tetiklenir. Kullanıcıya onay penceresi çıkartmak için de kullanılabilir.
- function sil()
- { return confirm('Silmek istediginize emin misiniz?'); }
- <form onReset="sil()">
- <input type="text" name="mail">
- <input type="reset" value="sil">
- </form>

onChange

- Bilgi girişi yapılan alanlarda, değişikliğin gerçekleştiği bilgisi onChange olayı ile tetiklenir.
 - <html> <head> <title>onChange</title>
 - <script language="JavaScript1.2">
 - function yaz(secnesnesi){
 - return secnesnesi.options[secnesnesi.selectedIndex].value}
 - </script> </head> <body> <form name="form1">
 - <select name="mesaj" size="3" onchange="alert(yaz(this.form.mesaj))">
 - <option name="mesaj" value="Bu">mesaj1
 - <option name="mesaj" value="program">mesaj2
 - <option name="mesaj" value="onChange komutu ile yapılmıştır">mesaj3
 - </select> </form> </body> </html>



onLoad, onUnLoad

- onLoad olayı sayfaya giriş yapıldığında gerçekleşir.
 onUnLoad olayı sayfadan çıkıldığında gerçekleşir.
- function giris()
- { alert("Sayfaya Giriş Yaptınız!"); }
- function cikis()
- { alert("Sayfadan çıktınız.."); }
- <body onLoad="giris()" onUnload="cikis()">
- </body>

onError, onAbort

- Ziyaret edilen sayfadaki nesneler, çeşitli nedenlerden dolayı tam olarak yüklenememiş olabilir. Genellikle resim nesnelerinin yüklenmesinde problem çıkabilir. Bu tür durumları ziyaretçiye bildirmek için onError veya onAbort olayları kullanılır.
- <img src="resim.gif" onError="alert('Resim dosyası</p>
- yüklenemedi')">

onFocus, onBlur

- onFocus olayı kullanıcı kontrollerine giriş yapılırken gerçekleşir. OnBlur olayı ise ve kullanıcı kontrollerinden çıkış yapılırken gerçekleşir.
- function dogru()
- { document.form1.mesaj.value="Lütfen hata yapmayın!"; }
- function sor()
- { document.form1.mesaj.value="isminiz alındı"; }
- <form name="form1" method="post">
- <h3>Lütfen isminizi yazınız!</h3>
- <input type="text" size="20" name="isim"</p>
- onfocus="dogru()" onblur="sor()">
- <input type="text" name="mesaj"> </form>

JavaScript - Hata Denetimi

try { // Hata oluşabilecek kodlar}
 catch (err) { // Hata oluştuğunda çalışacak kodlar }
 finally { alert(" bu bloktaki kodlar her durumda çalışacaktır");}

 try komut bloğunda ister hata oluşsun ister oluşmasın finally bloğundaki kodlar çalışacaktır. Bu anahtar sözcüğünün kullanılması isteğe bağlıdır. Bu anahtar sözcük kullanılırsa mutlaka catch bloğundan sonra yazılmalıdır.

throw

- Bu anahtar sözcüğünü kullanarak try bloğu içinde kendimiz özel istisnai durumları oluşturabilirsiniz. Bu anahtar sözcük try blokları içerisinde kullanılmalıdır.
 - Örnek:

<script type="text/javascript"> try

- { for var i=0; i<10; i++)</p>
- { if("i==4"); { throw ("i=4 hatası"); alert("Hata sonrası kodlar"); } }
- } catch(err) { alert(err) }
 </script>
- Throw içerisine yazdığımız ifade catch (err) bloğundaki err parametresine aktarılır.

- Tarayıcı Nesneleri-Window Nesnesi
- Genel pencere özelliklerini tutan nesnedir. Pencere açma ve kapama işlemleri için bu nesne kullanılır.
- Genel kullanımı:
- window.open(" url ","pencere_ismi","pencere_ozellikleri");
- window.close();
- Open komutu ile yeni bir pencere açılırken, close komutu ile pencere kapatılır.
- Yeni bir pencere açmak için open komutuna ilk parametrenin girilmesi zorunludur. Pencere_ismi, birden fazla pencere ile işlem yapıldığı durumlarda kullanılabilir.

• Window Nesnesi

Pencereye ait özellikler ;

 Özellik 	Varsayılan Değer	Açıklama
• width	auto	Açılan pencerenin pixel genişliğini belirtir.
• height	auto	Açılan pencerenin pixel yüksekliğini belirtir.
• left	auto	Ekranın sol noktasına ile pencere arasındaki uzaklığı verir.
• top	auto	Ekranın üst noktasına ile pencere arasındaki uzaklığı verir.
• menubar	no	Menu çubuğunun görüntülenmesini sağlar.
• Toolbar	no	Araç çubuğun görüntülenmesini sağlar.
Location	no	Adres çubuğunun görüntülenmesini sağlar.
Status	no	Durum çubuğunun görüntülenmesini sağlar.
Scrollbars	no	Kaydırma çubuklarının görüntülenmesini sağlar
Resizable	no	Penceresinin boyutlandırılmasını sağlar.

• Window Nesnesi

- Örnek: Yeni bir pencere açma
- window.open('http://www.bilgeadam.com',
- 'bilgeadam', 'menubar=no, toolbar=no, scrollbars=yes, location=yes,width=300, heigt=300')
- veya
- Click here for specific popup window



- Window Nesnesi
- history
- go(number|url):
- Internet tarayıcısı ile daha önce ziyaret edilmiş sayfalara tekrar ulaşabilmek için window.history.go(-1) komutu kullanılabilir. -1 ifadesi ile bir önceki sayfaya gidilir. Sayı arttırılarak daha önceki sayfalara da gidilebilir. Pozitif değerler kullanılarak ileri sayfalara gidilebilir.
- Bir geri git
 İki geri git
- İki ileri git

• Window Nesnesi

 back() : geçmiş belgeyi döndürür (1 geri gider), geçmişte bir belge bulamaz ise null sonucunu döndürür yani hiç bir sayfayı döndürmez.

Geri git

- forward() : gelecek belgeyi döndürür (1 ileri gider), gelecekte bir belge bulamaz ise null sonucunu döndürür yani hiç bir sayfayı döndürmez.
- İleri git
- Internet tarayıcısının en alt kısmında bulunan status penceresine erişmek için window.status komutu kullanılır.
- window.status ="JavaScript öğreniyoruz!";

 Javascriptler başlı başına konu olup verilen özellikler dışında daha birçok özelliği vardır. Burada sadece temel bilgi verilmiştir.



- VbScript, Microsoft tarafından geliştirilmiştir. Ancak, VbScript tüm tarayıcılar tarafından desteklenmez. Bu nedenle tüm tarayıcılarda çalışacak script yazılmak isteniyorsa, JavaScript kullanılmalıdır.
- Vbscript dilinin kullanılabilmesi için script etiketi içerisindeki language özniteliğine vbscript değeri atanır.
- <script language="vbscript">...</script>

- İlk VbScript Örneği
- html>
- ody>
- <script language="vbscript">
- document.write("İlk VBScript!")
- </script>
- </body>
- </html>


Web Programlamaya Giriş-VbScript

- Değişken tanımlamaları dim anahtar sözcüğü ile yapılır.
- html>
- ody>
- <script language="vbscript">
- dim isim
- isim="Bilge Adam"
- document.write(isim)
- </script>
- </body>
- </html>

Web Programlamaya Giriş-VbScript

- Metot tanımlaması Visual Basic.NET diline benzer. Geriye sonuç döndürmeyen metotlar Sub, geriye sonuç döndüren metotlar ise function anahtar sözcüğü ile tanımlanır.
- Bu tanımlanmış metotlar "MetotAdı()" anahtar sözcüğü ile çağrılabilirler.

Web Programlamaya Giriş-VbScript

Function Tanımlaması

- <html> <head>
- <script language="vbscript">
- function sehir()
- sehir = "Trabzon"
- end function
- </script> </head> <body>
- <script language="vbscript">
- document.write("En sevdiğim sehir: " & sehir()) </script>
- function procedure geri dönüş değerine sahiptir.
- </body> </html>



http://jquery.com/ Çavuş Musa, "Kendinize Güvenerek JQuery ",Kasım(2010),Almanya http://www.cemdemir.net/jquery/jquery-nedir-nasil-kullanilir-nasil-ogrenilir-465.html <u>http://www.phpkodlari.com/javascript-dersleri/jquery-nedir/</u> <u>http://www.eburhan.com/jquery-dunyasina-adim-atiyoruz/</u> <u>http://www.codekodu.com/jquery.php</u>

Jquery Nedir?

- Jquery bir javascript kütüphanesidir. Daha çok sayfalardaki animasyonlar için Flash alternatifi olarak kullanılan bu teknoloji ile, slaytlar, foto galeriler, tab menüler yapmak oldukça kolaydır.
- CSS ve Jquery birlikte kullanılabilir. CSS ile yapılan statik siteleri, JQuery ile bir adım ileri götürerek görsel açıdan zengin, animasyonlu bir site haline getirebilirsiniz.
- JQuery Nasıl Kullanılır ?
- JQuery sadece 1 dosyadan oluşan .js uzantılı bir dosyadır. Sürekli güncellenen bu dosyayı http://jquery.com indirebilirsiniz.
- <script src="jquery-2.1.4.js"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></scri
- jQuery Framework'unda tüm genel değişkenleri jQuery alanı denilen bir yerde kayıt ediliyor ve bu değişkenlere, jQuery() veya \$() şeklinde ulaşabilirsiniz.

Jquery Temel Örnekler

- <html> <head> <title>Ilk jQuery Ornegi</title>
- <script src="jquery-2.1.4.js"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></scri
- <script type="text/javascript">
- \$(document).ready(function() {
- \$("#b1").click(function(){alert('Merhaba');}); });
- </script>
- </head>
- ody>
- <h1>Dugmeyi tikla</h1>
- <button id="b1">Tikla beni</button>
- </body>
- </html>



Jquery Temel Örnekler

- <html> <head> <script src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></scrip
- <script type="text/javascript">
- \$(document).ready(function() {
- \$("#b1").click(function() { alert('Merhaba'); });
- \$("#b2").click(function() { alert('Tesekkurler'); });
- \$("#b3").click(function(){ alert('İyi misin?');});
- });
- </script> </head>
- <body> <h1>Dugmeyi tikla</h1>
- <button id="b1"> Bana tikla </button>
- <button id="b2"> Benide tikla </button>
- <button id="b3"> Bende varim </button>
- </body> </html>



Jquery Temel Örnekler-Taglardaki parametre değerlerini değiştirme

- <!DOCTYPE html>
- <html> <head> <title>Parametre</title>
- <script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></script></script><
- <script type="text/javascript">
- \$(document).ready(function() {
- \$("#t1").click(function() { \$("img").attr({src: "resim1.jpg" }); });
- \$("#t2").click(function() { \$("img").attr({ src: "resim2.jpg" }); });
- \$("#t3").click(function() { \$("img").attr({ width: "300" }); });
- \$("#t4").click(function() { \$("img").attr({ width: "100" }); }); });
- </script> </head> <body>
- <button id="t1"> Resim 1 </button>
- <button id="t2"> Resim 2 </button>
- <button id="t3"> Buyuk </button>
- <button id="t4"> Kucuk </button>
- <hr/> </body> </html>

Bu örnekte boş bir tagı kullanıldı ve jQuery ile parametrelerin değerlerini değiştirildi.arayıcıya resimler yüklendi. \$("img") koduna iyi bakarsanız # işaretini kullanılmadığını görürsünüz. Bu şekilde tüm taglarına hitap edilmiş olunur. tagının parametrelerini değiştirebilmek için attr() metodu vardır. attr() metodunun içinde {} parantezlerini kullanarak hangi parametreleri değiştirmek istiyorsak o parametreyi yazarız ve iki nokta üst üste koyulduktan sonra değeri tırnak içerisine yazılır



← □ · ☆ ▼ ♂ Resim 1 Resim 2	
Resim 1 Resim 2	Buyuk Kucuk
Resim 1 Resim 2	Buyuk Kucuk
	- Dayan - House
K	

Nesneler ve sınıflar

- JavaScript ve jQuery'de hazır metotları olan nesnelerin kullanılması bize bir çok kolaylık sağlar. Bunu DOM teknolojisini kullanarak ya da kendi oluşturduğumuz sınıflarla yapabiliyoruz.
- <html> <head> <title>Nesne olusumu</title>
- <script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></script></script><
- <script type="text/javascript">
- //nesne tanımlama
- var x = { ad:"Ebru", yas:12, konus: function() { alert("Zipla"); } };
- alert(x.yas); x.konus();
- </script> </head> <body>
- <h1>Nesneleri blok halinde oluşturma işlemi</h1>
- </body> </html>
- Nesne= {... } içerisinde blok açılarak tanımlanıyor

		man	anna
Firefox *			
S Parametre	Resim buyu []] file://3.html []] CSS []] Nesne ol.	× >	ф 🔍
- III file:	///C:/Users/ebru/ 🏫 🔻 C 🛛 🗱 - AVG Sect 🔎 👘 🏫	E .3 -	-
Nesnel	eri blok halinde olusturma is	slem	ni

Nesneler ve sınıflar

- <html> <head> <title>Nesne Erisimi</title>
- <script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></script></script><
- <script type="text/javascript">
- function Kisi (ad, yas) { this.ad = ad; this.yas = yas; };
- var x = {ad: "Ahmet", yas: 43 }; //nesne oluşturma
- var y = new Kisi ("Ayşe", 56); // metoda erişim ile nesne oluşturma
- alert(x["yas"]); alert(y["ad"]); // nesnelere erişim
- </script>
- </head>
- ody>
- <h1>Nesne bölümlerine erişim</h1>
- </body>
- </html>

x ve y nesnelerin değişkenlerine ya da metotlarına farklı olarak köşeli parantezler sayesinde de erişim sağlanabilir. Fonksiyona dikkat!!!

Fonksiyon ve metotlar

- JavaScript de fonksiyonları iki çeşit çağırılabilir. Birincisi fonksiyon doğrudan çağırılır, ikincisi fonksiyon referansıyla çağırılır. jQuery'de referans ile çağırma çok kullanılmaktadır. JavaScript'te doğrudan çağırma daha çok kullanılıyor. İki çağırma şekli örnekte gösterilmiştir.
- <html> <head> <title>Fonksiyon cagirma</title>
- <script type="text/javascript" src=" jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script
- <script type="text/javascript">
- function cikti() { \$("#cikti").html("fonksiyon cikti()"); }
- \$(document), ready(function() {
- // cikti();
- \$("#b1").click(cikti); }); </script> </head> <body>
- <h1>Fonksiyon cagirma</h1> <button id="b1"> Tikla </button>
- <div id="cikti"> </div> </body> </html>



cikti() fonksiyonunu click() metodunun içerisinde parantez olmadan çağırıyoruz. Bu çağırma şekli referans ile çağırma seklidir. cikti(); seklindeki çağırma sekliyse doğrudan çağırma seklidir.

İç Fonksiyonlar

- jQuery'de iç fonksiyonlar çok kullanılmaktadır. İç fonksiyonlar, bir fonksiyonun içinde olan başka fonksiyonlardır.
- <html> <head> <title>Callback</title>
- <script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></script></script><
- <script type="text/javascript">
- function benimFonksiyon(icerik) { \$(".cikti").html(icerik); }
- function getText(metin){return function(){ benimFonksiyon(metin); }; }
- \$(document).ready(function() {
- \$("#b1").click(getText("Ormanlik"));
- \$("#b2").click(getText("Gul bahcesi")); });
- </script></head><body><h1>Callback</h1>
- <button id="b1">Tikla</button>
- <button id="b2">Tikla</button>
- <div class="cikti"></div></body></html></div></body></html>



jQuery Efektleri

- Jquery' de yazı ve resimlere efekt vermek için çok kullanışlı özellikler vardır. Bu özellikler arka arkaya çağrılarak çeşitli animasyonlar oluşturulabilir.
- Alacağı değerler:
 - Milliseconds, "slow", "fast"
- .fadeOut: Buharlaşarak kaybolma.
 .fadeIn: .fadeOut'un tersidir.
- .fadeTo: Şeffaflık verir. fadeTo(hız, Şeffaflık)
 .slideUp: Slide özelliği yukarı doğru kapanma
 .slideDown: Slide özelliği aşağıya doğru açılma
 .slideToggle: Slide özelliği açılıp-kapanma
 .hide: Gizlenme
 .show: Görünme
 .animate: Css kodlarına göre değişir.

Jquery Temel Örnekler- Animasyon

- <html> <head> <title>slideToogle </title>
- <script src="jquery-2.1.4.js"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></scri
- <script type="text/javascript">
- \$(document).ready(function() {
- \$("#toggle1").click(function() {
- \$('#b1').slideToggle('slow'); });
- \$("#toggle2").click(function(){
- \$('#h2').slideToggle('slow'); }); });
- </script> </head> <body>
- <h1 id="u1">jQuery slideToogle Açılılıp Kapanma</h1>
- <button id="toggle1"> Toogle Resim </button>
- <button id="toggle2"> Toogle Yazi </button>
-

-
 <h2 id="h2">Acayip birsey bu ya.</h2>
- </body> </html>





- animate(özellik, süre(milisaniye)) metodu verilen koordinatlar doğrultusunda nesneyi kaydırmaktadır. Yaptığımız sadece animasyon metotlarını ard arda çağırmak oldu.
- <html> <head> <script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></script></script></scr
- <title>jQuery ile animasyon</title> <script type="text/javascript">
- \$(document).ready(function(){
- \$("*").css({ background: "#ecaae2", color: "#1e1e1e" });
- \$("#resim1").css({ position: "absolute", left: "20px", top: "160px" });
- \$("#resim2").css({ position: "absolute", left: "450px", top: "60px" });
- \$("#b1").click(function(){ \$("#resim1").animate({ left: '+=200px' }, 2000);
- \$("#resim1").animate({ top: '300px' }, 600); \$("#resim1").slideUp("slow");
- \$("#resim1").slideDown("slow");\$("#resim1").animate({ left: '20px', top: '160px' }, 700); });
- \$("#b2").click(function(){ \$("#resim2").animate({ left: '+=200px' }, 2000).animate({ top: '300px'
- }, 600).slideUp("slow").slideDown("slow").animate({ left: '400px', top: '60px' }, 700); }); });
- </script> </head> <body> <h1>jQuery ile animasyon</h1>
-
-
- <button id="b1"> Tikla </button> <button id="b2"> Tikla </button> </body> </html>

queue()-Kuyruğa Ekle ve dequeue()-Kuyruktan Çıkar

- Fonksiyonları sıralı olarak çağırmak istiyorsak queue() ve dequeue() fonksiyonları kullanılmaktadır. Çağırılan fonksiyonları bir yığına koymak istersek queueu() metodu, yığından çağrılacak fonksiyonlarda dequeueu() metoduyla çağrılır. Kuyrukta fonksiyon kalmayınca sonlanır.
- <html> <head> <script type="text/javascript" src=" jquery-2.1.4.js "> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></script></script></script><
- <title>queue() ve dequeue()</title> <script type="text/javascript">
- \$(document).ready(function() { \$("*").css({ background: "#ab1b52", color: "#1e1e1e" });
- \$("#resim1").css({position:"absolute", left:"20px", top:"160px" });
- \$("#resim2").css({ position:"absolute", left:"450px", top:"60px" });
- \$("#resim1").queue(function(){ \$(this).slideUp("slow"); \$(this).slideDown("slow"); });
- \$("#resim1").queue(function(){ \$(this).fadeOut("slow"); \$(this).fadeIn("slow"); });
- \$("#resim2").queue(function(){ \$(this).fadeOut("slow"); \$(this).fadeIn("slow"); });
- \$("#b1").click(function(){ \$("#resim1").dequeue(); }); \$("#b2").click(function(){
- \$("#resim2").dequeue(); }); });
- </script> </head> <body> <h1>jQuery ile animasyon</h1>
- <button id="b1">Tikla </button> <button id="b2"> Tikla </button> </body> </html>

jQuery.each(sayi,function(belirtec,value){} yapısını şu şekilde düşünebilirsiniz

each() Metodu

for (indis=0;indis<sayi.length;indis++)</pre>

- jQuery döngülerin zahmetinden kurtarmak için each() fonksiyonunu bize sunuyor. each() fonksiyonuyla dizilere erişim sağlayabildiğimiz gibi nesnelere de erişim sağlayabilmekteyiz.
- <html> <head> <script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></sc
- <title>each()</title> <script type="text/javascript">
- function isim() { this.yas = 15; this.ad = "Ebru"; }
- var sayi=[1,2,3,5,7,11,13,17,19,23,29]; var kisi = new isim();
- \$(document).ready(function(){
- jQuery.each(sayi,function(indis,value){
- \$("#cikti1").append((indis + 1),". Sayi: ", value, "
"); });
- jQuery.each(kisi,function(indis,value) {
- \$("#cikti2").append("Belirtec: ",indis,", Deger: ", value, "
"); });
- }); </script> </head> <body>
- <h1>Sayilar</h1> <div id="cikti1"></div> <h1>Nesne</h1>
- <div id="cikti2"></div> </body> </html>



context özelliği- val():alma, val("değer"):yazma

- \$() veya jQuery() tanımlamalarının ikinci parametrelerinde kayıt edilecek verilerin context'lerini (verilerin kayıt edilecek bölgeleri) girilir.
- <html> <head><title>jQuery Context</title>
- <script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script>
- <script type="text/javascript">
- \$(document).ready(function(){
- \$("#b1").click(function(){
- \$("#cikti").html(\$("input:text", document.forms[0]).val()); });
- \$("#b2").click(function(){
- \$("#cikti").html(jQuery("input:text", document.forms[1]).val()); }); });
- </script> </head>
- <body> <h1>jQuery Context</h1>
- <form> <input/> </form>
- <form> <input/> </form><hr/><</pre>
- <button id="b1"> Tikla </button> <button id="b2"> Tikla </button>
- <div id="cikti"> </div> </body> </html>

Firefox 🔻				
<pre> m () each()</pre>	jQuery() ve			
 C · ☆ ▼ C 	- AVI 🔎 🗣			
jQuery Context				
ebru]			
ayyıldız]			
Tikla Tikla ebru				

Dinamik Element Oluşturma

- jQuery'de taglar dinamik olarak oluşturulabilir. String formatında oluşturulan veri append veya appendTo() metoduyla her hangi bir taga eklenebilir.
- <html> <head><title>jQuery() ve \$()</title>
- <script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></script></script><
- <script type="text/javascript">
- \$(document).ready(function() {
- \$("#b1").click(function(){ var sayi = Math.random();
- \$("Sayi" + sayi + "").appendTo("#t1"); }); });
- </script> </head>
- <body> <h1>jQuery() ve \$()</h1>
- <button id="b1"> Tikla </button>
- </body> </html>
- append ile kullanımı
- \$("#t1").append("Sayi" + sayi + ");



eq() Metodu

- DOM elementleriyle işlem yapabilmek için eq() çok yararlı bir metottur. Metodun parametresi, aynı guruptaki elementlerin belirtecidir. Burada tagı aynı grup olmuş oluyor ve i değişkeniyle eq() metodundan yararlanarak tüm tagların CSS özelliklerini değiştirir.
- <html> <head> <title>eq()</title> <script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></s
- <script type="text/javascript">
- \$(document).ready(function(){
- for (i = 0; i < \$("img").length; i++) {</p>
- \$("img").eq(i).css("left", 50 + (i * 200) + "px");
- \$("img").eq(i).css("top", 50 + (i * 80) + "px");
- });</script>
- <style type="text/css"> img { position: absolute; } </style> <</p>
- <body style="text-align:center;font-size:12px">
-
-
- </body></html>



h1:eq(1) yazarak sayfadaki ikinci <h1> tagıyla işlem yapılacağı anlatılmaktadır

- <html> <head> <title>CSS</title>
- <script type="text/javascript" sre="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></script></script
- <script type="text/javascript">
- \$("document").ready(function() {
- \$("h1:eq(1)").css ({ background:"yellow", color:"white", textAlign:"center" });
- \$("h2").css({ background:"black", color:"red", textAlign:"center" }); });
- </script> </head> <body>
- <h1>Birinci satir</h1>
- <h1>lkinci satir</h1>
- <h1>Ucuncu satir</h1>
- <h1>Dorduncu satir</h1>
- <h2>ve</h2>
- <h1>Besinci satir</h1>
- </body> </html>



get() Metodu

- DOM elementlerine ulaşmanın bir başka yolu da get() metodunu kullanmaktır.
 DOM nesnelerine doğrudan erişebilmek için bu yöntem çok uygundur. get() metoduyla \$() parantezinde belirtilen tüm tagları elde edilir.
- <html> <head> <title>get()</title> <script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></
- <script type="text/javascript">
- \$(document).ready(function(){
- var resim = \$("img").get();
- for (i = 0; i < resim.length; i++) {</p>
- resim[i].style.left = 50 + (i * 200) + "px";
- resim[i].style.top = 50 + (i * 80) + "px"; }
- }); </script>
- <style type="text/css"> img { position: absolute; }</style> </head>
- <body style="text-align:center;font-size:12px">
-
- </body> </html>



index() Metodu

- index() metoduyla bir elementin sayfadaki sıra pozisyonu bulunur. İlk resmin eklenme sırası 0, ikinci resmin ise 1 dir.
- <html><head><title>index()</title><script type="text/javascript" src="jquery-2.1.4.js"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></sc
- <script type="text/javascript">
- \$(document).ready(function(){
- \$("img").css("width", "250px");
- \$("img").click(function(){
- var index = \$("img").index(this);
- \$("#cikti").text("belirtec " + index); }); });
- </script> </head>
- <body style="text-align:center;font-size:12px">
-
- <h1 id="cikti"></h1>
- </body> </html>



data() ve removedata() Metodu

- Html elemanlarına bilgi ekleme ve kaldırma için kullanılır. Değer bulunamazsa undefined ifadesini yazar.
- \$("elaman").data("anahtar", "değer");
- <html><head><title>Veri Kayiti</title><script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script</p>
- <script type="text/javascript">
- \$(document).ready(function(){
- \$("button").click(function(e){var deger;
- switch (\$("button").index(this)) {
- case 0: deger = \$("img").data("bilgi"); break;
- case 1: \$("img").data("bilgi", "Bu bir deger"); deger = \$("img").data("bilgi"); break;
- case 2: \$("img").removeData("bilgi"); deger = "değer silindi"; break; }
- \$("span").text("" + deger); }); });
- </script> </head> <body> <h1>Veri Kayiti</h1>
- <button> Metainfo olustur ve deger ata </button>
- <button> Metainfo sil </button>
- <h2>Metainfo degeri: .</h2>
- </body> </html>



extend() Metodu

- extend() fonksiyonu kullanılan nesneye yeni özellikler ekleme yapılmasına izin vererek geliştirmektedir. Aşağıdaki örnekte sayı nesnesi bir metot olarak genişletilmiştir.
- <html> <head><title>extend()</title>
- <script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></script></script><
- <script type="text/javascript">
- jQuery.extend({ sayi: function() { return Math.random(); } });
- // jQuery.extend({sayi: 12});
- \$(document).ready(function() {
- \$("#cikti").text(jQuery.sayi()); });
- //\$("#cikti").text(jQuery.sayi); });
- </script> </head>
- <body> <div id="cikti"> </div> </body>
- </html>



CSS Erişimi - css() Metodu

- css() metodu iki şekilde kullanılır. Birinci şekildeki kullanımda verdiğimiz parametreye göre değer elde edilir. İkinci kullanım sekliyse, iki parametre kullanarak CSS parametresinin değeri düzenlenir. css("color") ile color değişkeninin değerini elde ediyoruz. css("color", "blue") veya css ({ color:"blue" }) ile color değişkenine blue değerini atıyoruz. \$("*").css(ile başlayan kod bölümünde ise bir nesne atıyoruz.
- <html> <head> <script type="text/javascript" src="jquery-2.1.4.js"> </script> <title>css() metodu</title>
- <script type="text/javascript">
- \$(document).ready(function(){
- \$("*").css({ background: "yellow", color: "gray" });
- \$("button:first").click(function() {
- \$("#cikti").text(\$("h1:first").css("color")); }); //1
- \$("button:last").click(function() { \$("h1:first").css("color", "blue"); }); }); //2
- </script> </head> <body>
- <h1>Merhaba</h1> <h1>İyi günler</h1>
- <button> Renk sorgula </button> <button> Renk duzenle </button>
- //\$("h1:eq(0)").css ({ color:"blue" });

Firefox T
css() metodu
Merhaba
İyi günler
Renk sorgula Renk duzenie rgb(0, 0, 255)

addClass() ve removeClass() Metodu

- jQuery'le dinamik olarak CSS sınıfları eklenebilir veya silinebilir. addClass() metodu sınıf ekler, removeClass() ise parametrede verilen sınıfı siler.
- <html> <head><script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></scr
- <title>addClass() ve removeClass()</title>
- <style type="text/css"> .mavi { color: blue; } </style> </head>
- <script type="text/javascript">
- \$(document).ready(function() {
- \$("button:first").click(function() {
- \$("h1:first").addClass("mavi"); });
- \$("button:eq(1)").click(function() {
- \$("h1:first").removeClass("mavi"); }); }); </script>
- </head> <body>
- <h1>Calisan demir paslanmaz</h1>
- <button> Sinif ekle </button> <button> Sinif sil </button>
- </body> </html>



Sınıf ekle düğmesine basılınca yazı mavi oluyor ve Sınıf sil düğmesine basılınca yazı siyah oluyor.

- <html> <head> <script src="jquery-2.1.4.js"></script> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></scrip
- <style type="text/css">
- .cssSinif { background: green; font-size: 37px; } </style>
- <script type="text/javascript"> \$(document).ready(function()
- \$("#a").click (function(event) { \$("#e").addClass("cssSinif"); });
- \$("#b").click(function(event){ \$("#e").removeClass("cssSinif"); });
- \$("#c").click(function(event){ \$("#e").hide("slow");});
- \$("#d").click(function(event){ \$("#e").show("fast"); });
- </script> </head> <body> <h1>jQuery ile Web Sayfasi</h1>
- <button id="a"> CSS-Sinifi ekle </button>
- <button id="b"> CSS-Sinifi kaldir </button>
- outton id="c">Gizle </button>
- <button id="d"> Goster </button> <hr/>
- <div id="e"> Bu dunya bizim kirletmeyelim </div> <hr/>
- </body> </html>

});



hasClass() Metodu

 Bir grup elementin veya bir grup tagın CSS sınıfı var olup olmadığını kontrol eder. hasClass() metodunda parametre olarak test etmek istediğimiz sınıfın adı yazılır.

toggleClass() Metodu

- addClass() ile sınıf ekleyebiliyoruz ve removeClass() ile sınıfı siliyoruz. tooggleClass() metodu bir iki metodun bir kombinasyonudur. Eğer sınıf varsa siler, yoksa ekler.
- <html> <head> <title>animate </title>
- <style type="text/css">.mavi {color: blue;}</style></head>
- <script src="jquery-2.1.4.js"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></scri
- <title>jQuery ile animasyon</title> <script type="text/javascript">
- \$(document).ready(function()
- {
- \$("button").click(function(){ alert(\$("h1").hasClass("mavi"));
- \$("h1").toggleClass("mavi");}); </script></head>
- statistics of the state of t
- sinif ekle/sil </button> </body> </html>

Firefox 🔻		
stodu addCla	ss() v 🔅 addClass() v	. 🗌 toggleCl × 📎 🕂 🤊
 € • ☆ ▼ 	С 📲 - АИ Р	*
Bedava	sirke bald	an tatlidir
Sinif ekle/sil		

Düğmeye basılınca yazı mavi oluyor ve tekrar basılınca yazı siyah oluyor.

position() Metodu

- position() metodundaki left ve top değişkenleriyle pozisyonlamayı çok iyi kontrol eder. left ve top değişkenleri bize sorguladığımız elementin pozisyonunu, o elementin parentini referans alarak iletir.
- <html> <head> <script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></script></script></scr
- <title>position() metodu</title> </head>
- <script type="text/javascript">
- \$(document).ready(function(){
- \$("button:first").click(function(){
- var position = \$("h1:last").position();
- \$("#cikti").text("x: " + position.left + ", y: " + position.top); }); });
- </script> </head> <body>
- <h1>Cok gezen cok bilir</h1>
- <h1>Cok bilen cok yanilir</h1>
- <button> lkinci atasozun poziyonu </button>
- <div id="cikti"> </div> </body>
- </html>



scrollTop() ve scrollLeft() Metodu

- scrollTop() ve scrollLeft() metotlarına bir parametre değeri verirsek seçtiğimiz element kaydırma çubuklarına tıklanmış gibi kayar. Parametresiz olarak kullanırsak, kaydırılmış elementin pozisyon değerlerini alır.
- <html> <head> <script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></script></script></scr
- <title>scrollTop() ve scrollLeft()</title> </head> <script type="text/javascript">
- \$(document).ready(function() {
- \$("button:first").click(function() { \$("div:eq(0)").scrollTop(45); \$("div:eq(0)").scrollLeft(17);
- \$("#cikti").text("scrollTop: " + \$("div:eq(0)").scrollTop() + ", scrollLeft: " + \$("div:eq(0)").scrollLeft()); });
- \$("div:eq(0)").css({ 'width': '400px', 'height': '100px', 'overflow': 'auto', 'position':"absolute",
- 'left':"20px", 'top':"100px" }); });
- </script> </head> <body>
- <button> Kaydir </button>
- <div> </div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></div></ti>
- </html>



height() ve width() Metotları

- Bu iki metoda parametre verilirse, elementlerin yüksekliği ve genişliği değiştirilir. Parametresiz kullanılırsa elementlerin genişliğini ve yüksekliğini elde edilir.
- <html> <head> <script type="text/javascript" src="jquery-2.1.4.js"> </script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script</script></script></script></script></script></script></scr
- <title>height() ve width()</title> </head> <script type="text/javascript">
- \$(document).ready(function(){
- \$("button:first").click(function() { \$("img").width(1000);
- \$("#cikti").text("Genislik: " + \$("img").width()); });
- \$("button:eq(1)").click(function(){ \$("img").height(100);
- \$("#cikti").text("Yukseklik: " + \$("img").height()); }); });
- </script> </head>
- <body> <button>Genislik</button> <button>Yukseklik</button>
- <hr />
- <div id="cikti"></div> </body> </html>



 JQuery'de Java script gibi bir çok özelliği olan script dilidir. Burada sadece en çok kullanılan temel özellikler verilmiştir. Açık kaynak kodlu olan bu yapı http://jquery.com/ adresinden tüm detayları ile incelenebilir. Zengin özellikleri kullanılabilir.

Örnekler-http://www.w3schools.com

- <!DOCTYPE html>
- html>
- <head>
- <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></
- <script>
- \$(document).ready(function(){
- \$("button").click(function(){
- alert("Value: " + \$("#test").val());
- \$("#test").val("yeni değer")
- });
- });
- </script>

Name: Mickey Mouse

Show Value



- </head>
- <body>
- Name: <input type="text" id="test" value="Mickey Mouse">
- <button>Show Value</button>
- </body>
- </html>

Örnekler-http://www.w3schools.com

- <!DOCTYPE html>
- <html><head>
- <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></
- <script>
- \$(document).ready(function(){
- \$("button").click(function(){
- \$(this).hide();
- **}**);
- });
- </script>
- </head>
- <body>
- <h2>This is a heading</h2>
- This is a paragraph.
- This is another paragraph.
- <button>Click me</button>
- </body>
- </html>

This is a heading

This is a paragraph.

This is another paragraph.

Click me

This is a heading

This is a paragraph.

This is another paragraph.

Örnekler-http://www.w3schools.com

- <!DOCTYPE html>
- <html> <head>
- <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></
- <script>
- \$(document).ready(function(){
- \$("#btn1").click(function(){
 - \$("img").before("Before");
- });
- \$("#btn2").click(function(){
 - \$("img").after("<i>After</i>");
- });
- **}**);
- </script>
- </head> <body>
-

- <button id="btn1">Insert before</button>
- <button id="btn2">Insert after</button>
- </body> </html>


Örnekler-http://www.w3schools.com

- <!DOCTYPE html>
- <html><head><script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></script></scri
- <script>
- \$(document).ready(function(){
- \$("#btn1").click(function(){
- \$("#test1").text("Hello world!"); });
- \$("#btn2").click(function(){
- \$("#test2").html("Hello world!"); });
- \$("#btn3").click(function(){
- \$("#test3").val("Dolly Duck"); });
- });</script></head><body>
- This is a paragraph.
- id="test2">This is another paragraph.
- Input field: <input type="text" id="test3" value="Mickey Mouse">
- <button id="btn1">Set Text</button>
- <button id="btn2">Set HTML</button>
- <button id="btn3">Set Value</button>
- </body> </html>

This is a paragraph.
This is another paragraph.
Input field: Mickey Mouse
Set Text Set HTML Set Value
Hello world!
Hello world!
Input field: Dolly Duck
Set Text Set HTML Set Value

ASP.NET (Active Server Page)

ASP.NET Nedir?

- Active Server Page- Aktif Sunucu Sayfası, Microsoft firmasının geliştirdiği .Net tabanlı web programlama dilidir.
- Sunucu taraflı işlemlerin gerçekleştirilmesinde kullanılan yaygın olan dillerdendir. Sunucu taraflı kodlar herhangi bir .Net tabanlı dil ile yazılabilir.

ASP ile ASP.NET in Karşılaştırılması

ASP	ASP.NET
Klasik ASP sunucusu .asp uzantılı dosyalarla	ASP.NET sunucusu .asp, aspx ve .asmx uzantılarını
ilgili taleplere cevap vermektedir.	kullandığı için tüm bu dosyaları aynı sunucu üzerinden
	çalıştırmamız mümkündür.
Az sayıda dil seçeneğinden birini kullanarak	Sunucu taraflı kodlar herhangi bir .NET dilinde yazılabilir.
sunucu taraflı script yazılabilir ve çalışma	Bu diller yorumlanmaz derlenir.
anında yorumlanır.	
Asp script'lerinin işlevselliği COM nesneleriyle	.NET System sınıflarından herhangi biri kullanılabilir,
genişletilebilir.	mevcut COM nesneleri de çağrılabilir.
Kod ve HTML genellikle sayfa içinde yerel	ASP.NET sayfasında yerel kod da kullanılır fakat genelde
olarak karışmıştır.	code-behind sınıflarında HTML den ayrı yer alır.
Gönderilen HTML form alanlarını işler	Gönderilen form alanlarını geçerlilik denetimini yapar ve
	işler
Ayarlar, özel olaylarla ilgili kodu çalıştıran	Ayarlar, XML formatındaki dosyalarda tutulur.
özel ASP dosyasında tutulur.	

WEB SİTESİNİN OLUŞTURULMASI File/New/Web Site...

🥐 S	tart Page - Microsoft Visual	Stud	io	
File	Edit View Tools Test V	Vindov	w Community	Help
	New 🕨	67	Project	Ctrl+Shift+N
	Open 🕨	٩	Web Site	
	Close	۳	File	Ctrl+N
đ	Close Solution		Project From Ex	isting Code
	Save Selected Items Ctrl+S	JU	uuioz	005
	Save Selected Items As			

New Web Site				? 🗙
<u>T</u> emplates:				
Visual Studio i	installed template	5		
ASP.NET We	<mark>b Site</mark> 5ite	🥵 ASP.NET Web Service ASP.NET Crystal Reports Web Site	🞲 Personal Web Site Starter Kit e	
My Templates	5			
Search Onlin	e Templates			
A blank ASP.NET V	Veb site			
Location:	HTTP	http://localhost/WebSite/Projer	n	Browse
Language:	Visual C#	· /		
			ОК	Cancel

WEB SİTESİNİN OLUŞTURULMASI

 C# Web uygulama projesinde Location alanında Web sunucusunun adı yazılı olacaktır. (<u>http://localhost</u>).
 Girdiğimiz bilgilerin doğru olduğunu kontrol ettikten sonra OK tuşuna basalım böylece proje oluşacaktır.



WEB SİTESİNİN OLUŞTURULMASI

 Http yerine File System ve Ftp seçenekleri ile de uygulamalarınızı geliştirebilirsiniz.

Location:	File System 🔽	C:\Documents and Settings\etanyildizi.FIRAT\Belgelerim\Visual Stu 🔽 📴 📴 rowse
Language:	File System HTTP FTP	
		OK Cancel

Language kısmında ise script dilini değiştirebilirsiniz.

Location:	File System	*	C:\Documents and Settings\etanyildizi.FIRAT
Language:	Visual C#	~	
	Visual Basic		
	Visual J#		

WEB SİTESİNİN OLUŞTURULMASI

 Visual Studio ile çalışırken web sitesi uygulaması geliştirmek için bir yöntem daha vardır. ASP.NET web uygulaması oluşturmak. Bunun için File > New Project'i seçmek.

New Project					? ×
Recent Templates		.NET Framewo	rk 4 🔹 Sort by: Default	•	Search Installed Templates
Installed Templates	*	ASP.I	NET Web Application	Visual C#	Type: Visual C#
✓ Visual C# Windows		ASP.I	NET MVC 2 Web Application	Visual C#	Web user interface
Web Diffice		ASP.	NET Empty Web Application	Visual C#	
Reporting SharePoint	=	ASP.	NET MVC 2 Empty Web Application	Visual C#	
Silverlight Test		ASP.	NET Dynamic Data Entities Web Application	Visual C#	
WCF Workflow		ASP.I	NET Dynamic Data Linq to SQL Web Application	n Visual C#	
 Visual C++ Visual F# Other Project Type 		ASP.	NET AJAX Server Control	Visual C#	
 Database Modeling Projects 	· _	ASP.	NET AJAX Server Control Extender	Visual C#	
Online Templates		ASP.	NET Server Control	Visual C#	
<u>N</u> ame:	WebApplication1	<i></i>			
Location: C:\Users\Docum		ents\Visual Stud	io 2010\Projects		Browse
Solution na <u>m</u> e:	WebApplication1	L.			✓ Create directory for solution Add to source control
					OK Cancel

WEB SITESININ OLUŞTURULMASI

Sayfa Mizanpajı

 Default.aspx dosyasının üzerinde herhangi bir noktayı sağ fare tuşu ile tıklayarak Properties' i seçelim. Sayfa ilgili tüm ayarlamalar burada yapılmaktadır.

Pr	operties	√ +⊐ X	2
D	OCUMENT	•	los l
0	Ż↓ □		ution E
	ALink		1 <u>4</u>
	Background		ore
	BgColor		
	Class		1
	Culture		2
	Debug		- OP
	Dir		grtie
	EnableSessionState		Ľů,
	Id		
	Language	C#	
	Link		
	MasterPageFile		
	Style		
	StyleSheet		
	StyleSheetTheme		
	Text		
	Theme		
	Title	Dinamik Sayfa Örneği	
	Trace		
	TraceMode		
	UICulture		
	VLink		

WEB SITESININ OLUŞTURULMASI

Sayfa Mizanpajı

 Style kısmı ile sayfa boşluk ayarları, arkaplan rengi yazı fontu gibi temel özellikler belirlenebilir.

Style Builder					? 🛛
Font Background Text Position Edges Edges Lists	Eont name Font name System font: Font attributes Size Size Absolute: Relative: Relative: Relative: Relative: Relative:		Effect:	Small caps:	
		Sample text			
				ОК	Cancel

ASP.NET Sayfasının Yapısı

• <%@ Page Language="C#" %> -----→ Sayfa direktifi



ASP.NET ile Uygulama Geliştirmek

- ASP.NET ile geliştirilen uygulamaların; Internet, Extranet veya Intranet üzerinde çalışabilmesi için Web Sunucularına ihtiyaç duyulur.
- IIS (Internet Information Services), Windows sistemler için web tabanlı uygulama geliştirme ve yayınlama amacıyla kullanılan web sunucusudur.

Uygulama Yayınlamak

- Default.htm ismindeki HTML sayfayı IIS üzerinden yayınlayın.
- 1. "C:\Inetpub\wwwroot" klasorüne gidin.
- 2. wwwroot penceresi içerisinden Dosya menüsünü tıklayın.
- 3. Dosya menüsü içerisinden Yeni alt menüsünü tıklayın.
- 4. Yeni alt menüsü içerisinden Metin Belgesi komutunu vererek Yeni Metin Belgesi oluşturun.
- 5. Oluşturduğunuz metin belgesi içerisine aşağıdaki HTML(Hyper Text Markup Language) kodları ekleyin ve dosyayı kaydedin.

Uygulama Yayınlamak

- html>
- <head>
- <title>HTMLPage1</title>
- </head>
- ody>
- Hoş Geldiniz.
- </body>
- </html>

Uygulama Yayınlamak

- 6. Metin belgesinin ismini Default.htm olarak değiştirin.
- 7. Intrernet Explorer açın ve aşağıdaki adreslerden herhangi birini adres çubuğuna yazın.
 - a. http://localhost
 - b. http://127.0.0.1
 - c. http://MakinaAdı
 - d. http://lpNumarasi
- Localhost: Lokal makina adı.
- 127.0.0.1 : Lokal IP numarası.
- Makina Adı: Ağ içerisindeki bilgisayar adı.
- Ip Numarası: Ağ içerisindeki Ip Numarası.

ASP.NET WEB FORM VE KONTROLLERİ İLE ÇALIŞMAK

- ASP.NET ile uygulama geliştirirken kullanılan temel bileşenler Web Formlar ve Web kontrolleridir.
- Web Form, IIS tarafından çalıştırılan HTML kod ve kontrollerin birleşiminden oluşur.
- Bu formlara eklenen kontroller, sunucu veya istemci taraflı çalışabilirler.

Web Form Bileşenleri

- Web Form, ASP.NET uygulamalarının yapı taşıdır. Visual Studio .NET ortamı aracılığı ile eklenen kontrollerin ve Visual C# kodlarının birleşimi Web Formu oluşturur. Web formlar, .aspx uzantılı ara yüz dosyası ve .aspx.cs uzantılı kod dosyalarından oluşur.
- Web Formların genel özellikleri aşağıdaki gibidir.
 - Page Özelliği
 - Form Özelliği

- Tüm sayfa içinde tanımlanacak fonksiyonların değerlerini içerir. <@Page> etiketi ile gösterilir ve her .aspx uzantılı dosyada bulunması gerekir.
- <@Page> etiketinde, sayfanın yapısı ile ilgili özellikler bulunur.

Language

 Sayfa içinde kullanılacak dil seçeneğini bildirilir. ASP.NET uygulamalarında genellikle VB ve C# dilleri tercih edilir.

•

- <"@Page Language="c#" ...">
- <"@Page Language="vb" ...">

CodeBehind

- Web formların, Visual Basic .NET veya C# uzantılı kod dosyasını belirtir.
- <@Page CodeBehind="WebForm1.aspx.vb" ...>
- <@Page CodeBehind="WebForm1.aspx.cs" ...>

- SmartNavigation
- SmartNavigation özelliğine True değeri ayarlanırsa, sayfa yeniden yüklendiği zaman, kaydırma çubuklarının sayfa içerisindeki yeri korunur. Böylece sayfa ilk konumunda kalır.
- <@Page Language="c#" CodeBehind="WebForm1.aspx.cs" SmartNavigation="True" >



• ViewState

- EnableViewState özelliği ile objenin içerisine girilen bilgi ne olursa olsun, sunucu bunu bir değişkende tutup tekrar kullanıcıya geri döndürür. Bu durum sunucuya gönderilen veriler üzerinde hata oluşması durumunda, bilgilerin kaybolmamasını sağlar. Üç şekilde yapılabilir;
- Sayfa kapsamında; bu özelliğin tüm kontrolleri içermesi için, Page yönerge satırında tanımlanması gerekir. Bu özellik True veya False değeri alabilir.
- <@Page EnableViewState="True" ...>



• ViewState

- Kontrol kapsamında; kontrole ait EnableViewState özelliği kullanılabilir. Bu durumda, Page yönerge satırında belirtilen değer geçersiz olur.
- <asp:Button ... EnableViewState="false" ...>
- Uygulama kapsamında ise web.config dosyasında
- <Pages EnableViewState="false" ... />
- şeklinde bir ifadeye yer verilmeli. Böylece web uygulamamızdaki bütün sayfalarda, ViewState'e durum bilgisi değerleri eklenmesi engellenir.

Viewstate ile İlgili Önemli Noktalar

- ViewState ile ilgili birkaç önemli noktayı vurgulamakta fayda var.
 - ViewState ancak sunucu taraflı formlarda oluşturulur.
 Diğer bir ifadeyle form tanımlamanız
 - <form runat=server> şeklinde olmalıdır.
 - ViewState değerleri, kesinlikle ziyaretçinin (client) sayfasında saklanır ve sunucuda hiç bir yer kaplamazlar.
 - ViewState sayfanın ve sunucu kontrollerinin bütün değerlerini, gizli bir form elemanı olarak tek bir string biçiminde, kodlanmış halde tutar. Bunlar gizli form elemanlarıdır ve browser'dan sayfanın kaynak kodlarına baktığınızda görürsünüz.

StateBag

- ViewState kullandığımız kontrollerde, kullanıcılar, kontrole veri girilmesi gerekir. Ancak bazen, programcı için sayfadaki herhangi bir durumu muhafaza etmek ve saklamak gerekebilir. Diğer bir ifadeyle ViewState mekanizmasını kendi girdileri için kullanmak isteyebilir. Bu durumda
 StateBag denilen ve form postalandığı zamanki değerleri tutan bir nesne kullanılır.
- •
- Kullanıcının girmediği, düzenlenmiş bilgileri saklamak için kullanılan bir nesnedir. İşleyişi aynen ViewState sisteminde olduğu gibidir.
- ViewState ["değişkenin_ismi"]= değer

StateBag

- <%@ Page Language="c#" Debug="true" codePage="28599" %>
- <script runat="server">
- void yazdir(Object Sender, EventArgs e)
- { ViewState["isim"] = "zafer";
- mesaj.Text = (String)(ViewState["isim"]); }
- </script>
- </head>
- <body> <form id="Form1" runat="server"> </P>
- <asp:Label id="mesaj" Runat="server" />
- </P>
- <asp:Button id="buton" Text="gönder" Runat="server" OnClick="yazdir" />
- </form></body></HTML>

Namespace Eklemek

- Visual Studio kendisi için gereken Namespace'leri, uygulamalarına otomatik olarak dahi eder ve kullanır.
- Bunlar;
 - System
 - System.IU
 - System.Collections
 - System. Web
 - System. Web.UI
 - System. Web.UI.HTMLControls
 - System. Web.UI.WebControls
- Oluşturduğumuz Namespace'leri de ASP.NET uygulamalarında kullanmak için Namespace'i sayfanın başında bildirmemiz (*import*) gerekir.
- <%@ import Namespace="namespace in ismi" %>

FORM Özelliği

Method

- Web kontrol özelliklerinin, sunucuya gönderilme şeklini belirler. İki değer alabilir:
 - **Post:** İsim ve değer bilgilerini, HTML bilgisinin üst bilgisine yazarak gönderir.
 - <form method="Post" ...>
 - Get: İsim ve değer bilgilerini, sayfa adının sonuna ekleyerek gönderir.
 - <form method="Get" ...>

FORM Özelliği

• Id

- Formun isim bilgisini verir. CodeBehind sayfası içerisinde, forma işlem yaptırmak için kullanılır.
- <form id="deneme" ...>

Runat

- Web formlarda kullanılan kontrollerin sunucu ile haberleşerek çalışabilmesi için runat="server" bildirimi kullanılır. Bu özellik sadece server değerini alabilir.
- <form runat="server" ...>

SERVER(SUNUCU) KONTROLLER

- Web sunucu üzerinde çalışan kontrollerdir. İki tür server kontrolü vardır. Bunlar:
 - HTML Server Kontrolleri
 - Web Server Kontrolleri (ASP.NET Kontrolleri)
- ASP.NET server kontrolleri System.Web.UI.Control sınıfından türetilir. Her ASP.NET server kontrolü
 <asp:Kontrollsmi> etiketi ile bildirilir.
- HTML kontrolleri ise System.Web.UI.HtmlControls isim alanında bulunur.

SERVER(SUNUCU) KONTROLLER

- Button, TextBox, DropDownList gibi server kontrollerinin çalışma modeli, istemci taraflı HTML kontrollerinin çalışma modelinden oldukça farklıdır.
- ASP.NET server kontrolleri, tamamen sunucu üzerinde çalışır ve geri plandaki tüm işleyişleri ara yüzle gizlenerek gerçekleştirilir.
- Bir kontrolün sunucu tarafında çalıştığı runat="server" özelliği ile belirlenir.
- <asp:Button id="Buton1" runat="server" Text="Tiklayınız" />

SERVER(SUNUCU) KONTROLLER

- Örnekte istemci tarafında çalışan HTML Button kontrolü gösterilmektedir.
- <INPUT type="button" value="Bu Bir Html Button" >
- Bu kontrolün sunucu tarafında çalışması için, kontrole runat özelliği eklenmelidir. Böylece kontrol HTML server kontrolü haline getirilir.
- <INPUT type="button" id="button1" runat="server" value="Bu Bir Html Button" >

KONTROLLERİN SINIFLANDIRILMASI

- ASP.NET Web kontrolleri dört grupta listelenir. Bunlar;
 - Standart Kontroller
 - Doğrulama Kontrolleri
 - Zengin Kontroller
 - İlişkisel Liste Tabanlı Kontroller

STANDART KONTROLLER

- Bu kontroller, HTML kontrollere alternatif olarak tasarlanmıştır.
- Eski tip HTML kontrolleri ile yeni ASP.NET kontrolleri arasındaki en belirgin fark, her Web kontrolünden önce asp: ön ekinin kullanılıyor olmasıdır.

STANDART KONTROLLER

Web kontrol	Html Kontrol
<asp:button></asp:button>	<input type="submit"/>
<asp:checkbox></asp:checkbox>	<input type="checkbox"/>
<asp:hyperlink></asp:hyperlink>	
<asp:image></asp:image>	
<asp:imagebutton></asp:imagebutton>	<input type="image"/>
<asp:linkbutton></asp:linkbutton>	Yok
<asp:label></asp:label>	
<asp:listbox></asp:listbox>	<select size="5"> </select>
<asp:panel></asp:panel>	<div> </div>
<asp:textbox></asp:textbox>	<input type="text"/>
<asp:radiobutton></asp:radiobutton>	<input type="radiobutton"/>
<asp:dropdownlist></asp:dropdownlist>	<select> </select>
<asp:table></asp:table>	

Label

- Label, kullanıcıya bilgi vermek veya bir işlevin sonucu bildirmek için kullanılır.
- <asp:Label ID="Label1" runat="server" Text="Label Control" Font-Italic="true" />
- Label web kontrolü, Internet Explorer tarayıcısında aşağıdaki şekilde görülür.
- Label
 Control

Toolbox		
😑 Stand	ard	
Pointe	er	
A Label		
abl TextB	jox	
ab Butto ab LinkBu	Label Version 2.0.0.0 from Microsoft Corpo .NET Component	ration
• Özellikler

AccessKey	With	Font-Italic	Font-Underline
BackColor	Visible	Font-Name	ForeColor
BorderColor	CssClass	Font-Names	Height
BorderStyle	Enabled	Font-Overline	ID
BorderWidth	EnableViewState	Font-Size	Runat
MaxLength	Font-Bold	Font-Strikeout	TabIndex

Olaylar

OnDataBinding	OnDisposed	OnPreRender	
OnInit	OnLoad	OnUnload	

- Örnek:
- <%@ Page language="c#" %>
- html>
- <head>
- <script runat="server">
- private void tikla(object sender, System.EventArgs e)
- { Label3.Text=TextBox1.Text;
- Label3.Font.Size=Convert.ToInt32(TextBox2.Text); }
- </script>
- </head>
- <body>
- <form id="Form1" method="post" runat="server">

- <asp:textbox id="TextBox1" style="Z-INDEX: 101; LEFT: 136px; POSITION: absolute; TOP: 32px" runat="server" Width="136px"></asp:textbox>
- <asp:textbox id="TextBox2" style="Z-INDEX: 105; LEFT: 136px; POSITION: absolute; TOP: 72px" runat="server" Width="136px"></asp:textbox>
- <asp:label id="Label1" style="Z-INDEX: 104; LEFT: 32px; POSITION: absolute; TOP: 72px" runat="server" Width="96px" Height="16px" BackColor="White">Puntoyu Giriniz</asp:label>
- <asp:label id="Label2" style="Z-INDEX: 102; LEFT: 32px; POSITION: absolute; TOP: 32px" runat="server"Width="96px" Height="16px" BackColor="White">Adınızı Giriniz </asp:label>

Menüden Format/Set Position ile elamanları taşıyabilirsiniz

- <asp:button id="Button1" style="Z-INDEX: 103; LEFT: 192px; POSITION: absolute; TOP: 112px" onclick="tikla" runat="server" Width="80px" Height="24px" Text="Uygula"></asp:button>
- <asp:label id="Label3" style="Z-INDEX: 106; LEFT: 32px; POSITION: absolute; TOP: 176px" runat="server" Width="232px"

Height="24px"></asp:label>

</form> </body> </html>



TextBox

- TextBox, kullanıcının genelde bilgi girişi sağlamak için kullanlır.
 En sık kullanılan giriş kontrolüdür.
- <asp:TextBox id="userName" type="text" runat="server">
- Bu kontrol, Internet Explorer tarayıcısında şu şekilde gösterilir.
- <input name="userName" id="Text1" type="text" />

Web kontroller, WebControl sınıfından türemişlerdir. Bu yüzden Web kontroller BackColor, BorderColor, Enabled, Font, Height, Width özelliklerine sahiptir.



TextBox

• Özellikler

AccessKey	AutoPostBack	Font-Italic	Font-Underline	
BackColor	Columns	Font-Name	ForeColor	
BorderColor	CssClass	Font-Names	Height	
BorderStyle	Enabled	Font-Overline	ID	
BorderWidth	EnableViewState	Font-Size	Runat	
MaxLength	Font-Bold	Font-Strikeout	TabIndex	
TextMode	Visible	With	Wrap	
Rows				
 	L			

• Olaylar

TextChanged	OnDisposed	OnPreRender	OnDataBinding
OnInit	OnLoad	OnUnload	

- Button, form üzerindeki olayları sunucuya yollamak için kullanılır. En sık kullanılan onay kontrolüdür.
- Örnekte Dugme1 isimli button tıklandığında, Label kontrolune mesaj yazılır.
- <asp:Button id="Dugme1" runat="server" Text="Tiklayınız" OnClick="Dugme1_Click" runat="server"/>
-
- private void Dugme1_Click(object sender, System.EventArgs e)
- {
- Message.InnerHtml = "Beni Tıkladın.";
- }

• Örnek:

- <%@ Page Language="C#" AutoEventWireup="true" Inherits="_Default" %>
- <!DOCTYPE html ">
- <html ><head runat="server"> <title> TextBox</title>
- <script runat="server">
- private void kapa(object sender, System.EventArgs e) {Button1.Enabled=false;}
- private void gizle(object sender, System.EventArgs e) {Button1.Visible=false;}
- private void kalin(object sender, System.EventArgs e) {Button1.Font.Bold=true;}
- private void geri(object sender, System.EventArgs e)
- Button1.Font.Bold=false;Button1.Visible=true;
 - Button1.Enabled=true;
- }
- </script>
- </head>
- <body>

- <form id="form1" runat="server">
- <asp:Button id="Button1" style="Z-INDEX: 100; LEFT: 128px; POSITION: absolute; TOP: 32px" runat="server" Text="Test Butonu" Width="96px" Height="32px"></asp:Button>
- <asp:Button id="Button7" style="Z-INDEX: 107; LEFT: 112px; POSITION: absolute; TOP: 176px" runat="server" Text="İşlemleri Geri Al" Width="128px" Height="32px" BorderStyle="Inset" BackColor="#00cc99" OnClick="geri"></asp:Button>
- <asp:Button id="Button2" style="Z-INDEX: 102; LEFT: 224px; POSITION: absolute; TOP: 112px" runat="server" BackColor="#00cc99" BorderStyle="Ridge" Text="Gizle" Width="72px" Height="32px" OnClick="gizle"></asp:Button>

- <asp:Button id="Button3" style="Z-INDEX: 103; LEFT: 40px; POSITION: absolute; TOP: 112px" runat="server" BackColor="#00cc99" BorderStyle="Ridge" Text="Kapa" Width="80px" Height="32px" OnClick="kapa"></asp:Button>
- <asp:Button id="Button4" style="Z-INDEX: 104; LEFT: 136px; POSITION: absolute; TOP: 112px" runat="server" BackColor="#00cc99" Text="Kalın" Width="72px" Height="32px" BorderStyle="Ridge" OnClick="kalin"></asp:Button>
- </form>
- </body>
- </html>



Uygulama

- Aşağıda verilen işlemler için bir eğitim sayfası oluşturulacaktır.
- 1-Girilen iki sayı için 4 işlem yapan
- 2-Girilen iki sayıyı sınırları kabul edip buna göre fibonaci serisini yazan
- 3-Aritmetik ortalamalarını bulan
- 4-Girilen iki sayıyı sınırları kabul edip buna göre bu iki sayı arasında asal sayıları bulan web sayfasını oluşturunuz.

CheckBox, CheckBoxList

- CheckBox, kullanıcıya seçenekler arasından seçim yapma imkânı sunar. Onay kutusu işaretlenmiş ise True, işaretlenmemiş ise False değerini alır.
- Onay kutusunun durumu CheckedChanged metodu ile takip edilebilir.
- Örnekte CheckBox kontrolünün onay kutusu tıklandığı anda "Seçili", seçim işlemi geri alındığı anda "Seçili değil" mesajı yazılır. Seçim yapıldığı anda mesajın yazdırılmasını sağlayan AutoPostBack özelliğinin, True değeridir.

- <%@ Page Language="C#" %>
- <html > <head runat="server"> <title> CheckBox</title>
- <script runat="server">
- public void Check_Clicked(Object Sender, EventArgs e) {
- if(checkbox1.Checked) { Message.InnerHtml="Seçili"; }
- else { Message.InnerHtml="Seçili Değil"; } }
- </script> </head>
- <body>
- <form id="form1" runat="server">
- <asp:CheckBox id="checkbox1" runat="server" AutoPostBack="True"</p>
- Text="Üye Olmak İster misiniz?" TextAlign="Right"
 OnCheckedChanged="Check_Clicked"/>
-
 </form> </body> </html>





- <%@ Page Language="C#" AutoEventWireup="true" CodeFile="CheckBoxList.aspx.cs" Inherits="_06_web_form_elemanlari_03_CheckBoxList" %>
- <html> <head runat="server"><title></title></head>
- <body bgcolor="#99cccc">
- <form id="form1" runat="server"> <div>
- <asp:CheckBoxList id="secenek_listesi" CellPadding="5" CellSpacing="5"
 Borderwidth="1" runat="server">
- <asp:ListItem>kitap</asp:ListItem>
- <asp:ListItem>CD</asp:ListItem>
- <asp:ListItem>Poster</asp:ListItem> <asp:ListItem>Bilgisayar</asp:ListItem>
- <asp:ListItem>Kirtasiye</asp:ListItem>
- <asp:ListItem>DVD</asp:ListItem> </asp:CheckBoxList>
- <P>
- <asp:CheckBox id="gosterim" OnCheckedChanged="gosterim_bicimi_degisti" Text="Secenekleri Tablo halinede göster" Checked="true" AutoPostBack="true" runat="server" />

- <asp:CheckBox id="yon" OnCheckedChanged="yon_degisti" Text="Yatay Göster"
 AutoPostBack="true" runat="server" />
- <asp:Button id="Button1"Text="Gönder"onclick="Buton_tiklandi" runat="server"/>
- <P> <asp:Label id="Label1"font-name="Verdana"font-size="10pt"runat="server" />
- </div> </form> </body></html>

- using System;
- using System.Collections.Generic;
- using System.Linq;
- using System.Web;
- using System.Web.UI;
- using System.Web.UI.WebControls;
- public partial class _06_web_form_elemanlari_03_CheckBoxList : System.Web.UI.Page
- {

```
    protected void Buton_tiklandi(Object Sender, EventArgs e)
```

- {
- String secilen = "Sectikleriniz:
";
- for (int i = 0; i <= secenek_listesi.Items.Count 1; i++)</pre>
- {//For secenek_say = 0 to secenek_listesi.Items.Count-1
- if (secenek_listesi.Items[i].Selected)
- { secilen = secilen + secenek_listesi.Items[i].Text + "
; }

```
    Label1.Text = secilen;
```

```
• ]
```

```
• }
```

- protected void gosterim_bicimi_degisti (Object Sender, EventArgs e)
- {
- if (gosterim.Checked == true)
- { secenek_listesi.RepeatLayout = RepeatLayout.Table; }
- else
- { secenek_listesi.RepeatLayout = RepeatLayout.Flow; }
- }

```
    protected void yon_degisti(Object Sender, EventArgs e)
```

• {

```
if(yon.Checked==true)
```

```
• { secenek_listesi.RepeatDirection = RepeatDirection.Horizontal; }
```

• else

```
• { secenek_listesi.RepeatDirection = RepeatDirection.Vertical; }
```

- }
- }

- <asp:CheckBoxList runat=server> <asp:CheckBoxList>
- arasındaki değerler bir bütün olarak seçenek listemizi oluştururlar. Bu kontrolün önemli özellikleri vardır. Bunlardan bir tanesi RepeatLayout, seçeneklerin görüntülenme şeklini belirler. İki tane değer alır; Table_ve Flow.



- Diğer önemli bir özellik ise, elemanların sıralanma yönüdür. Bu özellikte iki değer alır: Horizontal ve vertical. Buna göre listedeki elemanların sayfada gösterilişi yatay veya dikey olarak belirlenebilir.
- CeckBoxList 'deki elemanların sayısını CheckBoxList_ID. Items. Count ile elde edebiliriz. Liste elemanlarının genel özelliği olan items, Count ile listedeki elemanların sayısını döndürür.
- Bir CheckBox'da kutunun işaretlenip işaretlenmediğini;
 - if (gösterim.Checked == true) ifadesi ile kontrol edebilirsiniz.
- CheckBoxList'de ise bunu:
 - **if (secenek_listesi.Items[no].Selected)** şeklinde bir ifadeyle kontrol edebilirsiniz.
- CheckBox'ın temel olay tanımı OnCheckedChanged 'dir. Bu olay ifadesi, seçenekte bir değişim olup olmadığını belirtir. Eğer değişim var ise olay gerçekleşir.
- CheckBoxList'te ise temel olay tanımı OnSelectedIndexChange 'dir. Burada ise listedeki! seçeneklerde bir değişim olması olayı gerçekleştirir.

	S			-	-	-	
	http://lo	calhost:9240/	WebSite4/CheckB	oxList.aspx	_		_
kitap	CD	Poster	✓ Bilgisavar	Kırtasive			
	1777 A						
✓ Seçenek	leri Tablo	halinede gö	ster				
✓ Yatay G	öster						
Gönder							
Seçtiklerini	z:						
Bilgisayar							

- Uygulama: Aşağıda verilen ekran görüntüsüne web sayafasını CheckBoxList özelliğini kullanarak yapınız.
- (<asp:CheckBoxList ID="CheckBoxList1" runat="server"> </asp:CheckBoxList>)

🙆 WebForm1 - Microsoft Internet Explorer	_ 🗆 🗙
<u>File E</u> dit <u>V</u> iew F <u>a</u> vorites <u>T</u> ools <u>H</u> elp	
🚱 Back 🔹 🕥 - 🥝 🥥 🌀 💭 Search	*
Address 🗃 http://localhost/WebApplication21/WebForm1.asp> 💌 💽 Go	Bağlantılar »
	~
Adımızı Giriniz Özay ÇELİK 🗹 Kalın	
Puntoyu Giriniz 8 Italic V AltÇizgili	
Uygula	
<u>Özav ÇELİK</u>	~
👩 Done 🙀 Local intra	anet .:

RadioButton, RadioButtonList

- RadioButton, CheckBox kontrolüne benzerlik gösterir.
- RadioButton kontrolunun GroupName özelliği ile, birden fazla RadioButton arasında grup oluşturulur.
- Aynı grup içerisinden sadece bir RadioButton seçilebilir. Birden fazla seçeneğin işaretlenmesine izin verilmez.
- Onay kutusunun durumu **Checked** metodu ile takip edilebilir.

- Örnekte, RadioButton kontrolleri arasında muzik isminde bir grup oluşturulmuştur.
- Bu grup içerisindeki Pop, Jazz ve Classic RadioButton kontrollerinden sadece bir tanesi seçilebilir.
- BtnOnay isimli button tıklandığında, seçilen RadioButton kontrolunun değeri Message isimli Label kontrolune yazılır.

- <%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>
- <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
- <html xmlns="http://www.w3.org/1999/xhtml" >
- <head runat="server"> <title> RadioButton</title>
- <script runat="server">
- void BtnOnay_Clicked(Object Sender, EventArgs e) {
- if (Radio1.Checked) { Message.InnerHtml = "Seçiminiz" + Radio1.Text; }
- else if (Radio2.Checked) { Message.InnerHtml = "Seçiminiz " + Radio2.Text; }
- else if (Radio3.Checked) { Message.InnerHtml = "Seçiminiz " + Radio3.Text; } }
- </script>
- </head> <body>
-

- <form id="form1" runat="server">
- <h4>Beğendiğiniz müzik türünü seçiniz:</h4>
- <asp:RadioButton id=Radio1 Text="Pop" Checked="True" GroupName="muzik" runat="server"/>
-

- <asp:RadioButton id=Radio2 Text="Jazz" GroupName="muzik" runat="server" />
-

- <asp:RadioButton id=Radio3 Text="Classic" GroupName="muzik" runat="server" />
-

- <asp:button Text="Seçiniz" id="BtnOnay" OnClick="BtnOnay_Clicked" runat="server" />
-

- </form></body> </html>

RadioButtonList

RadioButtonList.aspx

- <%@ Page Language= "C#" AutoEventWireup="true" CodeFile="RadioButtonList.aspx.cs" Inherits="RadioButtonList" %>
- <html><head runat="server"><title></title></head>
- <body bgcolor="#daeef3">
- <form id="form1" runat="server">
- <h3> RadioButton</h3>
- <h4>lütfen bir takım seçin:</h4>
- <asp:RadioButtonList id="radyo_liste" RepeatLayout="Table"
 RepeatColumns="2" runat="server">
- <asp:ListItem Selected="True">Galatasaray</asp:ListItem>
- <asp:ListItem>FenerBahçe</asp:ListItem>
- <asp:ListItem>Beşiktaş</asp:ListItem>
- <asp:ListItem>Tabzonspor</asp:ListItem>
- <asp:ListItem>Gaziantep</asp:ListItem>
- <asp:ListItem>Ankaragücü</asp:ListItem>
- </asp:RadioButtonList>

RadioButtonList

- <h4> Lütfen bir araba markası seçin: </h4>
- <asp:RadioButtonList RepeatLayout="Flow"
 RepeatDirection="Horizontal" id="radyo_liste_2" runat="server">
- <asp:ListItem>Mercedes</asp:ListItem>
- <asp:ListItem>BMW</asp:ListItem>
- <asp:ListItem>Volvo</asp:ListItem>
- <asp:ListItem>Honda</asp:ListItem>
- <asp:ListItem>Ford</asp:ListItem>
- <asp:ListItem>Mazda</asp:ListItem>
- </asp:RadioButtonList>
- •
- <asp:Button id="Button1" text="Gönder" OnClick="gonder_klik"
 runat="server"/>
- •
- <asp:Label ID="Label1" Font-Bold="true" runat="server"/>
- </form>
- </body> </html>

RadioButtonList

RadioButtonList.aspx.cs

- using System;
- using System.Web.UI;

}

- using System.Web.UI.WebControls;
- public partial class RadioButtonList : System.Web.UI.Page
- { protected void gonder_klik(object sender, EventArgs e)
- {Label1.Text = "Takım olarak" +radyo_liste.SelectedItem.Text;

```
    Label1.Text += "<br>> Araba Markası Olarak;" +
```

```
radyo_liste_2.SelectedItem.Text + " sectiniz.";
```

- •
- }
- RadioButtonList konrolünün oluşturulması ve yönetimi daha basittir.
 RadioButtonList'de, listedeki elemanların sayfada konumlandırılmasını, yatay, dikey, table veya flow olarak, RepeatDirection ve RepeatLayout özellikleri ile kolayca belirleyebiliyoruz veya Repetcolumns ile liste elamanlarının kaç sütunlu olarak görüntüleneceğini bildirebiliyoruz.

🕙 RadioButton - Microsoft Internet Explorer 📃 🗖	×
Dosya Düzen Görünüm Sık Kullanılanlar Araçlar 🎽 🧂	7
🌀 Geri 🝷 🐑 💌 😰 🏠 🔎 Ara	»
Adres 🕘 http://10.6.2.77/ 🔽 🎅 Git 🛛 Bağlantılar 🎽 📆 🕇	
Seçiminiz Classic Beğendiğiniz müzik türünü seçiniz :	<
 ○ Pop ○ Jazz ③ Classic Seçiniz 	III
	~
🙆 Bitti 🤍 🖉 Internet	

Uygulama: RadioButtonList'in özelliğini kullanarak aşağıda verilen uygulamayı yapınız.



- Hyperlink, sayfalar arası dolaşımı sağlar. Hyperlink kontrolünün görünümü metin veya resim olabilir.
 - ImageUrl özelliği ile görüntülenecek resim dosyası belirlenir.
 - NavigateUrl özelliği ile gidilecek sayfa belirlenir.
 - **Text** özelliği anchor text kısmına gelecek olan metni belirliyor. Yani linkte yer alacak tıklanacak metin buraya yazılır.
 - Target özelliği linke tıklandığında yeni sayfanın nasıl açılacağını belirlemek için kullanılır.
 - ToolTip özelliği önemli bir özelliktir. Linklere title ekleyecektir. Yani HTML kodunda yer alan title="" tagini ToolTip özelliği ile belirlenir. Text ve ImageUrl özelliklerinin her ikisi de ayarlanırsa resim linki önceliği alır, Tooltipe yazdığımız metin, imageurl özelliği kullanılmışsa sayfaya alt=""" şeklinde eklenecektir.

• Örnek

- HyperLink.aspx
- <%@ Page Language="C#" AutoEventWireup="true" CodeFile="HyperLink.aspx.cs" Inherits="HyperLink" %>
- <html xmlns="http://www.w3.org/1999/xhtml">
- <body>
- <form id="form1" runat="server">
- <asp:hyperlink ImageUrl="../resimler/xml_kapak.gif" id="Link1"
 Target="_blank" runat="server">
- XML/Zafer Demirkol
- </asp:hyperlink>
- </form>
- </body>
- </html>

- HyperLink.aspx.cs
- using System;
- using System.Collections.Generic;
- using System.Linq;
- using System.Web;
- using System.Web.UI;
- using System.Web.UI.WebControls;
- public partial class HyperLink: System.Web.UI.Page
- {
- protected void Page_Load(object sender, EventArgs e)
- {
- Link1.NavigateUrl = "http://www.kitaplar.net";
- }
-]

- <asp:HyperLink id="hyperlink1" runat="server"</p>
- ImageUrl="image1.gif"
- NavigateUrl="http://www.bilgeadam.com"
- Text="Bilge Adam"
- ToolTip = "Bilge Adam BTA"
- Target="_blank"/> eğer buraya ifade girilmemiş ve ImagURL yok ise Text değeri gelir </asp:HyperLink>

Target Özelliği	Açıklama
_blank	Yeni sayfa
_self	Aynı sayfa içinde
_search	Arama sayfası görünümünde

Image

- Image, sayfa içinde resim görüntülemek için kullanılır.
- ImageUrl özelliği ile görüntülenecek resim dosyası belirlenir.
- ImageAlign özelliği resmin hizalanması için kullanılır.
- AlternateText resime alternatif metin göstermek için kullanılır.
 - (Resim bulunamadığında gösterilecek ifade)
- Örnekte Image kullanımı gösterilmektedir.
- <asp:Image id="Image1" runat="server" AlternateText="Logomuz" ImageAlign="left" ImageUrl="logo.gif"/>

ImageButton

- Buton kontrolü gibi sunucuya postalama yapan kontroldür.
 Buton kontrolünden farkı, görüntü olarak istediğimiz resmi (*image*) kullanabilmemizi sağlamasıdır.
- void ImageButton_Click(object Source,ImageClickEventArgs e) {
- Message.InnerHtml="Resimli Düğme kontrolünü Tıkladınız" +
- "Koordinatlar: (" + e.X.ToString() + ", " + e.Y.ToString() + ")" ;
- }
- <asp:ImageButton id="imagebutton1" runat="server"</p>
- AlternateText="Resimli Düğme Kontrolü" ImageAlign="right"
- ImageUrl="image1.gif" OnClick="ImageButton_Click"/>
-

-
ImageButton

- <%@ Page Language="C#" AutoEventWireup="true" CodeFile="ImageButton.aspx.cs" Inherits="ImageButton" %>
- <html xmlns="http://www.w3.org/1999/xhtml">
- <head runat="server"><title></title></head>
- <body bgcolor="#daeef3">
- <h3> ImageButton ve Image Map şeklindeki kullanımı </h3>
- <form id="form1" runat="server">
- •
- •
- <asp:ImageButton ID="buton" ImageUrl="~/resimler/asp_kapak.gif"
 BorderWidth="2px" OnClick="ImageButon_klik" runat=server />
- <asp:Label ID="Label1" Text="X:" runat=server />
>

- <asp:Label ID="Label2" Text="Y:" runat=server />
- •
- <asp:Label ID="Label3" Text="" runat="server" />
- </form> </body> </html>

ImageButton

- public partial class ImageButton: System.Web.UI.Page
- {
- protected void ImageButon_klik(object sender, ImageClickEventArgs e)
- { int x = e.X;
- int y = e.Y;
- Label1.Text = "X: " + x.ToString();
- Label2.Text = "Y: " + y.ToString();
- if (y >= 47)
- Label3.Text = "Başlığın altını tıkladın"; }

```
• else
```

}

```
{ Label3.Text = "<b>Başlık</b> kısmını tıkladın!"; }
```

- •
- }

LinkButton

- LinkButton, HyperLink görünümlü Button kontrolüdür.
- LinkButton kontrolünün HyperLink kontrolünden farkı ise olaylarının olmasıdır.
- void LinkButton1_Click(Object sender, EventArgs e)
- { Label1.Text="Link Button'a tıkladınız"; }
- •
- <asp:LinkButton Text="Mesajı Görmek İçin Tıklayınız."
- Font-Name="Verdana" Font-Size="14pt" onclick="LinkButton1_Click" runat="server"/>
-
 <asp:Label id=Label1 runat=server />

- DropDownList, açılan kutuda veri görüntülemek için kullanılır.
 Bu listeden seçilen değerlere göre işlem yapılır.
- DropDownList öğeleri Items koleksiyonunda tutulur.
- Items koleksiyonunun Count özelliği ile toplam öğe sayısı bulunur.
- DropDownList kontrolune tasarım veya çalışma zamanında öğe eklenebilir.

- void Button_Click(object sender, EventArgs e) {
- Label1.Text = "Konuştuğunuz Dil " + dropdownlist1.SelectedItem.Text + "."; }
- <asp:DropDownList id="dropdownlist1" runat="server">
- <asp:ListItem>Türkçe</asp:ListItem>
- <asp:ListItem>İngilizce</asp:ListItem>
- <asp:ListItem>Almanca</asp:ListItem>
- <asp:ListItem>İtalyanca</asp:ListItem>
- </asp:DropDownList>
- <asp:Button id="Button1" Text="Submit" OnClick="Button_Click" runat="server"/>
- <asp:label id="Label1" runat="server"/>

- ListItem etiketi içindeki değerler, DropDownList öğelerini temsil eder.
- Örnekte DropDownList kontrolune çalışma zamanında öğe eklenmektedir.
- <asp:DropDownList id="DropDownList1" style="Z-INDEX: 101; LEFT: 128px; POSITION: absolute; TOP: 160px" runat="server" Width="152px"/>
- void Page_Load(System.Object sender, System.EventArgs e)
- {
- for (int i=0;i<=5;i++)
- The second state of the
- }
- Çalışma zamanında eleman eklemek için, Items koleksiyonunun Add metodu kullanılır.

- private void tikla(object sender, System.EventArgs e)
- Label3.Text = TextBox1.Text;
- Label3.Font.Size = Convert.ToInt32(Textbox2.Text); }
- private void secim(object sender, System.EventArgs e)
- {
 if (DropDownList1.SelectedItem.Text == "Bold")
- Label3.Font.Bold = true; }
- if (DropDownList1.SelectedItem.Text == "Italic")
- Label3.Font.Italic = true;
- if (DropDownList1.SelectedItem.Text == "Altcizgili")
- Label3.Font.Underline = true; }

 DropDownList kontrolünün "OnSelectedIndexChanged" olayını kullanılarak secim metodu çalıştırılır ve yazının fontu ayarlanır.

🧃 WebForm1 - Mic	rosoft Interne	t Explorer		_ 🗆 🔀
<u>File E</u> dit <u>V</u> iew Fa	avorites <u>T</u> ools	<u>H</u> elp		
🚱 Back 🔹 🥥	- 🔕	0	Search	»
Address 👩 http://loca	lhost/WebApplicat	ion21/WebForm	1.aspx 💌 💽 Go	Bağlantılar 🎽
				<u>~</u>
Adınızı Giriniz	Zülfü ALAN	OĞLU		
Punto	20			
Font	Bold	~		
Button				
<u>Zülfü</u>				
<u>ALANO</u>	<u> OGLU</u>			~
👩 Done			👊 Local intra	anet 🔐

- ListBox, DropDownList kontrolüne benzer.
- Elemanlar liste halinde gösterilir ve SelectionMode özelliğine Multiple değeri atanarak, çoklu seçim yapma imkânı sağlanır. Elaman silmek için
 - ListBox1.Items.Clear();
 - ListBox1.Items.Remove(listItem);
 - ListBox1.Items.RemoveAt(int)
- Örnekte ListBox kontrolunün çoklu seçim özelliği kullanılmaktadır. ListBox kontrolu içerisinde seçilen tüm elemanlar Label kontrolune yazıdırılır.

- public void SubmitBtn_Click(object sender, EventArgs e)
- { Message. InnerHtml= "";
- foreach (ListItem i in ListBox1.Items)

```
{ if (i.Selected == true) { Message. InnerHtml += i.Text + " "; }
```

- } }
- <asp:ListBox id=ListBox1 Rows=4 SelectionMode="Multiple" Width="100px" runat="server">
- <asp:ListItem>Türkçe</asp:ListItem>
- <asp:ListItem>İngilizce</asp:ListItem>
- <asp:ListItem>Almanca</asp:ListItem>
- <asp:ListItem>İtalyanca</asp:ListItem>
- </asp:ListBox>
- <asp:button ID="Submit" Text="Submit" OnClick="SubmitBtn_Click"</p>
- runat="server" />
- <asp:Label id="Message" runat="server"/>

🕙 ListBox - Microsoft Internet Explorer -» <u>S</u>ik Kullanılanlar Düzen <u>Görünüm</u> <u>A</u>raçlar <u>D</u>osya 1 » 2 🏠 × Ara 🧲 Geri 🝷 Adres 🗿 http://10.6.2.77/V 💙 Bağlantılar 🌺 🔁 Git 91 Ŧ Türkçe İngilizce Almanca İtalyanca Submit Türkçe 🕘 Bitti 🥝 Internet

• Uygulama:

👩 WebForm1 - Micro	osoft Internet Explorer	_ 🗆 🔀
<u>File E</u> dit <u>V</u> iew F <u>a</u> v	rorites <u>T</u> ools <u>H</u> elp	
🚱 Back 🔹 🕥	- 3 3 3	>>> Search
Address 👩 http://localh	ost/WebApplication21/WebForm1.as	ipx 💌 💽 Go 🛛 Bağlantılar 🎽
Adınızı Giriniz	Özay ÇELİK	
Punto	10	
Font	Courier New Arial Arial Y	
	Button	
Özay ÇELİK		
		~
👩 Done		🗣 Local intranet 🛒

Panel

- Kontrolleri, sayfada programatik olarak oluşturmamıza yardımcı olan bir kontroldür.
- Panel kontrolü, içinde diğer kontrol veya kontrolleri barındırır ve bunları sayfada görüntüler. Herhangi bir sunucu kontrolünü, sayfada programatik olarak oluşturmak demek kontrolü kod kısmında tanımlayıp panel kontrolüne ekleyerek görüntülemek demektir.
- Panel'in diğer bir avantajı içinde barındırdığı elemanları bir grup olarak görünür veya görünmez yapabiliriz.

Panel

- <%@ Page Language="C#" AutoEventWireup="true" CodeFile="panel.aspx.cs" Inherits="Panel" %>
- <html xmlns="http://www.w3.org/1999/xhtml">
- <head runat="server"> <title></title> </head>
- <body> <form id="form1" runat="server">
- <asp:Panel ID="Panel1" runat="server" BackColor="#ccffcc" Height="120px" Width="200px"
- Font-Name= "verdana" Font-Bold="True" Font-Size="16px">
- Panel Örneği
- </asp:Panel>
- <asp:CheckBox ID="gizle" Text="Paneli gizle" AutoPostBack="True" OnCheckedChanged="panel_gizle"
- runat="server"></asp:CheckBox>
- </form> </body> </html>

	(←) → @ http://localhost: ♀ - ≧ Ċ × @ localhost
Panel	Panel Örneği
• using System:	gonder
 using System, Collections.Generic; 	
 using System.Linq; 	Paneli gizle
• using System.Web;	
 using System.Web.UI; 	🔩 %100 🔻
 using System.Web.UI.WebControls; 	
• public partial class Panel : System.Web.UI.Pa	ge
<pre>• { protected void Page_Load(object sender,</pre>	EventArgs e)
<pre>• { TextBox metin_kutusu = new TextBox();</pre>	
 Panel1.Controls.Add(metin_kutusu); 	
 Panel1.Controls.Add(new LiteralContro 	l(" "));
<pre>Button buton = new Button();</pre>	
 Panel1.Controls.Add(buton); 	
<pre>• buton.Text = "gonder";// buton.id="po</pre>	<pre>stala_1";</pre>
 Panel1.Controls.Add(new LiteralContro 	l(" "));
• }	
 protected void panel_gizle(object nesne, 	EventArgs e)
• { if (gizle.Checked) { Panel1.Vis	ible = false: }
$\int D_{2} = \int D_{2} = \frac{1}{2} $	··, ,
	ſ
• }	

Panel

Uygulama: Arka arkaya 10 paneli İleri ve Geri butonları ile görüntüleyiniz.

Table, TabloRow, TableCell

- Table, satırlarına ve sütunlarına programlama yoluyla müdahale edilebilen tablo kontroldür. Tek bir , veya herhangi bir HTML tablo tag'ı yazmadan, uygulamanızda tablo oluşturmayı sağlar.
- Table kontrolü içinde **TableRow** ve **TableCell** nesneleri kullanılır.
- TableCell, tabloda bir hücreyi temsil eder.
- TableRow ise tabloda bir satırı temsil eder.

Table

- protected void Page_Load(object sender, EventArgs e)
- { //Satır ve Sütun Oluşumu
- int nrows = 3, ncells = 2, i, j;
- for (j = 0; j <= nrows 1; j++)
- •
- TableRow r; r = new TableRow();
- for (i = 1; i <= ncells; i++)
- TableCell c; c = new TableCell();
- c.Controls.Add(new LiteralControl ("Satır " + j.ToString() + ", hücre " + i.ToString()));
- r.Cells.Add(c);
- Table1.Rows.Add(r);

 //LiteralControl Label etiketi yerine tercih edilen bir kontroldür. Html komutlarını ve birleştirme ve dönüştürme komutlarını kullanmaya imkan tanır. new LiteralControl ("
)

^{• }}

Table

 <asp:Table id="Table1" GridLines="Both" HorizontalAlign="Center" Font-Name="Verdana" Font-Size="8pt" CellPadding=15 CellSpacing=0 Runat="server"/>



Table

• Uygulama:

e () () () () () () () () () () () () ()	/localhost:2396/Tal 🎗	0 - 🗟 C X 遵 lo	calhost	× ∩ x	x 7 {
satır 01, hücre 01 satır 11, hücre 01 satır 21, hücre 01	satır 01, hücre 11 satır 11, hücre 11 satır 21, hücre 11	satır 01, hücre 21 satır 11, hücre 21 satır 21, hücre 21	satır 01, hücre 31 satır 11, hücre 31 satır 21, hücre 31		
satır 31, hücre 01	satır 31, hücre 11	satır 31, hücre 21	satır 31, hücre 31		
Tablo Satırları: 4	Tablo hücreleri:	1 2 3 4			
				ه %100	•

Placeholder

 PlaceHolder kontrolü, Runtime esnasında oluşturulan diğer kontrol elemanlarını barındırır. Bu anlamda Panel kontrolüne çok benzer. Diğer kontrollerin aksine, görünür bir arayüz birimi yoktur. Sadece kendisine, program esnasında eklenen diğer kontrolleri sayfada gösterir.

ÖRNEK

- PlaceHolder.aspx
- <%@ Page Language="C#" AutoEventWireup="true" CodeFile="PlaceHolder.aspx.cs" Inherits="PlaceHolder" %>
- <html xmlns="http://www.w3.org/1999/xhtml">
- <body bgcolor="#daeef3">
- <form id="form1" runat="server">
- <asp:PlaceHolder ID="ph" runat="server" />
- </form></body></html>

Placeholder

- PlaceHolder.aspx.cs
- public partial class PlaceHolder : System.Web.UI.Page
- { protected void Page_Load(object sender, EventArgs e)
- ph.Controls.Add(new LiteralControl("isim: "));
 - TextBox metinkutusu = new TextBox();
- //metinkutusu.Id="ad" ;
- ph.Controls.Add(metinkutusu);
 - ph.Controls.Add(new LiteralControl(""));
- Button buton = new Button();
 - buton.Text = "gonder";
 - //buton.Id="gndr" ; ph.Controls.Add(buton);

• }

}

- Kullanıcının girdiği değerleri kontrol etmek için kullanılır.
 Kontrolün yapılacağı alana ve veriye göre, farklı
 Validation doğrulama kontrolleri kullanılır.
- ASP.NET, belirli bir aralıkta veri girişi sağlayan, karşılaştırma yapan ve belirli değerlerin boş geçilmemesini sağlayan çeşitli doğrulama kontrolleri sunar.
- Validation Kontrolleri kullanım amacına göre ayrılmıştır. 5 çeşit Validation geçerlilik kontrolü bulunmaktadır. Bu 5 farklı kontrol dışında birde ValidationSummary kontrolü var. Bu kontrol aslında bir geçerlilik kontrolünde ziyade, diğer geçerlilik kontrollerinden gelen hata mesajlarını özet olarak vermektedir.

R	Pointer
₽,	CompareValidator
₹	CustomValidator
	RangeValidator
×*	RegularExpressionValidator
	RequiredFieldValidator
	ValidationSummary

VALIDATION KONTROLLER	GÖREVİ		
RequiredFieldValidator	Bir kontrol içerisine değer girilip girilmediğini kontrol eder. Veri girilmesi zorunlu alanlarda kullanılır.		
CompareValidator	Kontrol içerisine girilen değeri, sabit değerle veya başka bir kontrole girilen değerle karşılaştırır.		
RangeValidator	Kontrol içerisine girilen değerin, iki sabit değer arasında olmasını sağlar.		
RegularExpressionValidator	Bir kontrol içerisine girilen değerin, istenilen formatta girilmesini sağlar.		
CustomValidator	Özel doğrulama kontrolü yazmayı sağlar.		
ValidationSummary	Sayfada kullanılan tüm validation kontrollerin, doğrulama hatalarını özet olarak görüntüler.		

- Doğrulama kontrollerinin ortak özellikleri aşağıdaki gibidir.
 - **ControlToValidate:** Hangi kontrolün doğrulanacağını belirtir. (Hangi form elemanı kontrol edilecekse o formun ID si yazılır)
 - ErrorMessage: Geçerli giriş yapılmamışsa görüntülenecek hata mesajını verir.
 - Text: ErrorMessage ve Text özelliği birlikte kullanılabilir. Bu durumunda Text özelliğindeki mesaj görüntülenir. Doğrulama kontrollerin ErrorMessage özelliğine girilen tüm mesajlar ValidationSummary içerisinde listelenir.

- Doğrulama kontrollerinin ortak özellikleri aşağıdaki gibidir.
 - Display: Validation kontrolün nasıl görüntüleneceği bilgisini tutar. Static, Dynamic ve None değerleri alır. Eğer birden fazla validation kontrol eklenecekse; bu Display seçeneğinin Dynamic olması gerekir. Validator'ın bu özelliği sayesinde; yanyana eklenmiş validator'ların hemen input kontrolün yanında çıkması sağlanır. Statik olursa; nerede yazılmış ise, orada çıkar. Default olarak Display özelliği Static'tir. Aynı kontrol içinde birden fazla validator kullanıldığı zaman, Dynamic özelliğine çevirmeyi unutmayınız. None kullanılırsa error mesajı için blok koyar.

RequiredFieldValidator

- RequiredFieldValidator, belirtilen kontrolün boş geçilmemesini sağlar.
- Doğrulama yapılacak web kontrolünün ismi
 ControlToValidate özelliğine girilir Geçerli giriş
 yapılmadığında ortaya çıkacak hata mesajı ErrorMessage
 özelliği ile belirtilir.

- Örnekte RequiredFieldValidator kullanımı gösterilmektedir.
- <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
- <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
- ControlToValidate="TextBox1" ErrorMessage="Boş Girilemez">
- </asp:RequiredFieldValidator>



CompareValidator

- Kontrol içerisine girilen değeri, sabit değerle veya başka bir kontrol ile karşılaştırmak için kullanılır.
- Doğrulama yapılacak web kontrolünün ismi ControlToValidate özelliğine girilir.
- Karşılastırma yapılacak sabit değer ValueToCompare özelliğine girilir.
- **Type** özelliğine girilen değerin veri türü,
- **Operator** özelliğine ise mantıksal operatör girilir.
- Örnekte txtYas kontrolune yirmi veya yirmiden büyük tamsayı girişi sağlayan doğrulama işlemi yapılmaktadır.

- <asp:CompareValidator id="CompareValidator1"</p>
- style="Z-INDEX: 109; LEFT: 232px; POSITION: absolute;
- TOP: 88px" runat="server"
- ErrorMessage="Yaşınız 20 ye eşit veya büyük olmalıdır."
- ValueToCompare="20"
- ControlToValidate="txtYas"
- Type="Integer"
- Operator="GreaterThanEqual"
- Width="128px">
- </asp:CompareValidator>

 Doğrulama yapılacak web kontrolu, başka bir web kontrolü ile karşılaştırılacaksa ControlToCompare özelliği kullanılır.

Operatör	Açıklama
DataTypeCheck	Sadece veri türünü karşılaştırır
Equal	Eşittir
GreaterThan	Büyük
GreaterThanEqual	Büyük eşit
LessThan	Küçük
LessThanEqual	Küçük eşit
NotEqual	Eşit değil

- <asp:CompareValidator id="CompareValidator1"</p>
- style="Z-INDEX: 109; LEFT: 240px; POSITION: absolute;
- TOP: 128px" runat="server" Width="144px"
- ErrorMessage="yaşınız kontrol alanında yazılan
- değerden büyük olmalıdır."
- ControlToValidate="txtYas"
- Type="Integer"
- Operator="GreaterThan"
- ControlToCompare="txtKontrol">
- </asp:CompareValidator>

RangeValidator

- Kontrol içerisine girilen değerin, belirli bir değer aralığında olmasını sağlar. Doğrulama kontrollerinin ortak özelliklerine ek olarak MinimumValue, MaximumValue ve Type özellikleri vardır.
- Örnekte txtYas kontrolune girilen değerin, otuzbeş ile elli arasında olamasını sağlayan doğrulama işlemi yapılmaktadır.

RangeValidator

- <asp:RangeValidator id="RangeValidator1"</p>
- style="Z-INDEX: 109; LEFT: 240px; POSITION:
- absolute; TOP: 128px" runat="server"
- ErrorMessage="Yaşınız 35 ile 50 arasında olmalıdır."
- ControlToValidate="txtYas"
- Type="Integer"
- MaximumValue="50"
- MinimumValue="35">
- </asp:RangeValidator>

RegularExpressionValidator

- Kontrol içerisine girilen değerin, istenilen formatta girilmesini sağlar.
- ValidationExpression özelliğine girilen özel karakterler ile veri giriş formatı sağlanır.
- Bu özel karakterler Tablo da gösterilmektedir.

RegularExpressionValidator

	Pro	perties		X
	Re	gularExpressionVal	idator1 System.Web.UI	l
		≜↓ 🗉 🖋 🖻]	
Regular Expression Editor		ErrorMessage	E-posta adresi hatalı	*
Standard expressions:	⊳	Font		
German portal code		ForeColor	Red	
Internet e-mail address		Height		
Internet URL		SetFocusOnError	False	
Japanese phone number		SkinID		
Japanese postal code 🔹 🔻		TabIndex	0	
Validation expression:		Text		
Validation expression.		ToolTip		Ξ
\w+([-+.]\w+)"@\w+([]\w+)"\.\w+([]\w+)"		ValidationExpression	2	
		ValidationGroup		
OK		ViewStateMode	Inherit	
		Vicible	Тпо	· •
	Va	lidationExpression		
	Reg	gular expression to o	determine validity.	
Karakter	Tanımı			
----------	--			
а	Bir harf kullanımını zorunlu kılar.			
1	1 sayısı kullanılmak zorunda.			
?	0 veya 1 öğe olmak zorunda			
*	0'dan n'e kadar bir değer			
+	1'den n'e kadar bir değer			
[0-n]	0'dan n'e kadar sayı değer dizisi			
{n}	N ile belirtilen değer uzunluğunda olmalı			
	Farklı geçerli dizinler.			
1	Bir komut karakterini devam ettiren karakter			
\w	Bir karakter olmak zorunda.			
\d	Bir rakam olmak zorunda.			
۱.	Bir nokta olmak zorunda.			

- ValidationExpression= "\w+([-+.]\w+)*@\w+([-.]\w+)*\.\w+([-.]\w+)*">
- **\w+ :** En az bir karakter içeren metin anlamına gelir.
- ([-+.]): -, +, . karakterlerinden herhangi biri anlamına gelir.
- * : 0'dan n'e kadar bir değer girilmesi gerektiği anlamına gelir.
- @ : @ işaretinin kullanılması gerektiğini belirtir.
- Örnekte ValidationExpression özelliğine girilen özel karakterler ile e-mail formatı oluşturulmaktadır.
- ValidationExpression ="\w+@\w+\.\w+"
- ValidationExpression="\d{5}" 5 basamaklı sayı girilecek

Doğrulama türü	Geçerlilik ifadesi
E-posta	\w+([-+.]\w+)*@\w+([]\w+)*\. \w+([]\w+)
Telefon numarası	\d{1}-\d{3}-\d{3}-\d{4}
URL	http(s)?://([\w-]+\.)+[\w-]+(/[\w/?%&=]*)?

- Örnekte RegularExpressionValidator kontrolu ile txtMail metin kutusu için geçerli e-mail girişi sağlanmaktadır.
- <asp:regularexpressionvalidator id="RegularExpressionValidator1" style="Z-INDEX: 111; LEFT: 256px; POSITION: absolute; TOP: 168px" runat="server"
- ErrorMessage="Mail giriş hatası"
- ControlToValidate="txtMail"
- ValidationExpression="\w+([-+.]\w+)*@\w+([-.]\w+)*\.\w+([-.]\w+)*">
- </asp:regularexpressionvalidator>

CustomValidator

 Aynı anda birden fazla nesnenin değerini kontrol etmek veya kullanıcı tanımlı kontrol yazmak için CustomValidator kontrolü kullanılır.

Özellik	Açıklama				
ClientValidationFunction	İstemci fonksiyon ismini belirtir.				
ControlToValidate	Doğrulama yapılacak kontrolü belirler.				
Display	Text özelliğine girilen hata mesajının nasıl görüntüleneceği				
	belirtir.				
EnableClientScript	İstemci skriptleri aktif hale getirir. Varsayılan değer True dur.				
Enabled	Sunucu ve istemci taraflı skriptleri aktif hale getirir. Varsayılan				
	değer True dur.				
ErrorMessage	Kontrol hata mesajını gösterir.				
IsValid	Kontrol işlemi başarı ile sonuçlanmışsa True, değilse False				
	değerini döndürür.				
Text	Kontrol hata mesajını gösterir. ErrorMessage ve Text özelliği				
	birlikte kullanılabilir. Bu durumunda Text özelliğindeki mesaj				
	görüntülenir.				

- ServerValidate olayı ve OnServerValidate metodu ile sunucu taraflı kontroller tetiklenir.
- Örnekte sunucu taraflı metod oluşturulmaktadır.
- public void CustomValidator_SunucuKontrol(object s,
- ServerValidateEventArgs e)
- { }
- ServerValidateEventArgs parametresinin iki özelliği vardır:
- IsValid: Bu özellik True ise kontrol içerisine girilen değerin doğruluğu sağlanmıştır.
- Value: Doğrulama kontrolünün değerini verir.

- CustomValidator, kredi kart numaralarının doğruluğu kontrol etmek için kullanılabilir.
- Örnekte metin kutusuna girilen değerin çift olması kontrol edilmektedir. CustomValidator kontrolunun HTML kodları aşağıdaki gibidir.

- <%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>
- <html> <head>

function ClientValidate(source, arguments) {

```
if (arguments.Value % 2 == 0) {
```

```
arguments.IsValid = true;
```

```
} else {
```

```
arguments.IsValid = false;
```

</script>

}

}

- <script runat="server">
- void ValidateBtn_OnClick(object sender, EventArgs e)
- { if (Page.IsValid) { Message.Text = "Page is valid.";}
 - else { Message.Text = "Page is not valid!";}

- void ServerValidation(object source, ServerValidateEventArgs args)
- { try {
- int i = int.Parse(args.Value);
- args.IsValid = ((i%2) == 0);

catch(Exception ex) { args.IsValid = false;

}

</script>

}

- </head>
- <body> <form id="Form1" runat="server">
- <h3>CustomValidator ServerValidate Example</h3>
- <asp:Label id="Message" runat="server" Text="Enter an even number:"</pre>
- Font-Name="Verdana" Font-Size="10pt" />
 - <asp:TextBox id="Text1" runat="server" />

Iocalhost

•	<asp:customvalidator <br="" id="CustomValidator1">ControlToValidate="Text1" ClientValidationFunction="ClientValidate" OnServerValidate="ServerValidation"</asp:customvalidator>	← → Ů Web adresi ara veya g □☆ … CustomValidator ServerValidate Example Enter an even number: ✓ ✓ ✓ ✓ ✓
•	Display="Static" ErrorMessage="Not an even number!"	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$
•	ForeColor="green" Font-Name="verdana" Font-Size="10pt" runat="server"/>	CustomValidator ServerValidate Example Enter an even number: 5 Not an even number! Validate localhost × + ×
	<asp:button <br="" id="Button1">Text="Validate" OnClick="ValidateBtn_OnClick" runat="server"/> </asp:button>	 ← Ů localhost:18385 □ ☆ ··· CustomValidator ServerValidate Example Page is valid. 8 Validate

ValidationGroup

- Doğrulama kontrolleri sayfa içerisine eklenen buton sayesinde hepsi kontrol edilmektedir. Eğer doğrulama kontrollerini tek bir butona değil birden fazla buton ile değerlendirilmeye alınacaksa ise ValidationGroup özelliği kullanılmalıdır.
- Bu sayede her doğrulama grubu, diğer doğrulama grupları sayfasında bağımsız olarak doğrulanabilir.

- <%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>
- <!DOCTYPE html>
- <html> <head id="head1" runat="server">
- <title>Button.ValidationGroup Örneği</title> </head>
- <body> <form id="form1" runat="server"> <h3>Button.ValidationGroup Örneği</h3>
- <asp:label id="AdLabel" text="Adınızı Giriniz:" runat="Server"
 AssociatedControlID="AdTextBox"> </asp:label>
- <asp:textbox id="AdTextBox" runat="Server"> </asp:textbox>
- <asp:requiredfieldvalidator id="RequiredFieldValidator1" controltovalidate="AdTextBox" validationgroup="PersonalInfoGroup" errormessage="Adınızı Giriniz." runat="Server"> </asp:requiredfieldvalidator>
- <asp:label id="YasLabel" text="Yaşınızı giriniz:" runat="Server"
 AssociatedControlID="YasTextBox"> </asp:label>
- <asp:textbox id="AgeTextBox" runat="Server"> </asp:textbox>

- <asp:requiredfieldvalidator id="RequiredFieldValidator2" controltovalidate="YasTextBox" validationgroup="PersonalInfoGroup" errormessage="Yaşınızı girmediniz." runat="Server"> </asp:requiredfieldvalidator>
-

 <!--Eğer Button1 tıklanırsa sadece PersonalInfoGroup validation kontrolü olan doğrulama kontrolleri denetlenir-->
- <asp:button id="Button1" text="Değerlendir" causesvalidation="true"
 validationgroup="PersonalInfoGroup" runat="Server" />

- <asp:label id="ilinizLabel" text="İkametgah İli:" runat="Server"
 AssociatedControlID="ilinizTextBox"> </asp:label>
- <asp:textbox id="İlinizTextBox" runat="Server"> </asp:textbox>
 <asp:requiredfieldvalidator id="RequiredFieldValidator3"
 controltovalidate="İlinizTextBox" validationgroup="LocationInfoGroup"
 errormessage="İl ismini giriniz." runat="Server"> </asp:requiredfieldvalidator>
-

 <!-- Eğer Button2 tıklanırsa sadece LocationInfoGroup validation kontrolü olan doğrulama kontrolleri denetlenir.-->
- <asp:button id="Button2" text="Validate" causesvalidation="true"
 validationgroup="LocationInfoGroup" runat="Server" /> </form> </body> </html>

ValidationSummary

- ValidationSummary, doğrulama kontrollerin ErrorMessage özelliğine girilen tüm mesajları listeler.
- Örnekte ValidationSummary kullanımı gösterilmektedir.
- •
- <asp:ValidationSummary id="ValidationSummary1" style="Z-INDEX: 114; LEFT: 72px; POSITION: absolute; TOP: 336px" runat="server" HeaderText="Hatalar">
- </asp:ValidationSummary>
- •
- DisplayMode özelliği ile ValidationSummary kontrolunun görüntüsü değiştirilebilir.
 DisplayMode, BulletList (simgeli madde imi), List ve SingleParagraph değerlerini alır. ShowMessageBox, hata listesinin mesaj kutusu içinde görüntülenmesini sağlar.

Doğrulama kontrolleri Netframe 4.5 den sonra jquery eklentisini istiyor.
 Eğer direk kullanırsanız aşağıdaki hatayı alırsınız.



- Çözüm için web.config veya global.asax dosyasında aşağıdaki satırları ekleyiniz.
- Çözüm 1: web.config
- <?xml version="1.0"?>
- <configuration>
- <appSettings>
- <add key="ValidationSettings:UnobtrusiveValidationMode" value="none"></add>
- </appSettings>
 - <system.web>
- <compilation debug="true" targetFramework="4.5" />
- <httpRuntime targetFramework="4.5" />
- </system.web>
- </configuration>

Çözüm 2 : global.asax <%@ Application Language="C#" %> <script runat="server"> void Application_Start(object sender, EventArgs e) ScriptManager.ScriptResourceMapping.AddDefinition("jquery", **new** ScriptResourceDefinition Path = "~/script/jquery-2.1.4.js", DebugPath = "~/script/jquery-2.1.4.js", CdnPath = "~/script/jquery-2.1.4.js", CdnDebugPath = "~/script/jquery-2.1.4.js" });

}

FORM GEÇERLİLİK DENETİMİ

• Örnek: Validator kontrollerini kullanarak istemci taralı geçerlilik denetimi

• Özellik:	kulaniciadivalidator	sifrevalidator
•		
ControlToValidate	TextBox1	TextBox2
•		
 ErrorMessage 	Kullanıcı adı hatalı.	Şifre hatalı
•	Kullanıcı adı 3 10 karakter arasında ve	
•	Yalnızca harf ve rakamdan oluşmalıdır.	
 ValidationExpression 	[a-zA-Z_0-9]{3,10}	.{6,10}
•		
 Display 	Dynamic	Dynamic
•		
EnableClientScript	true	true

۲

FORM GEÇERLİLİK DENETİMİ

🕙 Dinamik Resim Örneği - Microsoft Internet Explorer	_ 🗆 🔀
<u>File E</u> dit <u>V</u> iew F <u>a</u> vorites <u>T</u> ools <u>H</u> elp	A
🕝 Back 🔹 📀 👻 📓 🚮 🔎 Search 🤺 Favorites 🔇 Media 🤗 🔗	• 🎍 🏻 »
Address 🕘 http://localhost/Projem/Projem1/WebForm1.aspx 🛛 🍡 Go	Bağlantılar 🎽
	<u>^</u>
Kullanıcı Adı :	
Şitre :	
İptal Tamam	
	~
🙆 Done 🧐 Local intr	anet 🥁

FORM GEÇERLİLİK DENETİMİ

🕘 Dinar	nik Resim	n Örneği -	Microso	oft Internet Exp	olorer					IX
<u>File E</u> d	it <u>⊻</u> iew	F <u>a</u> vorites	<u>T</u> ools	Help						
G Bac	k - 🕑) - 🔀	2 🦿	Search 🔎	☆ Favorites	😢 Media	Ø	8 -	🎍 📄	»
A <u>d</u> dress	🕘 http://k	ocalhost/Pro	jem/Proje	m1/WebForm1.asp	x		~	🔁 🕞	Bağlantı	ar »
										^
				<i>"</i>						
	Kullanici	Adı :		OZAY®						
	Şifre	:		1543						
			ptal			Tamam				
	Kullanıc Şifre Ha	ı adı Hata talı	h. kullan	ucı adı 3-10 ka	rakter arasınd	a ve yanlızı	a raka	amdan ol	luşmalıdır.	~
<										>
ど Done							Sec. 1	🚽 Local int	ranet	

 Aşağıda verilen sayfa bu kontrollerin kullanımı için uygulamayı yapınız.

Adı	Adınızı girmediniz
Soyadı	Soyadınızı girmediniz
E-Posta	E-posta adresini girmediniz
Şifire	Şifrenizi girmediniz
Şifre Tekrar	Şifre tekrarını girmediniz
KAYDET	



AdRotator

- AdRotator, Web sayfaları üzerinde reklam yayını yapmak için kullanılır. Reklam için kullanılan banner nesneleri XML veya txt dosya içerisine kaydedilir.
- AdvertisementFile özelliği, reklâm bilgilerinin bulunduğu XML veya txt dosyayı içerir.
- KeywordFilter özelliği ise reklâmların filtrelenerek (Keyworda göre) görüntülenmesini sağlar.

AdRotator Özellikeri

- Ad: Her bir banner nesnesini temsil eder.
- **ImageUrl:** Resmin url yada path bilgisidir. Web dizinindeki alt yada üst klasörde olabilir.
- NavigateUrl: Banner yada resme tıklanıldığında, veyahut resim yüklenmez ise Alternatetext e tıklanıldığında gönderilecek web (url) adresi.
- AlternateText: Resmin üzerine gelindiğinde çıkan yada resim yüklenmediğinde resmin olması gereken yerde görünen bilgilendirici yazıdır.
- Impression: Gösterim sıklığıdır. Hangi reklamın impressions sayısı fazla ise, o reklam daha fazla görüntülenir.
- Keyword: Asıl amacı birçok kategoride reklam alan web sitelerinin kategori bazında filtreleme yapmasına olanak sağlamaktır. İlgili reklamın içerdiği anahtar kelimeler bu alana yazılabilir.

```
XMLFile.xml* → ×
                    Default.aspx.cs
                                       Default.aspx*
     <?xml version="1.0" encoding="utf-8" ?>
   ¬<Advertisements>
     <Ad>
   É
         <ImageUrl>cs_logo.gif</ImageUrl>
         <NavigateUrl>http://www.csharpnedir.com</NavigateUrl>
         <AlternateText> Türkçe C# ve .NET kaynağı
                                                       </AlternateText>
         <Keyword>C, C++, C#, .NET, ASP.NET, ADO.NET </Keyword>
         <Impressions>70</Impressions>
         <EkAlan>C#nedir.com desek yeter galiba </EkAlan>
       </Ad>
   ⊢ <Ad>
         <ImageUrl>msakademik.gif</ImageUrl>
         <NavigateUrl>http://www.msakademik.net</NavigateUrl>
         <AlternateText> Akademik dünya için .NET </AlternateText>
         <Keyword>Akademik, Üniversite, Yazılım geliştirme</Keyword>
         <Impressions>50</Impressions>
         <EkAlan> Akademisyenler ve Öğrencilerin .NET Sitesi </EkAlan>
       \langle Ad \rangle
   ⊢ <Ad>
         <ImageUrl>yage.gif</ImageUrl>
         <NavigateUrl>http://www.yazgelistir.com</NavigateUrl>
         <AlternateText> Microsoft Türkiye Yazılım gelişricileri gurubu </AlternateText>
         <Keyword>Ms, Microsoft, Yazılım Geliştirme </Keyword>
         <Impressions>45</Impressions>
         <EkAlan>Prefesyonel Yazılımcıların Adresi </EkAlan>
       \langle Ad \rangle
   ⊢ <Ad>
         <ImageUrl>ASPnedir.gif</ImageUrl>
         <NavigateUrl>http://www.aspnedir.com</NavigateUrl>
         <AlternateText>Nedir ASP?</AlternateText>
         <Keyword>ASP, Nedir.org, ADO</Keyword>
         <Impressions>55</Impressions>
         <EkAlan></EkAlan>
       </Ad>
     </Advertisements>
```

• AdRotator : (Sayfayı her yenilemede etki oranına göre farklı reklam geliyor)

- <%@ Page Language="C#" %>
- <script runat="server">
- void AdCreated (object Source, AdCreatedEventArgs e)

```
• { if( (string)e.AdProperties["EkAlan"] != "")
```

```
 lblEkAlan.Text = (string)e.AdProperties["EkAlan"];
```

```
• else lblEkAlan.Text = " Yok maalesef..." ;
```

- }
- </script>
- <html> <head> <title></title> </head>
- <body>
- <h1> ASP.NET'te AdRotator Kontrolu </h1>
- <asp:adrotator</pre>

```
ID="reklam"
```

- Target="_blank"
- AdvertisementFile="XMLFile.xml"
- OnAdCreated="AdCreated"
- runat="server"/>
-

- <asp:label ID="lblEkAlan" runat="server"/>
- </body>

Calendar

- Calendar, web sayları üzerinde Takvim göstermek için kullanılır.
- Örnekte Calender kullanımı gösterilmektedir.
- <asp:Calendar id="takvim" runat="server"/>
- Calendar kontrolüne ait özellikler tabloda listelenmiştir.

Özellik	Açıklama				
CellPadding	Hücreler ve kenarlıkları arasındaki boşluk değerini tutar.				
CellSpacing	Hücreler arasındaki boşluk değerini tutar.				
DayNameFormat	Gün isimlerinin görüntülenme biçimini tutar. FirstLetter, FirstTwoLetters,				
	Full ve Short değerleri vardır. Varsayılan Short değeridir.				
FirstDayOfWeek	Haftanın ilk gününü belirtir.				
NextPrevFormat	Next ve Previous linklerinin biçimini ayarlar. CustomText, FullMonth,				
	ShortMonth değerleri alır. Varsayılan CustomText değeridir.				
SelectedDate	Seçilen gün bilgisini tutar. Varsayılan değer günün tarihidir.				
SelectionMode	Calendar nesnesinin seçim modunu belirler. Day, DayWeek,				
	DayWeekMonth ve None değerleri alır. Varsayılan seçenek Day değeridir.				
ShowDayHeader	Gün isimlerini kolonların üzerinde görüntüler.				
ShowGridLines	Günleri hücreler içinde görüntüler.				
ShowNextPrevMonth	Önceki ve sonraki ay linklerinin görüntülenmesini sağlar.				
ShowTitle	Takvim başlığını görüntüler.				
TitleFormat	Başlık yazısının biçimini belirtir.				

- <%@ Page Language="C#" %>
- <!DOCTYPE html>
- <HTML>
- <HEAD><title>WebForm1</title>
- <script runat="server">
- private void tarih_secimi(object nesne, EventArgs e)
- {
- Label1.Text+="Secilen Tarih :";
- Label1.Text+=Calendar1.SelectedDate.ToShortDateString();
- Label1.Text+=" ";
- }
- </script>
- </HEAD>

- <body><form id="Form1" method="post" runat="server">
- <asp:Calendar id="Calendar1"</pre>
- OnSelectionChanged="tarih_secimi"
- SelectedDayStyle-Font-Bold="true"
- DayNameFormat="full"
- NextPrevFormat="FullMonth">
- </asp:Calendar>
- <asp:Label id="Label1"> </asp:Label>
- </form></body> </HTML>

 Burada dikkat edilmesi gereken nokta
 "OnSelectionChanged" olayı ile "tarih_secimi" yordamı işletilmektedir.
 "tarih_secimi" yordamında da
 Calendar' dan alınan bilgiler Label' in Text
 özelliğine eklenmektedir.

WebForm1 -	Microsoft In	ternet Ex	cplorer				-	
ile <u>E</u> dit <u>V</u> iew) F <u>a</u> vorites	<u>T</u> ools <u>H</u> e	lp					
🚱 Back 🝷		3 3		Search	*		Media 👖	
dress 👩 http:,	//localhost/Web/	Application2	2/WebForm1.	aspx		✓ (🔊 Go 🛛 Ba	ğlantılar »
								<u>~</u>
	<u>Ocak</u>			Şubat 2006			<u>Mart</u>	
	Pazartesi	Salı	Çarşamb	a Perşembe	Cuma	Cumartesi	Pazar	
	<u>30</u>	<u>31</u>	<u>1</u>	2	<u>3</u>	<u>4</u>	<u>5</u>	
	<u>6</u>	2	<u>8</u>	<u>9</u>	<u>10</u>	<u>11</u>	<u>12</u>	
	<u>13</u>	<u>14</u>	<u>15</u>	<u>16</u>	<u>17</u>	<u>18</u>	<u>19</u>	
	<u>20</u>	<u>21</u>	<u>22</u>	<u>23</u>	<u>24</u>	<u>25</u>	<u>26</u>	
	<u>27</u>	<u>28</u>	<u>1</u>	<u>2</u>	<u>3</u>	4	<u>5</u>	
	<u>6</u>	<u>7</u>	<u>8</u>	2	<u>10</u>	<u>11</u>	<u>12</u>	
	Secilen Tarih Secilen Tarih	1 :30.01.2 1 :10.03.2	2006 2006					~
						👊 L	ocal intranet	

- Web Forma kod eklemek için iki yol izlenir:
 - Inline Code: Web içeriği ile kod aynı sayfa içerisinde yer alır.
 ASP.NET kodu Script etiketi içerisine yazılır.
 - Code-behind: HTML içeriği ve C# kodu tamamen ayrı dosyalarda tutulur. Kod dosyasına code-behind sayfası denir. Bu metot Visual Studio .NET ortamının varsayılan çalışma şeklidir.

- Inline Kod Yazmak
- <mark><%@</mark> Page Language="C#" <mark>%></mark>
- <!DOCTYPE html>
- <head>
- <script runat="server">
- private void Page_Load (Sender As Object , e As EventArgs)
- { Message.Text = "Inline Kod Yazdık"; }
- </script>
- </head>
- ody>
- <form runat="server">

<asp:Label id="Message" runat="server" />

- </form>
- </body> </html>

Code-Behind Kod Yazmak



- Code-Behind Kod Yazmak
- Codebehind
- Code-behind sayfasının ismini temsil eder. Visual Studio .NET platformunun
- dosyayı birleştirebilmesi için gereklidir.
- Src
- Code-behind sayfasının ön derleme işleminden geçmediği durumlarda bu
- özellik kullanılır. Code-behind sayfasının ismini temsil eder.
- WebForm.aspx Code-behind kod yazmak
- <%@ Page Language="c#" Codebehind="WebForm.aspx.cs" Inherits="ilkAspNet.WebForm1"%>
- <HTML> <HEAD> <title>BilgeAdam</title> </HEAD>
- <body>
- <form id="Form1" method="post" runat="server">
- <asp:TextBox id="txtAd" runat="server"/>
- <asp:Button id="btnGonder" runat="server"</pre>
- Text="Gönder"></asp:Button>
- <asp:Label id="lblMesaj" runat="server"></asp:Label>
- </form>
- </body>
- </HTML>

- WebForm.aspx.cs Code-Behind Sayfası
- using System.Web;
- using System.Web.UI;
- using System.Web.UI.WebControls;
- private void Page_Load(object sender, System.EventArgs e)
- { Response.Write("Selam"); }
- private void Button1_Click(object sender, System.EventArgs e)
- { lblMesaj.Text = "Hosgeldin " + txtAd.Text.ToString(); }

Selam Hoşgeldin Bilgisayar Öğretmeni

Bilgisayar Öğ	retmeni
Gönder	

Olay Prosedürleri Oluşturmak

- Visual Studio .NET içerisinde server-side olay prosedürleri iki adım ile oluşturulur. Birinci adım olay üretecek kontrolü web form üzerine eklemektir.
- İkinci adım ise code-behind sayfasına olay prosedürünü eklemektir.
- Olay prosedürleri kontrolün ID özelliğine girilen değerden faydalanarak oluşturulur. Handles anahtar sözcüğü kontrolün hangi olay ile ilişkilendirileceğini belirler. Handles anahtar sözcüğü ile tek bir olay için birden fazla olay prosedürü oluşturulabilir.

- Olay Prosedürleri Oluşturmak
- <asp:Button id="btn1" runat="server"/>
- • •
- Tanım bölgesinde:protected System.Web.UI.WebControls.Button btn1
- private void btn1_Click(object sender, System.EventArgs e)
- {
- //...
- }
- Server olay prosedürlerinin çalışmasını sağlamak için kontrollerin
 WithEvents anahtar sözcüğü ile tanımlanması gerekir.
- Olay prosedürlerine iki parametre girilmelidir.
 - Sender: Olayı tetikleyen kontrol nesnesidir.
 - EventArgs: Olaya özgü parametreleri içeren nesnedir.

• Örnek:

- <asp:TextBox id="txtAd" runat="server" /> </asp:TextBox>
- <asp:Button id="btn1" runat="server"/>
- <asp:Label id="lblMesaj" runat="server" />
- private void Page_Load(System.Object sender, System.EventArgs e)
- { //sayfa içerisinde bulunan başka bir buton kendi olayı dışında btn1 butonuna ait olayı da çalıştıracaktır.
- Button1.Click +=new EventHandler(btn1_Click);
- }
- private void btn1_Click (System.Object sender, System.EventArgs e)

```
• {
```

```
string mesaj = "Merhaba " + txtAd.Text.ToString();
```

```
    lblMesaj.Text = Mesaj;
```

```
• }
```

• Sayfa Olayları

- Bir ASP.NET sayfası belirli olaylar sırası ile çalışır.
- 1- Page_Init (Page Initialize Sayfanın oluşmaya başlaması):
 - Bu aşamada sayfa başlatılır ve sayfadaki kontroller oluşturulur.

• 2- Page_Load (Sayfanın yüklenmesi):

- Bu olay, sayfa yüklendiği zaman tetiklenir.
- 3- Control Events (Kullanıcı kontrol olayları):
 - **Click** ve **Change** olaylarıdır. Bu olaylar kontrollerin tıklanması veya kontrol değerlerinin değişimi ile tetiklenir. TxtAd_Changed ve Btn1_Click gibi.

• 4- Page_Unload (Sayfanın Kapanması):

• Bu olay, sayfa kapandığı zaman tetiklenir.

• Sayfa Olayları

- Page Event yaşam döngüsü sonlandığında sayfaya ait bilgiler hafızadan silinir.
- Kontrol olaylarının çoğu, sayfa sunucuya geri gönderilene kadar gerçekleşmez. Örneğin Click olayı ile form sunucuya gönderilmeden Change olayları gerçekleşmez.

- Sunucuya veri gönderme işlemine "postback" denir. Örneğin Button kontrolü tıklanınca otomatik olarak sunucuya veri yollanır. Varsayılan olarak veri yollayan tek kontrol Button kontrolüdür.
- AutoPostBack özelliği, herhangi bir sunucu kontrolünün web sunucuya otomatik olarak bilgi göndermesini sağlar.
- AutoPostBack özelliği, DropDownList, ListBox, CheckBox, CheckBoxList, RadioButton, RadioButtonList, TextBox ve Button kontrolünde bulunur.
- Bu özelliğin True olması, seçim yapıldığında veya TextBox kontrolüne yeni bir değer girildiğinde sayfanın yeniden yüklenmesi anlamına gelir.

AUTOPOSTBACK

- DropDownList Kontrolunun Sunucuya Gönderilmesi
- <asp:DropDownList id="DropDownList1" runat="server" AutoPostBack="True">
- <asp:ListItem>Türkçe </asp:ListItem>
- <asp:ListItem>İngilizce</asp:ListItem>
- </asp:DropDownList>

}

- <asp:TextBox id="mesaj" runat="server"/>
- private void DropDownList1_SelectedIndexChanged(object sender, System.EventArgs e)

```
mesaj.Text = DropDownList1.SelectedItem.Value;
```

AUTOPOSTBACK

Page.IsPostBack

- Page Load olayı, sayfa yüklendiği zaman gerçekleşir. Sayfaya yapılan her istekte Page Load olayı içindeki kodlar çalışır. Aynı sayfanın her seferinde yeniden çalıştırılması, istenilen bir durum değildir.
- Page.IsPostback özelliği ile, sayfanın Load olayı içindeki kodlar sadece bir kez çalıştırılır. Böylece sayfa yeniden çağrıldığında bu işlemler gerçekleşmez.

AUTOPOSTBACK

Page.IsPostBack

- private void Page_Load(System.Object sender, System.EventArgs e)
 {
- //Sayfanın burasına gerçekleştirilecek kullanıcı kodları yazılır
- if(!Page.IsPostBack)
- { // sayfa ilk kez açıldığında yapılacak işlemler// dropdownlist,listbox,datagrid gibi veritabanından bilgi çekilip bu nesnelere atanması
- }
- else {
- // kontrollerden biri harekete geçtiğinde yapılacak işlemler (postback)
- //sayfa her yüklendiğinde çalışılan alan }

• }

- <%@ Page Language="C#" CodeFile="Default.aspx.cs" Inherits="_Default" %>
- <!DOCTYPE HTML >
- <head runat="server"> <title> Tablo</title> </head>
- <form id="Form2" runat="server"> Bildiğiniz Yabancı Dili Seçiniz:
- <asp:listbox id="lstDiller" runat="server" rows="3" AutoPostBack="true"
 onSelectedIndexChanged="secimGoster" Height="114px" Width="178px" />
-

- <asp:Label id="lblMesaj" runat="server" />

- </form>
- </body>
- </html>

```
using System;
  using System.Web;
  using System.Web.UI;
  using System.Web.UI.WebControls;
  public partial class _Default : System.Web.UI.Page
      protected void Page_Load(object sender, EventArgs e)
      { if (!Page.IsPostBack)
              lstDiller.Items.Add(new ListItem("ingilizce"));
           {
               lstDiller.Items.Add(new ListItem("Almanca"));
               lstDiller.Items.Add(new ListItem("Fransızca"));
               lstDiller.SelectedIndex = 0;
            }
       }
   public void secimGoster(object source, EventArgs e)
   { lblMesaj.Text = "Sectiginiz Dil " +
lstDiller.SelectedItem.Text;
     }
```

 AutoPostBack özelliğinin gereksiz yere kullanılması performansı olumsuz yönde etkiler.



- ViewState, kullanıcı ve sunucu arasında taşınan verilerin gizli bir alanda şifrelenerek saklanmasını sağlar. Forma ait tüm kontrollerin değerleri şifrelenir ve VIEWSTATE değişkende saklanır.
- Form sunucuya gönderildikten sonra bir hata oluşması halinde kullanıcıdan tekrar aynı verilerin girilmesi istenmez. Çünkü kullanıcının girmiş olduğu değerler bu gizli değişkende saklanır ve sayfa açılınca tekrar kullanıcıya sunulur.
- Basit bir web uygulaması açalım. İçerisinde hiç bir değişiklik yapmadan debug edip, kaynak kodunu açtığımızda şu html çıktısı ile karşılaşırız;

- <html xmlns="http://www.w3.org/1999/xhtml" >
- <head><title>Untitled Page</title></head>
- <body>
- <form name="form1" method="post" action="Default.aspx" id="form1">
- div>
- <input type="hidden" name="__VIEWSTATE" id="__VIEWSTATE"
 value="/wEPDwUJNzgzNDMwNTMzZGRKSpuLejDHB6P+RKbZyCauWTRAqA==" />
- </div>
- </form>
- </body></html>
- Burada görüldüğü üzere ASP.NET, kendiliğinden bu gizli kontrolü oluşturmaktadır.
 Value özelliğine baktığımızda ise sayfadaki bazı bilgilerin kodlanmış hali görülmektedir.



- Örnek
- protected void Page_Load(object sender, EventArgs e)
- { if (!Page.IsPostBack)
- { DropDownList1.Items.Add("CSharp"); DropDownList1.Items.Add("Nedir"); DropDownList1.Items.Add(".com"); }

```
• }
```

- protected void btnSirala_Click(object sender, EventArgs e)
- { ArrayList ary = new ArrayList(); ary.AddRange(DropDownList1.Items); ary.Reverse(); DropDownList1.Items.Clear();
- foreach (object obj in ary)
- { DropDownList1.Items.Add(obj.ToString()); }

```
• }
```



- ViewState'i kapatarak bu programı çalıştırın;
- <asp: DropDownList ID= "DropDownList1" runat="server"
 EnableViewState ="false" >
- ViewState kapatıldığında ise düğmeye basıldıktan sonra dropDownList1 boş gelecektir. Sayfa post back olunca tüm html kontrolleri sunucuda yeniden hazırlanıp gönderilmiştir.
- !Page.IsPostBack deyimi de olduğu için DropDownList'imizin items koleksiyonu boş kalmıştır.
- !Page.IsPostBack'in kaldırılması çözüm değildir, bu seferde dropDownList1'e her seferinde aynı liste baştan eklenecek dolayısıyla Reverse metodu istediği kadar çalıştırılsın devamlı ".com, Nedir, Csharp" şeklinde sıralama olacaktır. Eğer EnableViewState ="true" olursa bu defada liste her defasında artarak genişleyecektir.



• Örnek:

Unbound

- protected void Page_Load(object sender, Eventarys e)
- //ViewState kontrolümüze dışarıdan bir nesne eklenmeye çalışıldığı zaman aynı Hashtable gibi çalışır. Bizden bir 'Key' birde 'Value' değeri ister. Hashtable'dan farklı olarak 'Key' değerini string olarak alır.'Value' değerini ise object olarak aldıktan sonra StateItem tipine cast eder. Artık eklendikten sonra bu verilere, istenildiği kadar postback yapılsın, ne zaman arzu edilirse 'Key' değeri üzerinden ulaşılabilir.
- Tazele();
- }

- protected void btnCikar_Click(object sender_
- { ViewState.Remove(txtCikar.Text.ToString());
- //Remove metodu ise sadece key değeri üzerinden Tazele(); işlem yapmaktadır}

EKLE

CIKART

Unbound

}

- private void Tazele() { //ListBox1.Items.Clear();
- //Normalde bu metodu yazmadığımız takdirde ListBox nesnemiz içeriğini ViewState vasıtasıyla taşımaya devam eder. Bu da verilerin birikmesine sebep olmaktadır.
- Dolayısıyla html tarafında ListBox'ın ViewState özelliğini kapatarak hem bu satırı yazmaktan kurtuluruz, hem de ViewState'i daha az yorarız. Kod tarafında herhangi bir kontrolün Items.Clear() metodunu çalıştırdığımız zaman ViewState'ten o kontrole ait nesneler kaldırılmaktadır.
- foreach (StateItem sti in ViewState.Values)
- { ListBox1.Items.Add(sti.Value.ToString());

- Asp.Net 2.0 ile birlikte gelen en önemli yeniliklerden biriside Master Page kavramıdır. Master Page; bir sitedeki sayfalarının tamamının yada bir kısmının aynı şablon üzerinde oturmasını sağlayan yapıdır.
- Master Page kullanımı ile elde edilen avantajlar sadece görsel açıdan gelen bu kolaylık ile sınırlı değildir. Sonuç itibariye her Master Page aynı zamanda arka planda bir örnek olarak oluşturulan bir sınıf tanımlamasıdır.
- Bu nedenle Master Page uyarlanan içerik sayfalarının (Content Page) ortaklaşa kullanabileceği fonksiyonellikleri dahi barındırabilir. Buda tam anlamıyla kalıtım (inheritance) ile yapabildiklerimizin bir yansımasıdır.

- Sağladığı ContentPlaceHolder bileşeni sayesinde, Master Page'leri uygulayan diğer aspx sayfalarının, istenilen içeriğe sahip olmasını da sağlamaktadır. En güzel yanı ise, normal bir aspx sayfası gibi tasarlanabilmesi, yani html, image, server control gibi üyeleri içerebilmesidir. Bunlara ek olarak, olay güdümlü programlama modelinide destekler. Dolayısıyla bir Master Page aslında bir aspx sayfasından farksızdır.
- Ancak asıl fark, bir Master Page bir aspx sayfasına uygulandığında ortaya çıkar. Master Page' i uygulayan bir aspx sayfası tarayıcıda açıldığında tarayıcıya gelen sayfa, Master Page ile aspx sayfasının birleştirilmesi sonucu ortaya çıkan başka bir aspx sayfasıdır. Bu, .net framework'ün getirdiği partial class tekniği sayesinde gerçekleşebilmektedir. Bunun, Master Page'i uygulayan aspx sayfalarına getirdiği değişik kodlama etkileride vardır.



- File | New | Web Site ile bir Asp.Net/C# projesi oluşturarak başlıyoruz.
- Projemizi oluşturduğumuzda Solution Explorer'a Default.aspx olarak gelen sayfamızı üzerinde sağ tıklayıp Delete komutunu vererek siliyoruz. (2010 da web sayfası master page ile gelir.)
- Şimdi adım adım sayfalarımızı oluşturalım.
- Sayfamıza WebSite | Add New Item ile yeni bir MasterPage sayfası ekliyoruz.

emplates:				
Visual Stud	lio installed templa	ites		
Web Forn AJAX Clie AJAX Mar Browser I Crystal R Global Ap LINQ to S Resource SQL Serv WCF Serv MCF Serv	n rit Behavior ster Page File eport oplication Class SQL Classes o File er Database vice	Master Fage HUAA Clear Control AJAX Web Form Class DataSet HTML Page Report Site Map Style Sheet Web Configuration File Web Configuration File	Web User Control AJAX Clent Library AJAX-enabled WCF Service Class Diagram Generic Handler JScript File Report Wizard Skin File Text File Web Service XSLT File	
My Templa	ites			
A Master Page	for Web Applications			
ame:	MasterPage.ma	ster		
anguage:	Visual C#	Place code	e in separate file ster page	

amplatari				
Visual Studi	io installed templa	tes		
Web Form AJAX Cler AJAX Masl Browser F Crystal Re Global App Crystal Re Global App Crystal Re Global App Crystal Re Global App Crystal Re Global App Crystal Re Global App Crystal Re Global App Crystal Re Global App Crystal Re Global App Crystal Re Global App Crystal Re Global App Crystal Re Global App Crystal Re Cryst	nt Behavior ter Page ile aport plication Class QL Classes File ar Database ice	Master Rage ROAX Clean Control AJAX Web Form Class DataSet HTML Page Report Site Map Skyle Sheet Web Configuration File	Web User Control AJAX Client Library AJAX-enabled WCF Service Class Diagram Generic Handler JScript File Report Wizard Skin File Text File Web Service Web Service	
My Templal	tes			6
A Master Page I	for Web Applications			
ame:	MasterPage.mas	ter		
anguage:	Visual C#	Place code	in separate file	



Bunun ismini istediğiniz gibi değiştirebilirsiniz.

MasterPage sayfasının basit bir tasarımı



- Burada eklediğimiz nesneleri açıklamak gerekirse; TreeView sayfanın menüsünü oluşturacak nesnedir. SiteMap site haritasını gösteren nesnedir. ContentPlaceHolder'da MasterPage'i içinde diğer sayfalardan erişebileceğimiz içerik bölümüdür. "*head*" Content Place Holder' ı Sitemizin genelinde etkin olmayacak, Content Page içerisinde tanımlayacağımız (html' de ki meta etiketleri gibi) elemanlar için yer tutacak.
- Sayfanın html taglari;

- <html> <head runat="server"> <title>TreeView ile Menü Yapıyoruz</title>
- <asp:ContentPlaceHolder id="head" runat="server">
- <!--Burada sayfaya özgü head tanımlayabiliriz. -->
- </asp:ContentPlaceHolder>
- </head>
- <body>
- <form id="formicerik" runat="server">
- div>
- •
- <h2> Asp.Net TreeView ile Menü Yapımı</h2>
- •

- <asp:SiteMapPath ID="SiteMapPathHaritaNesnesi" runat="server"> </asp:SiteMapPath>
- •
- <asp:TreeView ID="TreeViewMenuNesnesi" runat="server" DataSourceID="SiteMapDataSource"> </asp:TreeView>

- <div>
- <asp:ContentPlaceHolder ID="iceriksayfasiCPH" runat="server">
- <!--Buraya ana sayfamızın içerikleri gelecek -->
- </asp:ContentPlaceHolder>
- </div> </div> </form> </body> </html>

• Web Site | Add New Item ile yeni bir WebForm ekliyoruz.

Add New Iten	n - C: Wocuments	and Settings\ByPsycho\Belgele	erim\Visual Studio_2008\TreeView-ile .	. ? 🛛
Templates:				
Visual Stud	lio installed templ	ates		^
AJAX Cre AJAX Cre AJAX Ma D Browser I Crystal R Global Ap LINQ to S Resource SQL Serv WCF Serv XML File	The Behavior ster Page File teport oplication Class SQL Classes a File rer Database vice	Master Page AJAX Client Control AJAX Web Form Class DataSet HTML Page Report Site Map Style Sheet Web Configuration File	Web User Control AJAX Client Library AJAX-enabled WCF Service Class Diagram Generic Handler JScript File Report Wizard Skin File Text File Web Service XSLT File	
A form for Web	h Annications			
Name:	Anasayfa.aspx			
Language:	Visual C#	Place code	e in separate file ster page	
			Add	Cancel

- Bu adımda dikkat etmemiz gereken bir nokta var. Normal WebForm'ları eklerken Select master page seçeneğini işaretlemiyorduk. Ama bu sefer bunu işaretlememiz gerekiyor. Çünkü eklenecek olan bu sayfa MasterPage kalıbı içinde yer alacak. Bu seçeneği işaretledikten sonra Add butonuna tıklıyoruz.
- Şimdi karşımıza gelen pencere bize MasterPage sayfamızı belirmemizi istiyor. Burada sayfamıza eklediğimiz MasterPage'ler görünüyor. Birden fazlada MasterPage kullanmak mümkündür. Bunun için bir sınırlama yok. Bunuda seçtikten sonra OK butonuna tıklıyoruz.



- <%@ Page Title="Anasayfa" Language="C#" MasterPageFile="~/MasterPage.master" AutoEventWireup="false" CodeFile="Anasayfa.aspx.cs" Inherits="Anasayfa"%>
- <asp:Content ID="Content1" ContentPlaceHolderID="head" runat="server">
- <!—Anasayfaya özgü başlık, meta etiketleri tanımlayabiliriz. -->
- </asp:Content>
- <asp:Content ID ="Content2" ContentPlaceHolderID="iceriksayfasiCPH" runat="server">
- Anasayfa içeriği burada yer alacak.
- </asp:Content>

- Bu şekilde aynı işlemi yaparak "Anasayfa.aspx", "Dokumanlar.aspx", "Giris.aspx", "Hakkimizda.aspx", "Makaleler.aspx" şeklinde 5 tane sayfa ekleyiniz.
- Sayfada ContentPlaceHolder(iceriksayfasiCPH) dışında başka bir bölgeye etki edemiyorsunuz. Çünkü diğer kısımları MasterPage içinde yaptık. ContentPlaceHolder ile de kullanıcının diğer sayfalardan etki edeceği alanı belirledik.

<asp: <asp: </asp: </asp: <th>ontent ID="Con Content> Content ID="Con Content ID="Con Content></th> <th>C#" MasterPageFile=" ntent1" ContentPlace ntent2" ContentPlace</th> <th>"-/MasterPage.master" HolderID="head" Runat HolderID="iceriksayfs</th> <th>AutoEventWireup="true" CodeFi ="Server"> MBICPH" Runat="Server"></th>	ontent ID="Con Content> Content ID="Con Content ID="Con Content>	C#" MasterPageFile=" ntent1" ContentPlace ntent2" ContentPlace	"-/MasterPage.master" HolderID="head" Runat HolderID="iceriksayfs	AutoEventWireup="true" CodeFi ="Server"> MBICPH" Runat="Server">
				MasterPage.n
		Asp.Net Tree	/iew ile Menü Ya	ipimi
Root N	lode > Parent Nod	le > Current Node	iaintai b	inet
B Ro	ot Parent 1 Leaf 1 Leaf 2 Parent 2 Leaf 1 Leaf 2	iceriksayfasiCPH (Custom)		

• Sayfalarımızı eklediğimizde bu şekilde bir görünüm elde edeceğiz.

Asp.Net	TreeView ile Menü Yapımı
Hot Node > Parent Node > Current No	de
■ Root ■ Parent 1 Leaf 1 Leaf 2 ■ Parent 2 Leaf 1 Leaf 2 Leaf 1 Leaf 2	analkurs.net
Bu adımda sitemizin haritasını oluşturacağız. Bunun için ilk olarak Website | Add New Item ile bir tane SiteMap nesnesi ekliyoruz.

Add New Iten	n - C:\Documents	and Settings\ByPsycho\Belgele	rim\Visual Studio 2008\TreeView-ile 📝	×
Templates:				
Visual Stud	lio installed templa	stes		^
Web Forn AJAX Cle AJAX Mar D Browser I Crystal R Global Ap LINQ to S Resource SQL Serv SQL Serv MUCF Serv MUCF Serv MUCF Serv	n ster Page File leport splication Class SQL Classes e File rer Database vice	Master Page AJAX Client Control AJAX Web Form Class DataSet HTML Page Report Site Map Syle Dinee Web Configuration File	Web User Control AJAX Client Library AJAX-enabled WCF Service Class Diagram Generic Handler JScript File Report Wizard Skin File Text File Web Service XSLT File	
A file used to c	create a site map			
Name:	Web.sitemap			
Language:	Visual C#	Place code Select mas	in separate file fer page	
			Add Cancel	



- Şimdi sıra SiteMap içinde sitemizin haritasını çıkarmaya. Bunun için Web.sitemap üzerinde çift tıklıyoruz ve aşağıdaki gibi düzenliyoruz.
- <?xml version="1.0" encoding="utf-8" ?>
- <siteMap xmlns="http://schemas.microsoft.com/AspNet/SiteMap-File-1.0" >
 - <siteMapNode url="Anasayfa.aspx" title="Anasayfa" description="">
 - <siteMapNode url="Giris.aspx" title="Giris" description="">
 - <siteMapNode url="Dokumanlar.aspx" title="Dökümanlar" description="" />
 - <siteMapNode url="Makaleler.aspx" title="Makaleler" description=""/>
 - </siteMapNode>
 - <siteMapNode url="Hakkimizda.aspx" title="Hakkimizda" description=""/>
 - </siteMapNode>
- </siteMap>
- "/ " ifadesi ağacın girintisini yani ağaca bağlı alt dizinleri gösterir. " dizini olmayan boş kalacak ve sonlandırma deyimi mutlaka bulunacak.

 Burada url; sayfanın adı, title; menünün adı; description; menünün üzerinde geldiğimizde çıkacak açıklamadır.
 Ana menüyü <siteMapNode url=""> </siteMapNode> şeklinde yaptık.
 Bunun alt menülerini oluşturmak içinde bu taglerin arasına <siteMapNode url=""/> şeklinde alt menüleri belirttik. Yani kategorileme gibi alt menüyü içte ana menüyü dışta belirttik.

+ Anasayfa ++ Giriş +++ Dokümanlar +++ Makaleler ++ Hakkımızda

 Bu kısımda haritamızı menü nesnelerimiz ile bağlayacağız. Bunun içinde sayfaya Toolbox'dan SiteMapDataSource nesnesini ekliyoruz.



 Bunu ekledikten sonra TreeView nesnesini seçtikten sonra sağ üst köşesinde bulunan küçük ok simgesine tıklıyoruz. Buradan açılan menüde Choose Data Source kısmında veri kaynağını belirtiyoruz. Buda sayfamıza daha önce eklediğimiz SiteMapDataSource olarak belirliyoruz. Daha sonra menümüzün direk olarak oluştuğunu göreceksiniz.

🗄 Anasayfa 🛛 🖌	🔨 TreeView Tasks			
🖻 Giriş	Auto Format			
Dökümanlar	Choose Data Source: SiteMapDataSource			
Makaleler	Refresh Schema	(None) SiteMapDataSpurce		
Hakkımızda	Edit TreeNode Databin <new data="" source=""></new>			
U V	Show Lines			

 Burada aklınıza şöyle bir soru gelebilir. Peki bu olayı SiteMap nesnesi için neden yapmadık. Çünkü SiteMap nesnesi sayfa çalıştığında otomatik olarak bunu çekecektir. Açılan sayfaların yolunu gösterecektir. Bu arada sayfaya eklediğimiz SiteMap nesnesi ile Solution Explorer'daki SiteMap'ı karıştırmayın. Birisinde harita oluşturulurken diğeri haritayı sayfada gösterir.

 Sitemizin başlangıç sayfamızı belirleyip sitemizin nasıl çalıştığına bakacağız. Bunun için Solution Explorer'daki Anasayfa.aspx üzerinde sağ tıklayarak Set as Start Page'e tıklıyoruz.



Hakkımızda

ASP.NET

İstek (Request) ve Cevap (Response)

 İnternette web sayfalarının görüntülenmesi için kullanıcının, web sunucusundan (server) bir istek yapması ve sunucunun da bu isteğe karşılık vermesi gerekmektedir. Bildiğimiz gibi kullanıcı bu isteğini, tarayıcısından (browser) ziyaret etmek istediği web sayfasının adresini yazarak yapar.



 Response.Write : İstenilen değişken veya yazının sayfada görüntülenmesi. ASP.Net'in çok kullanılan ifadelerinden biridir. Response.Write ifadesinde html tagları kullanabiliriz;

• <mark><%</mark> int x = 45; <mark>%></mark>

- Response.Write("<h2>"+x+"</H2>");

• <mark>%></mark>

- <mark><%@</mark> Page Language="C#" <mark>%></mark>
- <html> <head runat="server"> </head> <body>
- <form id="form2" runat="server">
- <div>

- <asp:Label ID="Label2" runat="server" Text="Ad"></asp:Label>
- <asp:TextBox ID="TextBox1" runat="server">
 </asp:TextBox></div>
- •
- <mark><%</mark>int d = 33; <mark>%></mark>
- <% Response.Write("<h1>Erkan TANYILDIZI, Yaş: </h1>");
- Response.Write("<h2>"+d+"</H2>"); %>
- </form> </body> </html>

<u>D</u> osya	Düzen		Sik Kullanilanlar	Araçlar	Yardım		
G	eri - 🕻		2 🏠	🔎 Ara	sik Kull	anılanlar 🧭	
Adr <u>e</u> s	🗿 http://	10.6.2.77/W	ebSite/Proje/Defa	ault.aspx	💌 🔁 G	t Bağlantılar	» 🔁 -
-		TA	NVII I	DIZ	I Vo	a •	
\mathbf{Er}	кап	IIA			г, га	Ş:	
Er 33	кап	IA			L, 1 a	Ş.	

- Örnek: Response_write2.aspx
- <%@ Page Language="C#" %>
- <!DOCTYPE html>
- <html><head runat="server"></head><body>
- div>
- •

```
Response.Write("<font size=" + i + ">Yazılım Mühendisliği <br>></font>");
```

• }

```
Response.Write ("<hr width=80% \>");
```

- %>
- Varsayılan font büyüklüğü 3 Yazılım Mühendisliği
- </div>
- </body>
- </html>



```
Varsayılan font büyüklüğü 3 Yazılım Mühendisliği
```

- Write ile yazdan ifadelerde dikkat etmemiz gereken başka bir nokta, sabit ifadelerin sunucu için anlamlı olabilecek ifadeler içermemesidir Örneğin
- <mark><%</mark> Response.Write ("<hr width=80%>") <mark>%></mark>
- •
- ifadesinde, sayfanın yüzde sekseninde enine bir çizgi oluşturmak istediğimizi bildiren 80% ifadesi aynı zamanda %> şeklinde, sunucu için anlamlı bir tag oluşturmaktadır.
- Bunu önlemek için ifademizi aşağıdaki şekilde yazmalıyız:
- <mark><%</mark> Response.Write ("<hr width=80%\>") <mark>%></mark>

- Response.Redirect (URL): Bu ifadeye geldiğinde mevcut ASP sayfası çalışmasını durdurup belirtilen URL adresindeki sayfayı browserımıza getirir.
- Kullanıcıyı bir sayfadan başka bir sayfaya yönlendirmek için kullanılır.
- HyperLink kontrolü gibi sayfalar arasında dolaşmayı sağlar. Parametre olarak gidilecek sayfanın adresini belirtilir.
- Response.Redirect("yonlendirilen.aspx");
- Response.Redirect("anasayfa.html");

- <%@ Page Language="C#" CodeFile="Redirect.aspx.cs" %>
- <html>
- <head runat="server"></head>
- <body>
- <form id="form2" runat="server">
- </l
- Response.Redirect("yonlendirilen.aspx"); %>
- </form>
- </body>

- <%@ Page Language="C#" CodeFile="yonlendirilen.aspx.cs" %>
- <html ><head runat="server"></head>
- <body>
 - <form id="form2" runat="server">
- <% Response.Write ("<h4> Bu sayfa, bir önceki Redirect.aspx sayfasının çalıştırılması ile görüntülenmiştir</h4>"); %>
- </form>
- </body>
- </html>



- private void ListBox1_SelectedIndexChanged(object sender, System.EventArgs e)
 - { Response.Redirect(ListBox1.SelectedValue); }

<asp:ListBox id="ListBox2" runat="server" AutoPostBack="True">

- <asp:ListItem Value="http://www.yahoo.com">yahoo
- </asp:ListItem>
- <asp:ListItem Value="http://www.google.com">google
- </asp:ListItem>
- </asp:ListBox>

Response.Buffer:

 Respose.Buffer=true şeklindeki bir ifade, server sayfa içindeki bütün sunucu scriptlerin işlevi tamamlanmadan, browser'a cevap göndermeyecektir. Eğer Response.Buffer=false ise tamponlama yapılmadan bilgileri hemen browsera gönderecektir.

• Response.Flush:

 Sunucu ASP sayfası içindeki satırlarda bu ifadeye rastlarsa, o zamana kadar buffer'da (tampon hafıza) sakladığı sayfa değerlerini, browser'a gönderir.

- <html >
- <head runat="server"></head> <body>
- «Response.Buffer = true; %
- Bu satır sunucunun tampon (buffer) hatızasında saklanan ilk satır. Bu satırdan sonraki " flush" metodu ile hemen browsera gönderilecek
- <mark><%</mark> Response.Flush();<mark>%></mark>
- Bu yazı flush metodundan sonra gösterilmiştir
- </body>
- </html>

- Response.Clear(): Sunucunun o zamana dek tampon hafızasında kullanıcıya göndermek üzere tutuğu bütün bilgileri siler.
- <mark><%</mark> Response.Buffer = true;<mark>%></mark>
- Bu satır sunucunun tampon (buffer) hafızasında saklanan ilk satır.
- bu satırdan sonraki Response.Clear ile ne yazık ki silinecek ve sayfada gözükmeyecek
- <mark><%</mark> Response.Clear();<mark>%></mark>
- Bu yazı Clear metodundan sonra
- gösterilmiştir </body> </html>



Response.End

- Response.Clear metodunun tam tersidir. Bu satırı gördüğünde sunucu işlemlerini durdurup, o zamana kadar olan tampon hafıza değerlerini browser'a gönderir ve scriptin çalışması durur.
- <html ><head runat="server"></head> <body>
- <mark><%</mark> Response.Buffer = true;<mark>%></mark>
- Bu yazıdan sonraki yazılar gözükmüyor, çünkü sunucuya bu satırdan hemen sonra "dur ve tampon hafızasındaki değerleri gönder" komutu verdik.
- « <% Response.End();%»</p>
- Bu yazı hiçbir zaman
- görüntülenmeyecek
- </body> </html>

🗿 http://10.6.2.77/WebSite/Proje/Default.aspx 🔳 🗖	
<u>D</u> osya Dü <u>z</u> en <u>G</u> örünüm <u>Sık Kullanılanlar A</u> raçlar <u>Y</u> ardım	2
🕞 Geri 👻 🕑 👻 🛃 🏠 🔎 Ara	»
Adres 🗃 http://10.6.2.77/WebSite/P 💌 🔁 Git 🛛 Bağlantılar 🌺 🐬	•
Bu yazıdan sonraki yazılar gözükmüyor, çünkü sunucuya bu satırdan hemen sonra "dur ve tampon hafizasındaki değerleri gönder" komutu verdik.	<
🕘 Bitti 🧶 🔮 Internet	

- Response.Expires : Sunucu sayfaları gönderdiğinde, bu sayfalar ziyaretçinin bilgisayarında geçici bir bellekte (chace) saklanır. Browser ilk önce, adreste belirtilen sayfayı bu bellekte arar ve bulursa görüntüler.
- Burada amaç "hız"dır. Ancak bazen sayfanın her seferinde sunucudan gönderilmesi gerekebilir. Sunucuda belki birtakım değişkenlerin değeri değişecek, ya da birtakım güncellemeler yapılıyor olabilecektir. Bu durumda sayfaların ziyaretçinin bilgisayarındaki geçici bellekte saklanmasını engellememiz gerekir. Bunu "expires" özelliği ile sağlayabiliriz. Response. Expires 'da verilen değerler saniye cinsindendir.

- <%@ Page Language="C#" %>
- <mark><%</mark> Response.Expires = 0;<mark>%></mark>
- <html ><head runat="server"></head>
- <body>
- Bu sayfa geçici bellekte saklanmayacak,
- her seferinde sunucudan gönderilecektir.
- </body>
- </html>
- Expires değerini sıfır yaparak sayfanın geçici hafızada (chace) saklanmasını engelledik.

- Response.ExpiresAbsolute: Sayfaların belirli bir tarihe kadar geçici bellekte saklanmasını sağlar.
- <%@ Page Language="C#" %>
- <% Response.ExpiresAbsolute = new DateTime (2020, 1,7, 0, 0, 0); </p>
- <html ><head runat="server"> </head><body>
- Bu sayfa 1 Haziran 2020 saat 00:00:00 a kadar geçici bellekte saklanacaktır.
- </body></html>

- Response.Charset : Bir çok farklı karakter seti mevcuttur. Bu özellikle bilindik karakter setleri sayfanızda kullanabilirsiniz. Bu özellikle ilgili önemli bir bilgide bu değişiklikleri Response.BufferOutput özelliği false iken yapmalıyız. Aksi halde sayfa hata verir. Türkçe karakter için
- <%Response.CharSet = "UTF-8"; %> kullanılmalıdır.
- Response.Cookies : Response.Cookies özelliği, HttpCookie nesnelerinin bir koleksiyonunu içerir. Cookie'ler temel olarak iki şekilde oluşturulur.
- // Anahtar kelimelerde Türkçe karakter kullanılmaz.....

- <%@ Page Language="C#" CodeFile="Default.aspx.cs" Inherits="_Default" %>
- <mark><%</mark> Response.Expires = 0;<mark>%></mark>
- <html xmlns="http://www.w3.org/1999/xhtml" >
- <head runat="server"> </head>
- ody>
- Cookie oluşturma
- <form runat="server" method="post">
- <asp:Button ID="Button4" runat="server" OnClick="Button1_Click"
 Text="Button" />
- </form>
- </body> </html>

- protected void Page_Load(object sender, EventArgs e) {
- // 1. yapı Anahtar ve değeri
- // 1. yol
- Response.Cookies["KullaniciBilgi"]["KullaniciAdi"] = "Yazılım Mühendisliği";
- Response.Cookies["KullaniciBilgi"]["SonZiyaret"] = DateTime.Now.ToString();
- Response.Cookies["KullaniciBilgi"].Expires = DateTime.Now.AddDays(3); //geçerlilik süresi
- //2.Yol
- HttpCookie cerez = new HttpCookie("sevilenRenk", "Mavi");
- Response.Cookies.Add(cerez);

•

- //2.yapı Tek anahtar
- //1.yol
- Response.Cookies["R"]="Beyaz"; //Hata tek anahtar oluşturulamaz
- //2.Yol
- HttpCookie cerez = new HttpCookie("Kullanici");
- cerez.Values["Sifre"] = "123Bilgisayar456";
- cerez.Values["SonErisim"] = DateTime.Now.ToString();
- cerez.Expires = DateTime.Now.AddDays(1);
- Response.Cookies.Add(cerez);
- }
- protected void Button1_Click(object sender, EventArgs e)
 { Response.Redirect("yonlendirilen.aspx"); }

🕘 http:/	/10.6.	2.77/We	bSite/Pr		
<u>D</u> osya I	Dü <u>z</u> en	<u>G</u> örünüm	<u>S</u> ik Kullanilanla	: »	-
G Geri	- 6	- 💌	2 🏠		»
Adr <u>e</u> s	http 🔽	🔁 Git	Bağlantılar	»	D -
Cookie	oluștu	rma			^
Buttor	1				~
é			Internet		:

- <%@ Page Language="C#" CodeFile="yonlendirilen.aspx.cs" %>
- <head runat="server"> <%Response.Charset = "UTF-8"; %> </head>
- <body>
- <form id="form2" runat="server">

- <asp:Label ID="Label2" runat="server" Text="Label"> </asp:Label>
-

- <asp:Label ID="Label3" runat="server" Text="Label"> </asp:Label>
- <asp:Label ID="Label4" runat="server" Text="Label"> </asp:Label>
- </form>
- </body>
- </html>

- yonlendirilen.aspx.cs
- protected void Page_Load(object sender, EventArgs e) {
- //1.yol Okuma
- if (Request.Cookies["KullaniciBilgi"]!= null)
- {Label1.Text = Server.HtmlEncode (Request.Cookies["KullaniciBilgi"]["KullaniciAdi"]);
- Label2.Text = Server.HtmlEncode (Request.Cookies["KullaniciBilgi"]["SonZiyaret"]);
- //Response.Write(Server.HtmlEncode(Request.Cookies["R"])); hata tek anahtar kullanımı gerçekleşmez
- HttpCookie cerez = Request.Cookies["Kullanici"];
- HttpCookie cerez2 = Request.Cookies["sevilenRenk"];

• //2.yol okuma

}

- Response.Write(Server.HtmlEncode(cerez.Values["Sifre"]));
- Response.Write("
");
 Response.Write("
");
- Response.Write(Server.HtmlEncode(cerez.Values["SonErisim"]));
- Response.Write("
"); Response.Write("
");
- Label3.Text = Server.HtmlEncode(cerez2.Value);

🚰 Untitled Page - Microsoft Interne 📃 🗖 隆
Dosya Düzen Görünüm Sık Kullanılanlar 🤉 🌺 🥂
🔇 Geri 👻 🛞 - 🔀 🛃 🔎 Ara
Adres 💩 http://10 🕶 🂽 Git 🛛 Bağlantılar 🎽 🧙 🗸
123Bilgisayar456
27.01.2008 15:52:48
Bigisayar -Elektronik
27.01.2008 15:52:48
Mavi
🙆 Bit

- Cookie'leri okumak için döngü yapısını da kullanabilirsiniz;
- System.Text.StringBuilder output = new System.Text.StringBuilder();
- HttpCookie aCookie;
- for (int i = 0; i < Request.Cookies.Count; i++)</pre>
- {
- aCookie = Request.Cookies[i];
- output.Append("Cookie value = " + Server.HtmlEncode(aCookie.Value)
- + "

");
- }
- Label1.Text = output.ToString();

HttpResponse Nesnesinin Metotları :

- *BinaryWrite* : Bir dosyanın içeriğini istemciye yazar.
- using System.IO;

```
...
  protected void Page_Load(object sender, EventArgs e)
  {    FileStream MyFileStream; long FileSize;
    string path = @"C:\de.txt";
    MyFileStream = new FileStream(path, FileMode.Open,
    FileAccess.Read);
    FileSize = MyFileStream.Length; byte[] Buffer = new
    byte[(int)FileSize];
    MyFileStream.Read(Buffer, 0, (int)FileSize);
    MyFileStream.Close();
    Response.Write("<b>Dosya İçeriği: </b>");
    Response.BinaryWrite(Buffer);
```

HttpResponse Nesnesinin Metotları :

- WriteFile : Bir dosyanın içeriğini istemciye yazar. BinaryWrite örneğindeki yaptığımız işi aşağıdaki kodlarla da yapabiliriz.
- protected void Page_Load(object sender, EventArgs e)
 - string path = @"C:\de.txt";
 - Response.Write("Dosya İçeriği: ");
- Response.Write("");
- Response.WriteFile(path);
 - Response.Write("");

Response Nesnesini Script İçerisinde Kullanılması

- Code-behind 'de yazdığımız kodları HTML'in içerisine gömebiliriz. Bunun için kullanacağımız Visual Studio .NET dillerinden hangisini kullanacağımızı daha önceden belirlemeliyiz.
- <%@ Page language="c#" Codebehind="WebForm1.aspx.cs" AutoEventWireup="false" Inherits="WebApplication7.WebForm1" %>
- Eğer HTML'nin içerisine kod gömmek istersek bunu <script>....</script> tagları arasına yazmalıyız. Buna örnek olarak aşağıdaki kodları inceleyiniz.

- <%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>
- <HTML> <HEAD><title>WebForm1</title>
- <script runat="server">
- private void Page_Load(object sender, System.EventArgs e) {
- string isim="Bilgisayar Öğretmenliği";
- Response.Write("isim
");
- Response.Write(isim);
- } </script> </HEAD>
- <body><form id="Form2" method="post" runat="server">
 </form> </body> </HTML>

 Uygulama: Gelecek ders için, Response Nesnesinin diğer özellik ve metotları öğrenilecek ve birer örnek üzerinde değerlendirilecek.

Request Nesnesi

Dersleme Ders notları C# İLE ASP.NET Zafer Demİrkol Çeşitli İnternet kaynakları
Request.Querystring

- Aynı sayfaya veya başka sayfalara değişken veya değişkenlerin taşınmasını sağlamada çok kullanılan bir ifadedir.
- <a href> tagıyla veya Response.Redirect ile , yönlendirilecek sayfa yol tanımından başka "?" işareti ile başlayan özel bir takım ifadeler var. "?" işareti ardından sayfaya taşıyacağımız değişkenin adını ve değerini belirtiriz. Birden fazla değişken ve değerleri bir birlerinden "&" işareti ile ayrılırlar.

- <mark><%</mark>@ Page Language="C#" <mark>%></mark>
- <html > <head runat="server"></head>
- ody>
-
- link kodu içindeki soru işaretinden sonraki değişkenler link adresinde belirtilen sayfaya taşınacak.
- </body>
- </html>



- <mark><%@</mark> Page Language="C#" <mark>%></mark>
- <html><head runat="server"></head>
- <body>

- <% string kategori = Request.QueryString ["kitapkategorisi"];</pre>
- string kitapismi=Request.QueryString["ismi"];%>
- Diğer sayfadan gönderdiğiniz kitap kategorisi değeri:
-
- «Response.Write(kitapismi);%»
-
- </body> </html>



- ASP.Net kontrolleri kullanıldığında başka bir sayfaya yönlendirme genellikle yapılmaz. Aynı sayfa içerisinde girilen veriler değerlendirilir. Gönderme yapılacağı zaman yine a href tagında olduğu gibi ? işareti ile değerler gönderilir.
- Response.Redirect("default2.aspx?isim=" + TextBox1.Text +
 "&soyisim=" + TextBox2.Text + "&id=" + TextBox3.Text);
- ile taşınan değişkenlere diğer sayfadan ulaşmak için Request.QueryString yazılır. QueryString'in içerisindeki verilere ise "indis" veya "Key: değişken adı" değeri üzerinden şu yöntemler kullanılarak ulaşılabilir:

- //Key: değişken adı değeri üzerinden ulaşmak için:
- string str = Request.QueryString["Key"];
- string str = Request.QueryString.Get ("Key");
- //İndis üzerinden ulaşmak için: string str = Request.QueryString[0];
- string str = Request.QueryString.Get(0);

- Tüm verilere Request.QueryString'in Count özelliği ile döngü kullanarak ulaşabiliriz. Çünkü QueryString'in 'Request.QueryString.Values' adlı değerleri tutan bir özelliği bulunmamaktadır.
- for (int i = 0; i < Request.QueryString.Count; i++)
 {</pre>
- DropDownList1.Items.Add(Request.QueryString[i]);
- }

- QueryString'de, 'Key' değerleri farklı olmak zorunda değildir. 'Key' değerlerinin aynı olacağı durumlarda şuna dikkat edilmelidir:
- //default.aspx.cs
 protected void Page_Load(object sender, EventArgs e) {
 Response.Redirect ("default2.aspx? 1=bir &1=iki &1=üç"); }
- //default2.aspx.cs
- protected void Page_Load(object sender, EventArgs e)
- { for (int i = 0; i < Request.QueryString.Count; i++)</pre>
- {DropDownList1.Items.Add(Request.QueryString[i]); }
- }
- Bu şekilde yazdığımız taktirde karşımıza şu sonuç çıkacaktır:
- Burada aynı 'Key' değerine sahip veriler tek bir string olarak alınmıştır.



- Bu string değerlerini birbirinden ayırmak için GetValues metodu kullanılır:
- GetValues metodu parametre olarak **'Key'** veya **'indis'** almaktadır.
- //default2.aspx.cs
- protected void Page_Load(object sender, EventArgs e)



- {
- foreach (string str in Request.QueryString.GetValues("1"))
- { DropDownList1.Items.Add(str); }
- }
- •

Böylelikle QueryString içerisindeki aynı 'Key' değerine sahip verileri birbirinden ayırılmış olur.

- Response.Redirect yerine Server.Transfer metodunu kullanmak:
- //default.aspx.cs
- protected void Button1_Click(object sender, EventArgs e)
 {Server.Transfer("default2.aspx?id=Transfer Metodu", true);
- //Buradaki ikinci özellik ise bool bir değer alıp QueryString gibi diğer sayfaya geçebilecek nesnelerinin gönderilip gönderilmeyeceğini belirtir.



• //default2.aspx.cs

• }

• protected void Page_Load(object sender, EventArgs e)
{ Label1.Text = Request.QueryString.Get(0); }



Transfer Metodu

- QueryString'e eklenemeyen karakterler
- QueryString'e, '#', '+', '&' gibi bazı semboller eklenirken hatalar olmaktadır. Bu hataları önlemek için Server.UrlEncode metodu kullanılmalıdır.
- protected void Page_Load(object sender, EventArgs e)
- { Label1.Text = Request.QueryString.Get(0); }
- protected void Button1_Click(object sender, EventArgs e)
- {
- string encoded = Server.UrlEncode(TextBox1.Text);
- Response.Redirect("default2.aspx?name=" + encoded);
- }

- Form ile gönderilen bilgilerin Request.Querystring ile alınması
- Querystring ile form değişkenlerini işleyebilmek için, formun "get" metodu ile oluşturulması gerekiyor.
- •
- <%@ Page Language="C#" %>
- <html ><head id="Head1" runat="server">
 <title></title></head>
- <body>
- HTML formları kullanılarak Request.QuerySting yöntemi
- <form action="yonlendirilen.aspx" method="get« > Lütfen
 isminizi giriniz:
- <input type="text" name="isim">

- <input type="submit" value="Gönder">
- </form>
- </body> </html>

🕙 http://10.6.2.77/WebSite/Proje/Default.aspx 🔳 🗖 🔀			
Dosya Düzen Görünüm Sık Kullanılanlar Araçlar Yardım 🛛 🦧			
🕞 Geri 🔹 🕥 🚬 😰 🏠 🔎 Ara			
Adres 🙆 http://10.6.2.77/WebSite 💟 💽 Git 🛛 Bağlantılar 🎽 📆 🔹			
MTML formları kullnılarak Request.QuerySting yöntemi			
Lütfen isminizi giriniz: Bilgisayar Öğrencileri			
Gönder			
ど Bitti			

- <%@ Page Language="C#" %>
- <html> <head id="Head1" runat="server"> </head>
- <body>
- <% Response.Write ("Merhaba: "+ Request.QueryString["isim"]+ "");
- Response.Write ("
 Bugün nasılsınız?");
- </body>
- </html>



- <%@ Page Language="C#" %>
- <html > <head id="Head1" runat="server"> <title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title></title>
- <script runat="server">
- protected void Button1_Click(object sender, EventArgs e)
- {
- Response.Redirect("yonlendirilen.aspx?isim=" + TextBox1.Text);
- }
- </script>
- </head>
- <body>
- ASP.Net formları kullanılarak Request.QuerySting yöntemi
- //dikkat action yok onun yerine Response.Redirect kullanılmış
- <form id="Form2" method="get" runat="server"> Lütfen isminizi giriniz:
- <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
- <asp:Button ID="Button4" runat="server" Text="Gönder"
 OnClick="Button1_Click" />
- </form> </body> </html>

🕙 http://10.6.2.77/WebSite/Proje/Default.aspx - Microsoft I 🔳 🗖	×		
Dosya Düzen <u>G</u> örünüm Sık Kullanılanlar <u>A</u> raçlar <u>Y</u> ardım			
🕝 Geri 🔹 📀 👻 😰 🏠 🔎 Ara 🤺 Sik Kullanılanlar 🤗	»		
Adres 🕘 http://10.6.2.77/WebSite/Proje/Default. 💽 🄁 Git 🛛 Bağlantılar 🎽 📆	•		
ASP.Net formları kullnılarak Request.QuerySting yöntemi			
Lütfen isminizi giriniz: Bilgisayar Öğrencileri			
Gönder	~		
🕘 Bitti 🤍 🔮 Internet			

- <%@ Page Language="C#" %>
- <html > <head id="Head1" runat="server"> </head>
- <body>
- Request.QueryString["isim"]+ "");
- Response.Write ("
> Bugün nasılsınız?");
- </body>
- </html>



Request.Form

- Request.Form özelliği kullanarak değer taşıma yapıldığında form metodu "get" yerine "post" olarak seçilir.
- <%@ Page Language="C#" %>
- <html> <head id="Head1" runat="server">
 <title></title> </head>
- <body>
- HTML formları kullanılarak Request.Form yöntemi
- <form action="yonlendirilen.aspx"
 method="post">Lütfen isminizi giriniz:
- <input type="text" name="isim">

- <input type="submit" value="Gönder">
- </form>
- </body> </html>

• yonlendirilen.aspx

- <%@ Page Language="C#" %><html>
- <head id="Head1" runat="server"> </head>
- <body>
- <%Response.Write ("Merhaba: "+ Request.Form["isim"]+ "");
- Response.Write ("
 Bugün nasılsınız?");
- </body>
- </html>

- ASP. Net formları ile Request.Form özelliğini kullanma;
- <%@ Page Language="C#" CodeFile="Default.aspx.cs" Inherits="_Default" %>
- <html><head> <title></title></head><body>
- <form id="Form2" method="post" runat="server">
- <asp:Label id="Label2" runat="server" >Ad-Soyad</asp:Label>
- <asp:Label id="Label3" runat="server">\$ehir</asp:Label>
- <asp:Label ID="Label4" runat="server">Öğrenci</asp:Label>
- <asp:TextBox id="TextBox1" runat="server"> </asp:TextBox>
- <asp:DropDownList id="DropDownList1" runat="server" >
- <asp:ListItem Value="ELAZIĞ"></asp:ListItem>
- <asp:ListItem Value="MALATYA"></asp:ListItem>
- <asp:ListItem Value="SİVAS"></asp:ListItem>
- <asp:ListItem Value="RİZE"></asp:ListItem>
- </asp:DropDownList>

- <asp:RadioButton id="RadioButton1" Text="Evet" Checked="True" GroupName="secim" runat="server" > </asp:RadioButton>
- <asp:RadioButton id="RadioButton2" Text="Hay1r" GroupName="secim" runat="server" > </asp:RadioButton>
- <asp:Button id="Button4" runat="server" Text="GİRİŞ"
 OnClick="Button1_Click"></asp:Button>
- </form>
- </body>

</html>

Default.aspx.cs

```
using System;
   using System.Web;
   using System.Web.Security;
   using System.Web.UI;
   using System.Web.UI.WebControls;
٠
   using System.Web.UI.WebControls.WebParts;
•
   using System.Web.UI.HtmlControls;
   using System.IO;
    public partial class Default : System.Web.UI.Page
•
protected void Button1 Click(object sender, EventArgs e)
     {
        Response.Write("<b>Form Alinan Form degerleri</b><br>);
        foreach (String s in Request.Form.AllKeys)
•
        {
          if ((s != "__VIEWSTATE") && (s != "__EVENTVALIDATION"))
           { Response.Write(s + "=" + Request.Form.Get(s) + "<br>); }
                    Response.Write(Request.Form["TextBox1"] + "<br>");
        }
            11
     }
   }
                                        // if saturi olmasa form elemanlarinin sadece denetim
```

elemanları, değerleri ve isimleri değil VIEWSTATE ve EVENTVALIDATION ait değerler ve isimleri ekrana yazdırılırdı.

🗿 http://10.6.2.77/WebSite/Proje/Default.aspx - Microsoft 💷 🗖 🔀				
<u>D</u> osya Dü <u>z</u> en <u>G</u> örünüm <u>S</u> ık Ku	ıllanılanlar Araçlar Yardım 🥂			
🕞 Geri 👻 🕥 👻 🛃	🏠 🔎 Ara 👷 Sik Kullanılanlar 🚱 🂙			
Adres 🕘 http://10.6.2.77/WebSite/	/Proje/Default 💌 🔁 Git 🛛 Bağlantılar 🌺 🔨 🔹			
Ad-Soyad	Resul TANYILDIZI			
Şehir	ELAZIĞ			
Öğrenci	● Evet ○ Hayır			
	GIRIŞ			
🕘 Bitti	😮 Internet			



- Başka Sayfa Kontrollerini Elde Etme (Cross-Page Posting)
- Bazı durumlarda bir sayfadan diğer sayfaya geçişte ana sayfadaki kontrollerin ve bazı yapısal elemanların değerlerini de taşınması gerekebilir.
- Örnek
 - cross_page.aspx
 - <%@ Page Language="C#" %>
 - <html> <head runat="server"> <title>Cross sayfa</title> </head>
 - <body>
 - <form id="forml" runat="server">
 - <div> Ad:<asp:TextBox ID="txtAd" runat="server"></asp:TextBox>

 - <asp:Button runat="server" 1D="cmdPost"
 PostBackUrl="CrossPage2.aspx" Text="Diğer Sayfaya Gönder" />

 - </div> </form> </body> </html>

PostBackUrl, diğer sayfaya yönlendirme yapıyor.

CrossPage2.aspx

- <mark><%</mark>@ Page Language="C#" <mark>%></mark>
- <script runat="server">
- protected void Page_Load(object sender, System.EventArgs e)
- {
- if (PreviousPage != null)
- { TextBox1.Text=((TextBox) PreviousPage.FindControl ("txtAd")).Text;
- </script>
- <html ><head runat="server"> <title></title> </head> <body>
- <form id="forml" runat="server"> <div>
- <asp:Label ID="lbl" runat="server" Text="Label"></asp:Label>

- <asp:TextBox ID="TextBox1" runat="server" ></asp:TextBox>
- </div> </form> </body>
- </html>

- PreviousPage.FindControl ifadesiyle bir önceki sayfadaki metin kutusunu elde edip tip dönüşümü yapılıyor. Diğer bir ifadeyle yeni sayfada elde edilen kontrolün bir metin kutusu olduğu belirtiliyor. Daha sonra onun metin değeri elde edilir.
- Sayfanın Title elemanını doğrudan PreviousPage.Title ifadesi ile elde edilir. Sayfamızı çalıştırdığımızda cross_page.aspx sayfasındaki değerlerin diğer sayfa tarafım dan elde edildiğini görülür.
- Bu teknikler sayesinde sayfalar arası devamlılığı ve durum yönetimim sağlanabilir.

Request.AcceptTypes özelliği (String dizisi)

- Web tarayıcıları çeşitli MIME(Multipurpose Internet Mail Extensions (Çok amaçlı İnternet Posta Eklentileri)) türlerini desteklemektedir. Bu sayede tarayıcı dosya HTML biçiminde veya gösterebileceği türde bir dosya olup olmadığını algılayarak ne yapması gerektiğini bilmektedir.
- Request.AcceptTeypes özelliği istemcinin tanıyabileceği bir string tipleri listesini içerir. Kullanımına örnek olarak istemci tarafından kabul edilen string tiplerini basan aşağıdaki döngüyü verebiliriz.
- <% String[] arr = Request.AcceptTypes;</pre>
- for (int indx = 0; indx < arr.Length; indx++)</pre>
- { Response.Write("Accept Type " + indx + ": "+ arr[indx] + "
>");}

• <mark>%></mark>

• Çıktı

- Accept Type 0: image/gif
- Accept Type 1: image/x-xbitmap
- Accept Type 2: image/jpeg
- Accept Type 3: image/pjpeg
- Accept Type 4: application/x-shockwave-flash
- Accept Type 5: application/vnd.ms-excel
- Accept Type 6: application/vnd.ms-powerpoint
- Accept Type 7: application/msword
- Accept Type 8: */*

- Request.Cookies Özelliği (HttpCookiesCollection)
- Bu özelliğe çerezlerde denir. Sayfanızın kullanıcı tanımak için sayfaya bağlanan kullanıcının bilgisayarına ufak bir dosya atmasıdır.
- Tek bir çerezi adı ya da index numarası ile almak için Get metodunu kullanabilirsiniz.
- •
- Örneğin tarayıcının kullaniciadi adında bir çerez gönderdiğini varsayalım.
 "kullaniciadi" anahtar kelimesini kullanarak Request.Cookies koleksiyonuna erişmek suretiyle çerezin içeriğini görüntüleyebilirsiniz.
- Response.Write(Request.Cookies["kullaniciadi"]);

- Mevcut olmayan bir çereze erişme girişiminde bulunursanız bir hataya yol açarsınız. Bu problemi önlemek için çerezin olup olmadığını kontrol edebilirsiniz.
- bool keyExists(String aKey)
- {

{

}

- HttpCookieCollection cerez2 = Request.Cookies;
- System.Collections.IEnumerator e = cerez2.GetEnumerator();
- while (e.MoveNext())
- if (e.Current.Equals(aKey))
- return (true);

```
return (false);
```

• Bunun bir başka yolu da aşağıdaki gibidir.

```
• bool keyExists2(String aKey)
• {
     int i;
     String[] keys; keys = Request.Cookies.AllKeys;
     for (i = 0; i < keys.Length; i++)</pre>
     {
       if (keys[i] == aKey)
            return (true);
     }
     return (false);
• }
```

 Çerezin alt anahtar ve değerleri Values özelliğini kullanarak elde edebiliriz. Fakat çerez var fakat alt anahtarlara sahip değilse hata verir.
 Bunu önlemek için HasKeys özelliğini kullanarak çerezin alt anahtarlarının olup olmadığını sınayabiliriz. Aşağıda bir çerez ve alt anahtarlarının oluşturma örneği verilmiştir.

```
void cerezolustur()
{
  HttpCookie cerez = new HttpCookie("Items", "Test");
  int i;
  //alt anahtar oluşturma
  for (i = 1; i < 10; i++)
  {
   cerez.Values.Add(i.ToString(), "Item");
</pre>
```

- }
- Response.AppendCookie(cerez); // Çerezi gönder
- }

- void showCookie()
- { HttpCookie cerez1; int i; String[] keys;
- HttpCookieCollection cerezler = Request.Cookies;
- cerez1 = cerezler.Get("Items");

```
• if (cerez1.HasKeys)
```

}

```
Response.Write(cerez1.Values.AllKeys.Length.ToString() + "<br>'');
```

10 =Test 1=Item 2=Item 3=Item 4=Item 5=Item 6=Item 8=Item 9=Item

```
keys = cerez1.Values.AllKeys;
```

```
for (i = 0; i < keys.Length; i++)
Response.Write(keys[i] + "=" + cerez1.Values[i] + "<br>");
```

else { Response.Write("bu cerez yok"); }

```
Response.Write(cerez1.Value);
```

Test&1=ltem&2=ltem&3=ltem&4=ltem&5=ltem&6=ltem&7=ltem&8=ltem&9=ltem

- Örnek: Çerezlerin var olup olmadığının kontrolü
- Default.aspx
- <%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>
- HTML> <HEAD>
- <script runat="server">
- private void Page_Load(object sender, System.EventArgs e)
- {Response.Cookies["okul"].Value="Firat";}
- private void sayfayagit(object sender, System.EventArgs e)
- { Response.Redirect("yonlendirilen.aspx");}
- </script> </HEAD>
- <body>
- <form id="Form2" method="post" runat="server">
- <asp:Button id="Button1"runat="server" OnClick="sayfayagit"
 Text="Çerez Gönder" ></asp:Button> </form>
- </body>
- </HTML>

🖉 WebForm1 - Microsoft Internet Explorer 💦 🔲	
Dosya Düzen <u>G</u> örünüm <u>S</u> ik Kullanılanlar <u>A</u> raçlar y ^{>>}	2
🕝 Geri 👻 🕑 👻 📓 🏠 🔎 Ara	»
Adres 🗃 http://10.6.2.77/We 🌱 ラ Git 🛛 Bağlantılar 🎽 🤅	2 -
Çerez Gönder	~
🕘 Bitti 🥥 Internet	

- yonlendirilen.aspx
- <%@ Page Language="C#" AutoEventWireup="true" CodeFile="yonlendirilen.aspx.cs" Inherits="yonlendirilen" %>
- <html > <head id="Head1" runat="server">
- <script runat="server">
- private void Page_Load(object sender, System.EventArgs e)
- { bool x= keyExists("okul");
- if (x == true)
- Response.Write(Request.Cookies["okul"].Value.ToString());
- else Response.Write("Çerez yok");
- if (y == true)
- Response.Write(Request.Cookies["renkler"].Value.ToString());
- else Response.Write("Çerez yok");
- showCookie();
- }

bool y = keyExists2("renkler");

- // Çerez kontrol
- bool keyExists(String aKey)
- {
- HttpCookieCollection cerez2 = Request.Cookies;
- System.Collections.IEnumerator e = cerez2.GetEnumerator();
- while (e.MoveNext())

```
• {
```

if (e.Current.Equals(aKey))

```
return (true);
```

```
• }
```

```
• return (false);
```

```
• }
```

- bool keyExists2(String aKey)
- { int i;
- String[] keys;
- keys = Request.Cookies.AllKeys;
- for (i = 0; i < keys.Length; i++)

```
• {
```

```
if (keys[i] == aKey)
```

• return (true);

```
• }
```

```
• return (false);
```

```
• }
```

```
•
```

- </script> <title>cerezoku</title>
- </head>
- <body> </body>
- </html>


- Request.Files Özelliği (HttpFileCollection)
- Request.Files, istemci tarafından gönderilen dosya adlarından oluşan bir koleksiyon gönderir.
- Dosya upload olaylarında kullanılan bir özelliktir. Dosya upload edildikten sonra dosyanın boyutu içeriği hakkında bilgi alınabilir.



- <form id="Form2" runat="server">
- <asp:FileUpload ID="FileUpload1" runat="server" />
- <asp:Button ID="Button4" runat="server" Text="Gönder"
 OnClick="Button1_Click" />
- </form>

- protected void Button1_Click(object sender, EventArgs e)
- {if (FileUpload1.HasFile) {
- string [] a ;
- HttpFileCollection File;
- File = Request.Files;
- a = File.AllKeys;

Response.Write(a.Length);

- for (int i = 0; i < a.Length; i++)</p>
- Response.Write("File: " + Server.HtmlEncode(a[i]) + "
");
- Response.Write(" size = " + File[i].ContentLength + "
");
- Response.Write(" content type = " + File[i].ContentType + "
>");
- }
- FileUpload1.SaveAs(Server.MapPath("dizin") + "\\" +FileUpload1.FileName);}

File:FileUpload1 size = 1122 content type = text/plain

- Request.ServerVariables Özelliği (NameValueCollection)
- Request.ServerVariables koleksiyonu istemciler tarafından her taleple birlikte gönderilen çok sayıda kullanışlı değer içerir.
- Bu değerlerden bazılarına sıklıkla gerek duyarız. Sıklıkla kullanılanlardan bazıları aşağıda verilmiştir.

APPL_PHYSICAL_PATH	Uygulamanızın fiziksel dizin yolu		
LOCAL_ADDR	Uygulamanıza ev sahipliği yapan IP adresi.		
LOGON_USER	Tarayıcıyı kullanan kişin ağ kullanıcı adıdır.		
PATH_INFO	Talep edilen dosyanın tam fiziksel dizin yolu.		
SERVER_NAME	Kullanıcı tarafından talep edilen sunucunun adı.		
SERVER_PORT	IP port numarası.		
HTTP_USER_AGENT	Talepte bulunan istemcinin markası ve sürümü.		

- HTTP Server Değişkenleri Kolleksiyonunun elemanları şöyle sıralanır:
- AUTH_PASSWORD Kullanıcı Web Server'a kullanıcı adı/parola yöntemiyle bağlanabiliyorsa, kullanılan parola

AUTH_TYPE Kullanıcı Web Server'a kullanıcı adı/parola yöntemiyle bağlanabiliyorsa, kullanılan yetkilendirme yöntemi

- **AUTH_USER** Kullanıcı Web Server'a kullanıcı adı/parola yöntemiyle bağlanabiliyorsa, kullanıcı adı
- **CERT_COOKIE** Kullanıcı siteye bağlanmak için yetkilendirme sertifikası kullanıyorsa kendisine verilen kimlik (ID)
- **CERT_FLAGS** Sertifikanın varlığını gösteren bit değeri
- **CERT_ISSUER** Sertifika varsa veren kurum
- **CERT_KEYSIZE** Secure Socket Layers kullanılıyorsa, bağlantı için anahtar değeri
- **CERT_SECRETKEYSIZE** Özel anahtar değeri
- **CERT_SERIALNUMBER** Sertifika seri no.
- CERT_SERVER_ISSUER Sertifikayı veren merci
- **CERT_SERVER_SUBJECT** Server Sertifikasının "konu" alanı değeri
- **CERT_SUBJECT** İstemci Sertifikasının konu alanı değeri

• CONTENT_LENGTH İstemcinin gönderdiği bilgi yumağının boyutu **CONTENT_TYPE** Ziyaretçiden gelen bilgilerin GET veya POST metoduna göre edindiği tür GATEWAY_INTERFACE Web Server'ın ziyaretçi ile etkileşim arayüzünün adı ve sürümü. Genellikle: CGI/1.1 HTTPS Ziyaretçi ile bağlantı güvenli ise ON, değilse OFF HTTPS_KEYSIZE Secure Sockets Layer için bağlantı anahtar sayısı HTTPS_SECRETKEYSIZE Özel Server sertifikasının gizli anahtar sayısı HTTPS_SERVER_ISSUER Özel Server sertifikasının veren merci HTTPS_SERVER_SUBJECT Özel Server sertifikasının konusu **INSTANCE_ID** Web Server'ın aynı anda kaç kere çalışmakta olduğu **INSTANCE_META_PATH** Şu anda çalışmakta olan Web Server'ın Meta yolu LOCAL ADDR İstemcinin IP numarası **LOGON_USER** İstemci Windows NT sisteminde ise oturum açma adı PATH INFO Çalışmakta olan ASP'nin göreli yolu ve adı PATH_TRANSLATED Çalışmakta olan ASP'nin gerçek yolu ve adı QUERY_STRING İstemcinin gönderdiği bilgi kümesi

REMOTE_ADDR İstemcinin Host'unun (ISS'inin) IP'si
REMOTE_HOST İstemcinin Host'unun (ISS'inin) adı
REMOTE_USER İstemcinin gerçek adı
REQUEST_METHOD İstemciden bilgi isteme yöntemi (GET veya POST)
SCRIPT_NAME Çalışmakta olan ASP'nin adı
SERVER_NAME Sunucu'nun adı
SERVER_PORT Sunucuya bağlantının geldiği TCP kapı numarası
SERVER_PORT_SECURE TCP kapısı güvenli ise 1, değilse 0
SERVER_PROTOCOL Server'ın çalıştırdığı HTTP'nin sürümü
SERVER_SOFTWARE Server programının adı ve sürümü
URL Şu anda geçerli URL

- <%@ Page Language="C#" CodeFile="Default.aspx.cs" Inherits="_Default" %>
- <html ><head runat="server"> <title>Untitled Page</title></head>
- <body> <form id="form2" runat="server"> <div>
- Kullandığın Tarayıcı:
- <%Response.Write (Request.ServerVariables["Http_user_agent"]);%>
- IP Adresin:
- «Response.Write(Request.ServerVariables ["remote_addr"]); %
- DNS :
- <%Response.Write (Request.ServerVariables["remote_host"]); %>
- Sayfayı çağırmak için kullandığın metot:
- Kesponse.Write (Request.ServerVariables ["request_method"]);%>
- Sunucunun Domain İsmi:
- Sunucunun Port Numaras::
- <%Response.Write(Request.ServerVariables["server_port"]); %>
- Sunucunun Kullandığı Yazılım:
- </div> </form></body></html>

🗿 Untitled Page - Microsoft Internet Explorer 📃 🗖	×
Dosya Düzen Görünüm Sık Kullanılanlar <u>A</u> raçlar <u>Y</u> ardım	ľ
🕞 Geri 🔹 🕑 👻 🛃 🏠 🔎 Ara 🤺 Sik Kullanılanlar 🧐 🎯 -	»
Adres 🕘 http://10.6.2.77/WebSite/WebSite/proje2/Default. 🔽 🔁 Git 🛛 Bağlantılar 🎽 😨	r
Kullandığın Tarayıcı: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322; .NET CLR 1.0.3705; .NET CLR 2.0.50727)	~
IP Adresin: 10.6.2.77	
DNS : 10.6.2.77	
Sayfayı çağırmak için kullandığın metot: GET	
Sunucunun Domain ismi: 10.6.2.77	
Sunucunun Port numarası: 80	
Sunucunun Kullandığı Yazılım: Microsoft-IIS/5.1	
	~
🕘 Bitti 🧶 🔮 Internet	



- Request.UserLanguages Özelliği (ShortedStringArray)
- Bu özellik kullanıcının dil tercihlerini listeleyen bir string dizisi içerir.
- <mark><%@</mark> Page Language="C#" <mark>%></mark>
- <html ><head runat="server"> title>Untitled Page</title></head>
- <body>
- <form id="form2" runat="server">
- div>
- Kullanılan Dil:
- <%int count; String[] userLang = Request.UserLanguages;</pre>
- for (count = 0; count < userLang.Length; count++)</pre>
- </div>
- </form>
- </body> </html>

- Ödev :
- Request Nesnesine ait özellik ve metotların bir kısmı ders notlarında mevcuttur.
- Gelecek ders için :
- Verilmeyen Request Nesnesinin diğer özellik ve metotları öğrenilecek ve birer örnek üzerinde değerlendirilecek.

- ASP.Net için ziyaretçi sunucuya bağlandığı an, bir oturum (session) başlanmıştır. Bu durumda sunucu session nesnesi yaratarak, ziyaretçiyi oturumdan çıkana kadar izler.
- Tarayıcı ile sunucu arasında ASPSessionID üzerinden kurulan ilişki, tarayıcı Session'nun oturumunu kapatana kadar devam etmektedir.
- Session zaman aşımına uğradıktan sonra, eğer kullanıcı sizin uygulamanıza yeniden dönerse, ASP.NET yeni bir Session oluşturacak ve yeni bir ASPSessionID verecektir.
- Her değişken tanımlaması en fazla o sayfa genelinde etkin olur. Bütün sayfalarda geçerli olabilecek değişkenleri, session nesnesi ile tanımlayabiliriz.

• Kullanımı:

Session.Add(isim,değer);

• veya

Session["isim"]=değer

- <<mark>%</mark>@ Page Language="C#" CodeFile="Default.aspx.cs" <mark>%></mark>
- <script runat="server">
- protected void Button1_Click(object sender, EventArgs e)
- { Response.Redirect("session.aspx"); }
- </script>
- <html > <body>
- <% Session["Bölüm"] = "Bilgisayar";</pre>
- Session.Add("Okul", "Fırat Üniversitesi");
 %>
- <form id="form2" runat="server">
- <asp:Button ID="Button4" runat="server"</pre>
- Text="Oturum Değişkeni" OnClick="Button1_Click" />
- </form> </body> </html>



- <%@ Page Language="C#" CodeFile="session.aspx.cs" %>
- <html> <head runat="server"> </head>
- <body>
- <form id="form2" runat="server">
- <% Response.Write("Bölüm : "+Session["Bölüm"]+"
);
- Response.Write("Okul : "+ Session["Okul"]);
- <mark>%></mark>
- </form>
- </body>
- </html>



 Session değişken içeriği dinamik olarak değiştirilebilir . Sayfa her çağrıldığında session değişken içeriği farklı değerler alabilir.

```
• protected void Page_Load(object sender, EventArgs e)
• {
• if (Session["sayac"] != null)
• Session["sayac"] = (int)Session["sayac"] + 1;
• else
• Session["sayac"] = 1;
• Response.Write("Sayac" + Session["sayac"].ToString() + "<br>>");
• }
```

Sayfa her yenilediğinde sayaç değeri artacaktır.

Session.Timeout Özelliği

- Bir oturum nesnesi oluşturulduğunda, eğer ziyaretçi belirli bir süre zarfında browser'ında hiçbir hareket veya yenileme yapmamışsa, sunucu oturum nesnesini otomatik olarak kapar (varsayılan değeri 20 dakikadır.).
 - Session.Timeout = dakika değeri;
- Session.Timeout özelliğinin bulunduğu sayfada dakika değeri kadar işlem yapılmadan beklenirse Session değişken içerikleri silinecektir.
- Session nesnesini Session. Abandon ifadesi ile aniden sonlandırılabilir. Abandon ile geçici Cookie, dolayısı ile Session silinir.

Session.Contents Özelliği

- Session nesnesinin içeriğinin elde edilmesi için kullanılan bir özelliktir.
- <%
- for(int i=0; i<Session.Contents.Count;i++)</pre>
- Response.Write(""+Session.Contents[i]+"
");
 %>

Session Değişkenleri Temizlemek

- Session değişkenini birden fazla şekilde oluşturabildiğimiz gibi bunları yok etmek içinde çeşitli yollar vardır.
- Null'a eşitlemek, Remove metodunu kullanmak.
 - Session["sayac"]=null
 - Session.Contents.Remove("sayac")
 - Session.Remove("sayac")
 - Session.RemoveAll
 - Session.RemoveAt(int)

• Örnek:

- <%@ Page Language="C#" %>
- <html > <head> <title>Session</title> </head>
- ody>
- <form action ="session.aspx" method="post">
- Lütfen formu doldurunuz:
- AD :<input type="text" name="ad" size ="10" />

- YAS: <input type="text" name="yas" size="10"/>

- <input type="submit" value="GÖNDER"/>
- </form>
- </body>
- </html>

🕘 Session	ı - Microsoft I	nternet 📒	
<u>D</u> osya D	ü <u>z</u> en <u>G</u> örünüm	<u>S</u> ik Kullanilank 🏾 🎗	* 🥂
Geri	• 🕥 •	2 🖒	**
Adr <u>e</u> s 🛃 H	nttp 💌 🔁 Git	Bağlantılar 🎽	🔁 -
Lütfen fo AD : Ra YAŞ: 10	ormu doldurun bia	uz:	~
GOND	ER		~
8		Internet	

- <%@ Page Language="C#" CodeFile="session.aspx.cs" %>
- <html ><head > <title>Untitled Page</title> </head>
- <body>
- Session.Timeout=1;
- string ad = Request.Form["ad"];
- int yas = Int16.Parse(Request.Form["yas"]));
- Session["ad"]=ad;
 Session.Add("yas",yas);
- Session["yetiskin_kitap"]="yetişkin kitapları";
- Session["cocuk_kitap"]="çocuk kitapları";
- if ((int)(Session["yas"])<= 18)</pre>
 - Session.Contents.Remove("yetiskin_kitap");
- else
- Session.Contents.Remove ("cocuk_kitap");
- for(int i=0;i<Session.Contents.Count;i++)</pre>

Response.Write(Session.Contents[i]+ "
");

- </body>
- </html>

🕙 Untitled Page - Microsoft In 🔳 🗖	×
Dosya Düzen Görünüm Sık Kullanılar 🌺 🧂	F
🌀 Geri 🝷 🐑 🔺 🛃 🏠	»
Adr <u>e</u> s 💰 htl 🗙 🄁 Git 🛛 Bağlantılar 🎽 📆	•
Rabia 10 çocuk kitapları	^
	<u> </u>
🥑 Internet	

🕙 Untitled Page - Microsoft In 🔳 🔲	×
<u>D</u> osya Düzen <u>G</u> örünüm <u>S</u> ık Kullanılar 🌺 🥻	
🌀 Geri 👻 🕑 👻 📝 🏠	>>
Adr <u>e</u> s 🙆 hti 💌 🛃 Git 🛛 Bağlantılar 🎽 🔁 🥆	•
Rabia 10 çocuk kitapları	
🔹 🚺 😨 Internet	

Cookie'siz Session Kullanımı

- Bazı kullanıcılar, güvenlik kaygılarından dolayı bilgisayarlarında cookie oluşturulmasına izin vermeyebilmektedirler. ASP de cookie olmayınca session değişkenleri de kullanılamıyordu. ASP.NET ile artık cookilere gerek kalmadan, session değişkenleri kullanmak mümkündür.
- Yapmamız gereken web.config dosyasınıda cookie kullanılmayacağını belirtmektir.

web.config

- <?xml version="1.0" encoding="iso-8859-9" ?>
- <configuration>
- <system.web>
- <sessionstate cookieless="true" timeout="10" />
- <globalization reguestEncoding="iso-8859-9" responseEncoding="iso-8859-9"
- fileEncoding="iso-8859-9" culture="tr-TR" uiCulture="tr" />
- </system.web>
- </configuration>

- ASP.NET değişkenlerin takibini cookie kullanmadan gerçekleştirmesini cookie kullanarak yaptığından farklı bir mantık izlemez, burada da yine session ID üretir fakat bunu cookie yerine sayfa URL adresine ekler. Başka bir sayfaya herhangi bir geçişte bu ID numarası da aktarılır. Böylece session değişkenlerinin oturum boyunca kullanımında bir süreklilik sağlanır ve sunucu kullanıcıya ait değerleri kullanabilir.
- Eğer uygulamamızı bu haliyle denersek sayfalar arasında gezinirken url bilgilerinin aşağıdaki gibi değiştiğini görürüz. Görüldüğü gibi ASP.NET_SessionId değeri otomatik olarak relative url bilgisinin önüne eklenmiştir.



• Ödev :

 Verilmeyen Session Nesnesinin diğer özellik ve metotları öğrenilecek ve birer örnek üzerinde değerlendirilecek.

- Application nesnesi ASP.NET web sitelerinde çalışan uygulamayı temsil eder ve uygulama adına işlemler yapabilmemizi sağlar.
- Session nesnesi ziyaretçi, sunucuya bağlı kaldığı sürece bütün sayfalar için geçerlidir. Her oturumda değişmeyen ve bütün ziyaretçiler için geçerli evrensel tanımlara veya değerlere ihtiyaç duyabiliriz.
- Application, bütün kullanıcılar için geçerli değişkenleri oluşturan, tanımlayan bir nesnedir.

- Application değişkeninin oluşturulması aynen Session değişkenleri gibidir.
- Application["Değişken_ismi"]="Değişken_değeri";
- Değişkeni elde etmek için :
- Response.Write(Application["değişken_ismi"]);

"Oyun_giris.aspx.cs"

• }

- protected void Page_Load(object sender, EventArgs e)
- {
 Application.Add("Oyun", 0);

- "Oyun_giris.aspx"
- <mark><%</mark>@Page Language="C#"CodeFile="Oyun_giris.aspx.cs" <mark>%></mark>
- <html > <head> <title>Session</title> </head>
- <body>
- <% Response.Write("<h3>oyun sayfamıza girmek ister misiniz?</h3>");
- Response.Write("Oyun sayfamızda su an " + Application["Oyun"] + "oyuncu var.");
- evet oyuna girmek istiyorum.
- </body>
- </html>



- "Oyun.aspx"
- <html><body>
- Kesponse.Write("<h3>Oyunumuza Hoş Geldiniz</h3>");
- Application.Lock();
- if ((int)Application["Oyun"] == 0)
 - Application["Oyun"] = 1;
- else { int deger = (int)Application["Oyun"];

```
deger += 1; Application["Oyun"] = deger; }
```

- Application.UnLock();
- Response.Write ("Şu anda oyunda "+Application["Oyun"] +" kişi var");
 %>
- Oyundan ayrıl
- </body> </html>



- "oyun_cikis.aspx"
- <%@ Page Language="C#" AutoEventWireup="true" CodeFile="oyun_cikis.aspx.cs" Inherits="oyun_cikis" %>
- <!DOCTYPE html >
- <html> <head runat="server"></head>
- <body>
- <% Response.Write("<h3> Oyunumuza katıldığınız için teşekkür ederiz.</h3>");
- Application.Lock();
- if ((int)Application["Oyun"] <= 0) Application["Oyun"] = 0;</pre>
- else { int deger = (int)Application["Oyun"];
 - deger -= 1; Application["Oyun"] = deger; }
- Application.UnLock();
- Response.Write("Şu anda oyunda "+Application["Oyun"]+" kişi kaldı.");
- Oyuna tekrar katil....
- •
- </body>
- </html>

🕘 Untitled Page - Microsoft Internet Explorer 📃				
Dosya Düzen <u>G</u> örünüm <u>S</u> ık Kullanılanlar <u>A</u> raçlar <u>Y</u> ardım	1			
🌀 Geri 👻 💿 🔸 😰 🏠 🔎 Ara tik Kullanilanlar	»			
Adres 🗃 http://10.6.2.77/WebSite/WebSite/r 💙 🄁 Git 🛛 Bağlantılar 🍟 🤇	b - d			
Oyunumuza katıldığnuz için teşekkür ederiz. 🔗				
Şu anda oyunda 2 kişi kaldı.				
<u>Oyuna tekrar katil</u>				
	<u>×</u>			

• Uygulama :

 Verilmeyen Application Nesnesinin diğer özellik ve metotları öğrenilecek ve birer örnek üzerinde değerlendirilecek.

Global.asax (Global Application Class - Genel Uygulama Sınıfı) Dosyası

- ASP.NET sayfası aslında arka planda bir class olarak tutulur ve sayfalarda kullandığımız web (sunucu) kontrollerinin (TextBox, Button gibi) her biri birer class 'tır. Bu class Page adındaki bir class'tan kalıtımsal olarak elde edilir.
- Olay tabanlı bir mimaride çalışıldığı için sayfadaki nesnelerinin belirli bir yaşam döngüsü vardır ve sayfaya her istek geldiğinde sunucuda bu yaşam döngüsü çalışır. Sayfanın Prelnit, Load, PreRender... gibi olayları adım adım gerçekleşerek sayfa üretilir, HTML çıktısına dönüştürülür ve çıktı istemciye gönderilir.
- Programcı olarak yaşam döngüsü içerisindeki bu olayları yakalayıcı metotlarla ele alabilir ve uygun noktalarda sayfaya müdahale edebiliriz.

Global.asax (Global Application Class - Genel Uygulama Sınıfı) Dosyası

- Sayfanın Load olayında çalışan Page_Load metodu çok sık kullandığımız ve sayfa yüklendiği esnada işlemler yapmamızı sağlayan olay yakalayıcı metodumuzdur. Benzer çalışma mantığı TextBox, Calendar, GridView... gibi sunucu kontrolleri içinde geçerlidir.
- Uygulamayı hiyerarşik olarak düşündüğümüz zaman kontroller sayfaları, sayfalar da uygulamayı oluşturmaktadır.
- Application nesnesinin de tıpkı sayfalarda olduğu gibi belirli bir yaşam döngüsü vardır ve bu yaşam döngüsünü programcı olarak ele alabilmemiz, belirli olaylar tetiklendiğinde işlemler gerçekleştirmemiz mümkündür.

Global.asax (Global Application Class - Genel Uygulama Sınıfı) Dosyası

- Sayfanın olaylarını sayfanın code-behind kısmına yazabildiğimiz metotlarda ele alabilmekteyiz.
- Uygulamanın olaylarını ise ele alabileceğimiz yer Global.asax uzantılı dosya olacaktır.
- Global.asax dosyası görsel bir arayüzü olmayan ve çalışan uygulamanın metotlarının yakalanmasını sağlayan dosya olarak ASP.NET web sitesi projelerinde kullanılabilmektedir.
- Application nesnesi aslında temel olarak durum yönetimini(state management) sağlayabilmek ve uygulama genelinde geçerli olacak nesneleri saklamak amaçlı kullanılmamaktadır.
- Application nesnesinin olaylarını (event) ele alarak uygulamayla ilgili belirli aşamalarında birçok önemli işlemde gerçekleştirilebilir. Bu olayları ele alabileceğimiz yer ise projemizin ana dizininde yer alacak olan Global.asax dosyasıdır.
- Global.asax dosyası web projelerinde standart olarak gelmemektedir. Bu dosyayı projemize eklemek için Solution
 Explorer'dan projemize sağ tıklayıp Add New Item seçeneğinden Global Application Class seçeneğini kullanabiliriz.

Add New Item - C:\Ça	lışmalarım\Makale\- I	Makalelerim\Global.asax Dosyasının Etkin Kullanımı\GlobalAsaxProje\				
<u>T</u> emplates:						
Visual Studio insta	lled templates					
🛅 Web Form		🔲 Master Page		🗄 Web User Control		
AJAX Client Beh	avior	AJAX Client Control		🗿 AJAX Client Library		
AJAX Master Pa	age	🔲 AJAX Web Form		🞇 AJAX-enabled WCF Service		
👰 Browser File		🕲 Class		- 🗟 Class Diagram		
🔣 DataSet		🔊 Generic Handler		Global Application Class		
HTML Page		😰 JScript File		LINQ to SQL Classes		
Report		🔊 Report Wizard		🛃 Resource File		
🚵 Site Map		📴 Skin File		间 SQL Server Database		
A Style Sheet		📄 Text File		🞇 WCF Service		
Web Configuration File		🔝 Web Service		🔮 XML File		
🔣 XML Schema		🔊 XSLT File				
My Templates						
Search Online Templates						
A class for handling	g Web Application ever	nts				
<u>N</u> ame:	Global.asax					
Language:	Visual C#	_	☐ Place code in se	parate file		
200			Select master pa	17e		
			T Telescellar	Pr		

- Global.asax dosyası uygulama çalışırken arka planda işlemler yapacağı ve görsel olarak kullanıcılara herhangi bir içerik sunmayacağı için sadece sunucuda çalışacak olan kodlardan oluşur.
- Global.asax dosyası, web uygulamasının çalıştırılmasından itibaren, uygulamada geçerli olan bütün tanımlamaları ihtiva eder.
- .aspx dosyalarından farklı olarak HTML kodu ve sunucu kontrolü barındırmayacağı için sunucuda çalışacak kodlar <script> etiketi içerisinde yer alır. Bu dosyada scriptler, veritabanı bağlantıları session ve application nesneleri gibi sitenin tamamında geçerli olacak yordamlar yazılabilir.

- Dosya açıldığında içerisine eklenmiş olarak gelen beş adet metot yer almaktadır. Bu metotların her birisi aslında Application nesnesinin birer olay yakalayıcı(event handler) metodudur.
- Application_Start, Application_End, Application_Error, Session_Start ve Session_End metotları Application nesnesinin en çok kullanılan metotlarından olduğu için içerilerine her bir metodun hangi olaylar gerçekleştiğinde çalışacağı bilgileriyle birlikte dosyaya eklenmiş olarak gelmektedir.
- Application nesnesinin olayları bu beş metoda bağlanan olaylardan ibaret değildir. Application nesnesinin tüm olaylarının listesini görebilmek için Visual Studio arayüzünü kullanabiliriz. Global.asax dosyası açıkken dosya listelerinin yer aldığı tab'ın hemen altındaki iki seçme listesinden bu listeye erişebiliriz.

Bu listeden herhangi bir olayı seçmemiz durumunda dosyamıza olayın yakalayıcı metodu otomatik olarak eklenir. Yandaki şekilde bu listeye nasıl ulaşabileceğimiz görülmektedir. (1 numaralı listeden Application nesnesi seçildiğinde sağdaki 2 numaralı listede olaylar listelenecektir)



- Application- Start: Bir ASP.NET uygulaması ilk çalıştığı anda tetiklenir. Bu olay siteyi barındıran sunucu -ki bu sunucu IIS olacaktır- başladıktan sonra istemciden gelen ilk istekte tetiklenecektir. Buradaki en önemli durumlardan birisi bu olayın sadece ilk istekte tetikleneceği ve uygulama bir daha başlayana kadar tetiklenmeyeceğidir. Ne zaman ki uygulama, web sunucusu veya sunucu bilgisayarı kapatılıp tekrar açılır, Start olayı o zaman tetiklenecektir.
- Uygulama ilk başladığında belirli kaynakların uygulama için açılması gerekiyorsa, belirli nesnelerin oluşturulması gerekiyorsa veya uygulama başlangıcı bir yerlere kaydedilmek (ör. log dosyasına) isteniliyorsa bu event kullanılabilir.

- Application- End: ASP.NET uygulaması sonlandırıldığında tetiklenen olaydır. Web sitesi durdurulduğunda, IIS veya sunucu bilgisayarı durdurulduğunda ya da kapatıldığında tetiklenir.
- Bu olay içerisinde ise genellikle Start olayında başlatılan, açılan kaynakların, nesnelerin durdurulması ve kapatılması gibi işlemler gerçekleştirilebilir.

- Session_Start: Bir ziyaretçi uygulamanın herhangi bir sayfasını ziyaret ettiğinde tetiklenir.
- Bu olay sadece istemcinin sadece ilk sayfa isteğinde tetiklenir ve sonraki sayfa isteklerinde tetiklenmez. Aslında sitemize bir ziyaretçinin giriş yaptığının habercisidir.
- Session'ın timeout süresinin geçmesi durumunda kullanıcı bir talepte daha bulunursa bu olay istemcinin yeni bir kullanıcı olarak sitemize geldiğini düşünerek tekrar tetiklenecektir. Kullanıcının sitemizi ilk ziyareti esnasında gerçekleştirilebilecek işlemler bu olay dahilinde gerçekleştirilebilir.

 Session_End: Sitemizi ziyaret eden kullanıcı Session bilgisinin timeout olması durumunda tetiklenir. Kullanıcı sitemizde oturum açtıktan sonra Session'ın timeout değeri süresince bir işlem yapmazsa bu olay tetiklenir. Genellikle Session_Start'ta yapılan işlemlerle ilişkili olarak kullanıcıların çıkışlarıyla ilgili işlemler gerçekleştirilebilir.

- Error: Uygulamanın herhangi bir dosyasında, herhangi bir anında hata oluşursa bu olay tetiklenir.
- Bu hata sunucu tarafında çalışacak kodlardan dolayı oluşan bir istisna (exception) olabileceği gibi, HTTP durum kodlarından (404-dosya bulunamadı-, 403-erişim yok-... gibi) oluşan herhangi bir hata da olabilir. Özellikle log dosyalarında hata kayıtlarının tutulması işlemi için uygulamamızın en ideal noktası Application'ın Error olayıdır.

- BeginRequest: Uygulamadaki herhangi bir dosyaya talep(request) geldiğinde tetiklenir.
- Gelen talebin durumuna göre farklı işlemler yapılmak istenirse(farklı sayfalara yönlendirme, farklı içerik görüntüleme, sayfaya veya dosyaya erişimi engelleme... gibi) bu olay ele alınabilir. URL Rewriting veya belirli dosyaya erişimin kontrol edilmesi gibi durumlarda kullanılabilir.

Global.asax

- <%@ Application Language="C#" %>
- <script runat="server">
- void Application_Start(object sender, EventArgs e)
- { // Application uygulaması çalıştığında devreye girer.
 - Application.Add("ToplamZiyaretci",0);
- }
- void Application_End(object sender, EventArgs e)
- { // Application uygulaması kapatıldığında devreye girer.
- }
- •
- void Application_Error(object sender, EventArgs e)
- { // Uygulamada beklenmedik bir hata meydana geldiğinden çalışır.
- }

- void Session_Start(object sender, EventArgs e)
- { // Yeni bir oturum başladığında çalışır.
- if ((int)Application["ToplamZiyaretci"] == 0)

```
    Application["ToplamZiyaretci"] = 1;
```

```
• else
```

• { int deger = (int)Application["ToplamZiyaretci"];

```
• deger += 1;
```

```
Application["ToplamZiyaretci"] = deger;
```

```
• }
```

```
• }
```

```
•
```

```
void Session_End(object sender, EventArgs e)
```

```
{ /* Oturum kapatıldığında devreye girer*/
```

```
int deger = (int)Application["ToplamZiyaretci"];
```

```
deger -= 1;
```

```
Application["ToplamZiyaretci"] = deger;
```

```
• }
```

```
</script>
```

- Örnek: Uygulamada oluşacak hataların log dosyasına kaydedilmesi (Error olayının kullanımı)
- Bir web uygulamasının hayata girmesinden sonra raporlanacak en önemli bilgilerden birisi de uygulamada oluşan hataların kaydedilmesidir.
- Geliştirme ve test aşamasında göremediğimiz bir çok hata kullanıcıların sayfalarımızı ziyaretleri esnasında oluşabilir. Application nesnesine ait Error olayı uygulamamızın herhangi bir dosyasında oluşacak bir hatada otomatik olarak tetiklenecektir.
- Önemli olan hatanın oluşmasını yakalamak olacağı gibi, hatanın sebebi ve hangi dosyada oluşacağı da bizim için önemli bilgilerdir.

- Örnek: Uygulamada oluşacak hataların log dosyasına kaydedilmesi (Error olayının kullanımı)
- Request nesnesinin Path özelliği o an hatanın oluştuğu sayfanın yol bilgisini, Server nesnesinin GetLastError metodu ise oluşan hatanın Exception tipinden bilgilerini getirecektir. Aşağıdaki örnek kod parçasında Global.asax dosyasına eklenecek Application_Error metodu ile txt uzantılı bir metin dosyasına kaydedebiliriz.

- Örnek(devam): Error olayının kullanımı
- Global.asax
- <%@ Application Language="C#" %>
- <%@ Import Namespace="System.IO" %>
- <script runat="server">
- void Application_Error(object sender, EventArgs e)
- {//Oluşan hatanın bilgilerini Hatalar.txt adındaki bir dosyaya kaydediyoruz
- StreamWriter sw = new StreamWriter (Server.MapPath ("~/Hatalar.txt"), true);
- sw.WriteLine(DateTime.Now.ToString());

- Örnek(devam): Error olayının kullanımı
 - //Server nesnesini GetLastError metodu sunucuda oluşan son hatayı Exception tipinden getirir. Bu da şu an oluşan hata olacaktır
 - if (Server.GetLastError().InnerException != null)
 - sw.WriteLine(Server.GetLastError().InnerException.Message);
 - else
 - sw.WriteLine(Server.GetLastError().Message);
 - //Request nesnesinin Path özelliği şu an istekte bulunulan sayfanın yol bilgisini getirir
 - sw.Write(Request.RawUrl != null ? Request.RawUrl : "");
 - sw.WriteLine();
 - sw.Close();
 - }
 - </script>

- Örnek: Uygulama genelinde geçerli olacak ve heryerden erişilebilecek nesneler oluşturmak (Application_Start olayının kullanımı)
- Global.asax
 <%@ Application Language="C#" %>
- <%@ Import Namespace="System.Collections.Generic" %>
- <script runat="server">
- void Application_Start(object sender, EventArgs e)
- string[] renkler = new string[] {"Beyaz", "Sarı", "Pembe", "Mor" };
- Application["renkListesi"] = renkler;
- Application["resimKlasor"] = Server.MapPath("v2/images/resimler");
- •
- </script>

}

- Global.asax dosyası, web sitemizdeki herhangi bir dosyaya gelen istekleri yakalayabilme ve isteğin durumuna göre de farklı işlemler yapabilme imkanı tanır. Application'ın BeginRequest adındaki olayı sitemizdeki tüm dosyalara gelen isteklerde tetiklenecektir.
- Dolayısıyla bu olayın metodu içerisinde yazacağımız kodlarla gelen isteğe(yani Response nesnesine) erişebiliriz. İstenen dosyanın uzantısına göre, dosyanın yoluna göre, içerdiği QueryString bilgisine göre işlemler yapılabileceği gibi, kullanıcının tarayıcısına veya IP numarasına bakılarak da farklı işlemlerin gerçekleştirilmesi bu olay içerisinde ele alınabilir.

- Örneğin: Sitenizde yoğun şekilde QueryString bilgilerini toplayarak veritabanında sorgular çalıştırıyorsunuz. Dolayısıyla kullanıcıların QueryString içerisine tehlikeli olabilecek SQL komutları eklemesi(SQL Injection) sizi tehdit edecektir. Böyle bir durumda elbetteki en kolay yöntem parametre kullanmaktır, fakat parametreyi kullanmayı unuttuğunuz bir sayfanız varsa?
- İşte böyle bir durumda projenin sağlığını biraz daha garanti altına almak istiyorsak Global.asax dosyasında gelen isteğin QueryString bilgilerini kontrol ederek tehlikeli bir ifade içermesi durumunda kullanıcıyı hata sayfasına yönlendirebiliriz. Aşağıdaki kod parçasında gelen her istekteki QueryString bilgisinin kontrol edilmesi ve tehlikeli ifade bulunması durumunda hata sayfasına yönlendirilmesi gerçekleştirilmiştir.

- Örnek-: Gelen talepleri inceleyerek farklı işlemler gerçekleştirmek (BeginRequest olayı)
- <%@ Application Language="C#" %>
- <script runat="server">
- protected void Application_BeginRequest(object sender, EventArgs e)
- { try
- {string q = this.Context.Request.QueryString.ToString();
- // Tehlikeli olabilecek kelimeler burada kontrol edilebilir

```
    if (q.Contains("delete") || q.Contains("update") ||
q.Contains("drop") || q.Contains("exec"))
```

```
Response.Redirect("hata.html");
```

```
• }
```

 catch {// Request nesnesi bazı isteklerde null değer getirebildiği için sorunla karşılaşmamak için ifadeleri try-catch bloğuna aldık, fakat hata durumunda bir işlem yapmayacağımız için de bu kısmı boş bıraktık }

```
} </script>
```

Server nesnesi sunucu üzerinde özellik ve metotlara ulaşmayı sağlar.

• Server.ScriptTimeout :

Normal bir sayfadaki scriptlerin belirli bir süre içinde çalışması beklenir.
 Bu süre aksi belirtilmedikçe 90 sn'dir. 90 sn içinde tamamlanmazsa sunucu scriptin çalışmasını durdurur.

<% Server.ScriptTimeout = 150; %>

Bu örnekte çalışma süresini 150 sn ile kendimiz belirliyoruz.

Server.Execute Metodu

- Bir başka dosya içindeki kodu sanki söz konusu kod mevcut dosyanın içindeymiş gibi çalıştırmanıza olanak tanır.
- <%@ Page Language="C#" CodeFile="Default2.aspx.cs" %>
- <html> <head runat="server"></head>
- <body>
- Genel olarak
 Server.Execute("Default3.aspx");
 kitaplarını
- </body> </html>
- <%@ Page Language="C#"%>
- <html> <head runat="server"></head>
- <body>
- <mark><%</mark> Response.Write(" Macera ");<mark>%></mark>
- </body>
- </html>



- Server. Transfer Metodu
- Bir tarayıcıyı bir başka konuma yönlendirme web tekniğinin yerini alır. Response.Redirect ile aynı özelliğe sahiptir.
- Ancak gönderildiği sayfaya sitil (örneğin ilk sayfadaki arka plan rengi gönderilen sayfadaki arka plan rengine baskın olur) ve içerikleri taşır.

- <%@ Page Language="C#" CodeFile="Default2.aspx.cs" %>
- <html> <head runat="server"></head><body>
- <form id="form2" runat="server" method="post">
- Genel olarak
- <% Server.Transfer("Default3.aspx");%> kitaplarını okurum.
- </form>
- </body> </html>
- <%@ Page Language="C#" CodeFile="Default3.aspx.cs" %>
- <html> <head runat="server"> </head>
- ody>
- <mark><%</mark> Response.Write("Macera");<mark>%></mark>
- </body> </html>

🚰 Untitled Page - Microsoft Inte	'n 💶 🗖 🔀
<u>D</u> osya Dü <u>z</u> en <u>G</u> örünüm <u>S</u> ık Kulla	lanlar 🎽 🥂
🕞 Geri 👻 🕑 👻 🚺 🚺	🏠 🔎 Ara 🏾 🎽
Adr <u>e</u> s 🙆 http://1(💙 🄁 Git 👘 Ba	antılar 🎽 📆 🗸
Genel olarak Macera	×
🙆 B 🛛 🔮 Interr	et

- Server.HtmlEncode Metodu
- Tarayıcıya HTML ve XML tagları yazılması istenen durumlar gerekebilir.
- Ancak HTML ya da XML kodunu yazmak için Response nesnesini kullanırsanız, tarayıcı onu HTML olarak yorumlar.
- Problemi çözmek için metni tarayıcıya göndermeden önce
 Server.HtmlEncode metodunu kullanarak düzgün olarak kullanın.

- <%@ Page Language="C#" %>
- <html > <head> </head> <body>
- <% Response.Write(Server.HtmlEncode("Server.HTMLEncode Metodunu Kullanarak tagı ekrana yazdırıldı."));
- </body> </html>

🕙 Untitled Page - Microsoft Internet Explorer 📃	
<u>D</u> osya Dü <u>z</u> en <u>G</u> örünüm <u>S</u> ik Kullanılanlar <u>A</u> raçlar <u>Y</u> ardım	1
🕞 Geri 🔹 🕥 👻 😰 🏠 🔎 Ara Kullanılanlar 🥝 🔗 🗸	»
Adres 🙋 http://10.6.2.77/WebSite/WebSite/proje2/Default3 💙 🎅 Git 🛛 Bağlantılar 🎽	🔁 🔹
Server.HTMLEncode Metodunu Kullanarak tagi ekrana yazdırıldı.	<
🙆 Bitti 🧼 Internet	

Server.MapPath Metodu

- Bir dosyanın tam konunu bulmak için Server.MapPath metodunu kullanırız.
- Aşağıdaki komut çalışmakta olan dizin yolunu döndürür.
- Server.MapPath(" ");
- Server.MapPath metodu ile kısayol "nokta" notasyonu kullanabilirsiniz. Tek bir nokta (./) mevcut dizini çift nokta (../) ise ebeveyn dizine göndermede bulunur.

- <%@ Page Language="C#" CodeFile="Default2.aspx.cs" %>
- <html> <head runat="server"></head>
- ody>
- <form id="form2" runat="server" method="post">
- div>
- Çalışılan sayfanın yolu
- Response.Write(Server.MapPath("Default2.aspx"));%>.
- </div>
- </form>
- </body>
- </html>

Untitled Page - Microsoft Internet Explorer	
<u>D</u> osya Düzen <u>G</u> örünüm <u>S</u> ık Kullanılanlar <u>A</u> raçlar <u>Y</u> ardım	1
🕞 Geri 🔹 🐑 👻 😰 🏠 🔎 Ara https://www.ski.kullanilanlar	**
Adres 🚳 http://10.6.2.77/WebSite/WebSite/pr 💙 ラ Git 🛛 Bağlantılar 🎽	🔁 🔹
Çalışılan sayfanın yolu c:\inetpub\wwwroot\WebSite\WebSite\proje2\Default2.aspx.	<
🙆 Bitti 🔮 Internet	

- Ödev :
- Verilmeyen Server Nesnesinin diğer özellik ve metotları öğrenilecek ve birer örnek üzerinde değerlendirilecek.

Dersleme Ders notları C# İLE ASP.NET Zafer Demİrkol Çeşitli İnternet kaynakları

TextReader ve Stream Nesnesi

- Dosya okuma için StreamReader yazma için ise StreamWriter nesnesi kullanılır.
- using System.IO;
- protected void Page_Load(object sender, EventArgs e)
- {
- TextReader tr = new StreamReader(Server.MapPath("veri.txt"));
- //tek satır okuma Response.Write(tr.ReadLine());
- Response.Write(tr.ReadToEnd());
- tr.Close();
- TextWriter tw = new StreamWriter(Server.MapPath("veri.txt"));
- tw.WriteLine(DateTime.Now);// Ornek olarak gunun tarihi yazılacak
- tw.WriteLine("Tarih değişti");
- tw.Close(); // stream kapat

• }

Dosya okunuyor

Untitled Page - Microsoft Internet Explorer	
Dosya Düzen <u>G</u> örünüm <u>S</u> ık Kullanılanlar <u>A</u> raçlar <u>Y</u> ardım	
🕝 Geri 👻 💽 😪 🛃 🖉 🖊 Ara 👷 Sik Kullanılanlar 🤗	* *
Adres 🕘 http://10.6.2.77/WebSite/WebSite/proje2/Defat 💙 🄁 Git 🛛 Bağlantılar	» 🔁 -
Bu günün tarihi değiştiriliyor Mevcut eski tarih: 13.02.2007 00:44:2	6
🝘 Bitti 🥥 Internet	.::

Sayfa yenilendikten sonra;

Untitled Page - Microsoft Internet Explorer	
<u>D</u> osya Dü <u>z</u> en <u>G</u> örünüm <u>S</u> ik Kullanılanlar <u>A</u> raçlar <u>Y</u> ardım	1
🕝 Geri 🔹 💿 🔹 🛃 🚮 🔎 Ara 🤺 Sik Kullanılanlar 🤡 👔	≥ *
Adres 💰 http://10.6.2.77/WebSite/WebSite/proje2/Defa 💌 🔁 Git 🛛 Bağlantılar 🎽	🔁 -
13.02.2008 14:12:59Tarih değişti	<
	~
😂 Bitti 🤍 🔮 Internet	

- Var olan dosyaya ekleme yapmak için ise aşağıdaki komut satırı kullanılır.
- TextWriter tw = new StreamWriter(Server.MapPath
 ("veri.txt") , true);

- Text Dosyasının Başından Belirli Sayıda Karekter Okumak
- protected void Page_Load(object sender, EventArgs e)
- TextReader tr = new StreamReader(Server.MapPath("veri.txt"));
- char[] strBuffer = new char[50];
- int d=tr.ReadBlock(strBuffer, 0,10);
- string f = new string(strBuffer);
- Response.Write(f); //ilk 10 karakteri ekrana yazar.
- Response.Write(d);// 10 yazar
- tr.Close();

• }

• {

- protected void Page_Load(object sender, EventArgs e)
- TextReader tr = new StreamReader(Server.MapPath("veri.txt"));
- char[] strBuffer = new char[50];
- int d = tr.ReadBlock(strBuffer, 0, 5);
 - /* string f = new string(strBuffer);
 Becone Write(f);
 ilk bec kanakte
 - Response.Write(f); ilk beş karakter*/
 - Response.Write(tr.ReadToEnd());

// veya Response.Write(tr.ReadLine()); //5. karakterden sonrasını okur

tr.Close();

• }

{
DOSYA OKUMA, YAZMA ve EKLEME

- Form İle Bir Text Dosyasına Yazdırmak
- ASP.NET sayfası üzerinden bir form sayesinde, ziyaretçilerin girdiği bilgiler alınıp, sunucu üzerinde bir text dosyasına kaydedilebilir.
- Bu text dosyasındaki bilgiler daha sonra kullanılmak üzere saklanabilir.

DOSYA OKUMA, YAZMA ve EKLEME

• Örnek:

protected void Button1_Click(object sender, EventArgs e)

```
• {
```

• TextWriter tw = new StreamWriter(Server.MapPath("veri.txt"), true);

```
tw.WriteLine(Request.Form["TextBox1"]);
```

```
tw.Close();
```

```
• }
```

protected void Button2_Click(object sender, EventArgs e)

```
• {
```

- TextReader tr = new StreamReader(Server.MapPath("veri.txt"));
- TextBox2.Text = tr.ReadToEnd();

```
tr.Close();
```

• }

DOSYA OKUMA, YAZMA ve EKLEME

🕘 Untitled	Page - Micros	soft Internet	Explore	r		×
<u>D</u> osya Dü <u>z</u>	en <u>G</u> örünüm	<u>S</u> ik Kullanilanla	r <u>A</u> raçlar	<u>Y</u> ardım		1
Geri 🝷	🔊 ·	2 🏠	🔎 Ara			»
Adr <u>e</u> s 🙆 htt	p://10.6.2.77/W	ebSite/We 🔽	🗲 Git	Bağlantılar	» 🔁	•
Önerilerini Bilgisay Elektron Bölümü	Z: yar nik	~	Gönderile	n		<
Alınan	Bilgisayar Elektronik Bölümü		~ >			<
ど Bitti			🥑 Ir	iternet		

- Directory Sınıfı
- Directory.CreateDirectory
- Belirtilen adreste bir klasör oluşturur.
- protected void Page_Load(object sender, EventArgs e)
- {
- Directory.CreateDirectory(Server.MapPath("belgeler"));
- }

- Directory. Delete :
- Delete(string yol): yoldaki boş olan bir klasörü siler.
 - Directory.Delete (Server.MapPath("belgeler"));
- Delete(string yol,bool a) : İkinci parametre(a) true ise yoldaki klasörü içindeki dosya ve klasörleri tamamen siler.
 - Directory.Delete (Server.MapPath("belgeler"),true);

- Directory. Exists: bool
- Belirtilen yolda klasör bulunuyorsa true aksi halde false döndürür.
 - Response.Write(Directory.Exists(Server.MapPath("belgeler")));

• true

- Directory.GetCreationTime DateTime
- Belirtilen yoldaki klasörün oluşturulma tarihini DateTime nesnesi olarak döndürür.
 - Response.Write(Directory.GetCreationTime(Server.MapPath("belgeler")));
 - 14.02.2008 16:13:57

Directory.GetCurrentDirectory()

- Çalışan programın hangi klasörde olduğunu verir.
 - Response.Write(Directory.GetCurrentDirectory());
 - C:\WINDOWS\system32
- Directory.GetFiles- string []
- Belirtilen yoldaki klasör içindeki dosyaları bir string türünden bir dizi olarak döndürür.
 - string[] a = Directory.GetFiles(yol, "*.*");

• **Directory.GetDirectories-** string []

 Belirtilen yoldaki bütün klasörlerin ismini bir string türünden bir dizi döndürür. Çalıştığımız bilgisayarın C sürücüsü içerisindeki klasörleri adını listelemek istersek,

- protected void Page_Load(object sender, EventArgs e)
 { string[] a;
- a = Directory.GetDirectories(@"c:\");
- foreach (string k in a)
- { Response.Write(k+"
>"); }

Directory

• }

- sınıfına ait
- diğer özellikleri
- inceleyiniz.

Untitled Page - Microsoft Internet Explorer		
<u>D</u> osya Dü <u>z</u> en <u>G</u> örünüm <u>Sık Kullanılanlar A</u> raçlar <u>Y</u> ardım		
🕝 Geri 👻 🐑 😪 😭 🔎 Ara 🬟 Sik Kullanılanlar	🚱 🔕 🙆	»
Adres 🕘 http://10.6.2.77/WebSite/WebSite/proje2/Default2.aspx	💌 🔁 Git 🛛 Bağlantılar 🎽	÷ 🔁
c:\Inetpub c:\jdk1.3.1_02 c:\KAV c:\MATLAB701		^
c:\MSOCache c:\odevler c:\ozel_bolum		
c\Program Files c\Python25 c\python_deneme c\RECYCLER		
c:\System Volume Information c:\TC c:\TCWIN c:\VIP c:\WINDOWS		
	Totornat	~

- DirectoryInfo Sınıfı
- DirectoryInfo sınıfı tek bir dizin ile ilgili bilgileri elde etmek için kullanılır.
 Statik olmayan metotları içerir.

- protected void Page_Load(object sender, EventArgs e)
- { string path = @"C:\Inetpub\wwwroot\WebSite\proje\k";
- DirectoryInfo k = new DirectoryInfo(path);
- Response.Write("Özellikler : ");
- Response.Write(k.Attributes + "
');
- Response.Write("Oluşturulma tarihi : ");
- Response.Write(k.CreationTime + "
');
- Response.Write("Klasör varm1 : ");
- Response.Write(k.Exists + "
);
- Response.Write("Uzant1 : ");
- Response.Write(k.Extension + "
');
- Response.Write("Tam Yol : ");
- Response.Write(k.FullName + "
');
- Response.Write("Son erişim zamanı : ");
- Response.Write(k.LastAccessTime + "
');
- Response.Write("Son yazma zamanı : ");
- Response.Write(k.LastWriteTime + "
');
- Response.Write("Klasörün adı : ");
- Response.Write(k.Name + "
>");
- Response.Write("Bir üst klasör : ");
- Response.Write(k.Parent + "
');
- Response.Write("Kök Dizin : ");
- Response.Write(k.Root);
- }

🗿 Untitled Page - Microsoft Internet Explorer 📃	
<u>D</u> osya Düzen <u>G</u> örünüm <u>S</u> ık Kullanılanlar <u>A</u> raçlar <u>Y</u> ardım	1
🕝 Geri 👻 💿 👻 😰 🏠 🔎 Ara tk Kullanılanlar 🍕	3 ×
Adres 🕘 http://10.6.2.77/WebSite/WebSite/proje 💙 🄁 Git 🛛 Bağlantılar 🎽	🔁 🔹
Özellikler : Directory Oluşturulma tarihi : 14.02.2008 18:46:03 Klasör varmı : True Uzantı : Tam Yol : C.\Inetpub\wwwroot\WebSite\WebSite\proje2\k\ Son erişim zamanı : 14.02.2008 19:37:45 Son yazma zamanı : 14.02.2008 18:46:03 Klasörün adı : k Bir üst klasör : proje2 Kök Dizin : C.\	<
🗃 Bitti 🥥 Internet	

• File Sınıfı

- File sınıfının bazı metotları, klasörler yerine dosyalar üzerinde işlem yapması hariç Directory sınıfının metotları ile aynıdır.
- Bu metotlar Exist(), Delete(), GetCreationTime(), GetLastAccessTime(), GetLastWriteTime(), GetLastWriteTime(), SetCreationTime(), SetLastAccessTime() ve Move()' dır.
- Ayrıca File sınıfı dosyalar üzerinde özelleşmiş işlemler yapabilmek için aşağıdaki metotları bulundurur.

- File.AppendText(string yol) : yol ile belirtilen dosya için StreamWriter nesnesi döndürür.
- protected void Page_Load(object sender, EventArgs e)
- {string path = @"veri.txt";
- if (!File.Exists(path))
- { using (StreamWriter sw = File.CreateText(path))

```
{ sw.WriteLine("Merhaba ");sw.WriteLine("ve ");sw.WriteLine("Hosgeldin");
```

```
• }
```

}

}

```
using (StreamWriter sw = File.AppendText(path))
```

```
• { sw.WriteLine("Bu"); sw.WriteLine("İlave"); sw.WriteLine("Metin"); }
```

```
using (StreamReader sr = File.OpenText(path))
```

```
• { while ((string s = sr.ReadLine()) != null)
```

```
{ Response.Write(s + "<br>");
```

🕙 Untitled Page - Micros 🔳 🗖 🚺					
Dosya Düzen Görünüm Sık.K. »	ľ				
🕞 Geri 👻 🕑 👻 📓 🏠	»				
Adr <u>e</u> s 🔽 🄁 Git 🛛 Bağlantılar 🎽 📆	•				
Merhaba ve Hoşgeldin Bu İlave Metin					
🥑 Internet					

FileInfo Sınıfı

- Tek bir dosya ile ilgili özellikleri içerir. FileInfo sınıfının özellikleri aşağıdaki programda toplu olarak verilmiştir.
- Statik olmayan metotları içerir.

- protected void Page_Load(object sender, EventArgs e)
- { string path = @"C:\Inetpub\wwwroot\WebSite\proje\veri.txt";
- FileInfo k = new FileInfo(path);
- Response.Write("Özellikler veya Dosya tipi : ");
- Response.Write(k.Attributes + "
');
- Response.Write("Oluşturulma tarihi : ");
- Response.Write(k.CreationTime + "
');
- Response.Write("Dosya varm1 : ");
- Response.Write(k.Exists + "
");
- Response.Write("Uzant1 : ");
- Response.Write(k.Extension + "
');
- Response.Write("Dosya Tam ad1 :");
- Response.Write(k.FullName + "
');
- Response.Write("Son erişim zamanı : ");
- Response.Write(k.LastAccessTime + "
');
- Response.Write("Son yazma zamanı : ");
- Response.Write(k.LastWriteTime + "
');
- Response.Write("Dosya ad1 : ");
- Response.Write(k.Name + "
");
- Response.Write("Boyut :");
- Response.Write(k.Length + "
");
- Response.Write("Bulunduğu klasör: ");
- Response.Write(k.DirectoryName + "
>");
- Response.Write("Salt Okunur Mu :" + k.IsReadOnly + "
);



Path sınıfı

 Path sınıfı string türden aldığı bir yol bilgisi üzerinde çeşitli işlemler yapan static üye metotlara sahiptir. Aşağıdaki örnekte bu metotların kullanımı gösterilmiştir.

- protected void Page_Load(object sender, EventArgs e)
- {
- string yol = @"C:\Inetpub\wwwroot\WebSite\proje\veri.txt";
- string yol2 = @"C:\Inetpub\wwwroot\WebSite\proje2\belgeler\veri.txt";
- string yeniyol=Path.ChangeExtension(yol,"jpg");
- Response.Write("Eski uzant1 : ");
- Response.Write(Path.GetExtension(yol) + "
");
- Response.Write("Yeni uzant1 : ");
- Response.Write("" + Path.GetExtension(yeniyol) + "
");
- Response.Write("Yeni Yol : ");
- Response.Write("" + Path.Combine(yol, yol2) + "
");
- Response.Write("Klasör : ");
- Response.Write("" + Path.GetDirectoryName(yol) + "
");
- Response.Write("Dosya Ad1 : ");
- Response.Write("" + Path.GetFileName(yol) + "
");
- Response.Write("Uzantısız Dosya Adı : ");
- Response.Write("" + Path.GetFileNameWithoutExtension(yol)+"
");

- Response.Write("Tam Yol : ");
- Response.Write("" + Path.GetFullPath(yol) + "
");
- Response.Write("Kök Dizin : ");
- Response.Write("" + Path.GetPathRoot(yol) + "
");
- Response.Write("Geçici Dosya Adı : ");
- Response.Write("" + Path.GetTempFileName() + "
");
- Response.Write("Geçici Dosya Dizini : ");
- Response.Write("" + Path.GetTempPath() + "
");
- Response.Write("Dosya Uzantısı Var mı? : ");
- Response.Write("" + Path.HasExtension(yol) + "
");
- Response.Write("Alt Dizin Ayıracı : ");
- Response.Write("" + Path.AltDirectorySeparatorChar + "
");
- Response.Write("Dizin Ayıracı : ");
- Response.Write("" + Path.DirectorySeparatorChar + "
");
- Response.Write("Geçersiz Yol Karakteri: ");
- Response.Write("" + Path.InvalidPathChars + "
");
- Response.Write("Yol Ayırıcı Karakter :");
- Response.Write("" + Path.PathSeparator + "
");
- Response.Write("Kök Dizin Ayıracı :");
- Response.Write("" + Path.VolumeSeparatorChar + "
");
- l

🕘 Untitled Page - Microsoft Internet Explorer 🛛 🔲 🔀				
Dosya Düzen Görünüm Sik Kullanılanlar Araçlar Yardım 🥂				
🕝 Geri 🔹 📀 - 💌 🗟 🏠 🔎 Ara 🧙 Sik Kullanılanlar 🚱 🔗 - چ 💿 - 🗔 🛍 🚺 🎽				
Adres 🕘 http://10.6.2.77/WebSite/WebSite/proje2/Default2.aspx 🛛 💽 Git 🛛 Bağlantılar 🌺 🐑				
Eski uzanti : .txt				
Y em uzanti : .jpg				
i em i ol : C:\Inetpub\wwwroot\WebSite\WebSite\proje2\belgeler\veri.txt				
Klasor : C:\Inetpub\wwwroot\WebSite\WebSite\proje2				
Dosya Adi : ven.txt				
Uzantisiz Dosya Adi : veri				
Tam Yol : C:\Inetpub\wwwroot\WebSite\WebSite\proje2\veri.txt				
Kök Dizin : C:\				
Geçici Dosya Adı : C:\Documents and Settings\etanyildizi\ASPNET\LOCALS~1				
\Temp\tmp5BF.tmp				
Geçici Dosya Dizini : C:\Documents and Settings\etanyildizi\ASPNET\LOCALS~1\Temp\ 🔤				
Dosya Uzantisi Var mi? : True				
Alt Dizin Ayıracı : /				
Dizin Ayıracı : \				
Geçersiz Yol Karakteri: System.Char[]				
Yol Ayırıcı Karakter :;				
Kök Dizin Ayıracı ::				
🕘 Bitti 🥥 Internet				

Dosya Upload (Server Dosya Gönderme işlemi)

- ASP.Net uygulaması oluştur
- 1-New Project> AsP.NET Web Application
- 2-http://localhost/CSharpUpload

<u>T</u> emplates:			.NET Framework 3.5 💉 🖻
Visual Stud	io installed templat	es	
asp.net 🕻	Web Site	🎎 ASP.NET Web Service	🎭 Empty Web Site
🞇 WCF Serv	rice	ASP.NET Reports Web Site	ASP.NET Crystal Reports Web Site 🎗
My Templa	tes		
Search Or	nline Templates		
_			
_			
_			
_			
-			
A blank ASP.NE	T Web site (.NET Fram	iework 3.5)	
A blank ASP.NE	T Web site (.NET Fram	ework 3.5)	Browse
A blank ASP.NE ocation: anguage:	T Web site (.NET Fram HTTP Visual C#	ework 3.5)	▼ Browse
A blank ASP.NE ocation: .anguage:	T Web site (.NET Fram HTTP Visual C#	ework 3.5) http://location/CSharpUpload	 ■ Browse

- Dosya Upload (Server Dosya Gönderme işlemi)
- Veri Dizini Oluştur
- 1- Solution Explorer penceresinde, CSharpUpload'ı sağ-tıkla, Add seçeneğini seç, New Folder seçeneğini aktif et ve yeni bir dizin oluştur.Varsayılan olarak NewFolder1 olarak dizin oluşturulur.
- 2- Dizin adını Data olarak değiştir. (NewFolder1 sağ tıkla, Rename seçeneğini tıkla ve Data olarak değiştir.



- Dosya Upload (Server Dosya Gönderme işlemi)
- 3-Windows gezgininden wwwroot içine bakıldığında
- C:\Inetpub\wwwroot\CSharpUpload\Data dizinin oluştuğu görülür.
- 4- Data dizinine sağ tıklayarak güvenlik özelliklerini Özellikler sekmesinden değiştirin.
- 5- Güvenlik sekmesine gelip Ekle seçeneğini tıklayınız

🗁 CSha	rpUplo	ad			
<u>D</u> osya	Dü <u>z</u> en	<u>G</u> örünüm	<u>S</u> ik Kullanılanlar	, »	2
Ger Ger	i • 🧲) - 😥	🔎 Ara		»
Adr <u>e</u> s 📔) C:\Inet	pub\www.roo	ot\CSharpUpload	~ $ i$	Git
Ø	App_Da	ta			
\bigcirc	Data				
	Default ASP.NE 1 KB	. aspx T Server Pag	je		
c #	Default Visual C 1 KB	. aspx.cs # Source file	9		
	web.coi XML Coi 9 KB	n fig nfiguration F	ile		
	CSha Dosya Cosya Adres Cosya Adres Cosya Cos	CSharpUplo Dosya Düzen Cosya Civen Adres Correction Adres Correction Adres Correction App_Data Default Aspr.NE Civen Default Visual Correction SMB	CSharpUpload Dosya Düzen Görünüm Comparent C:\Inetpub\www.rod Adres C:\Inetpub\www.rod Adres C:\Inetpub\www.rod Adres C:\Inetpub\www.rod App_Data Data Default.aspx ASP.NET Server Pag 1 KB Default.aspx.cs Visual C# Source file 1 KB Web.config XML Configuration F 9 KB	CSharpUpload Dosya Düzen Görünüm Sik Kullanılanları Geri C:\Inetpub\www.root\CSharpUpload Adres C:\Inetpub\www.root\CSharpUpload Adres C:\Inetpub\www.root\CSharpUpload Dof App_Data Default.aspx ASP.NET Server Page 1 KB Default.aspx.cs Visual C# Source file 1 KB Web.config XML Configuration File 9 KB	CSharpUpload Image: CSharpUpload Dosya Düzen Görünüm Sik Kullanılanlar * Geri Image: CSharpUpload

Data Özellikle	ri 🛛 🔁 🔀	
Genel Paylaş	ım Güvenlik Web Paylaşımı Özelleştir	
Grup ya da ku	Illanıcı adları:	
🔣 🚮 Administ	rators (ETANYILDIZI\Administrators)	
🕵 ASP.NE	T Machine Account (ETANYILDIZI\ASPNET)	
🕵 CREAT	Kullanada Dilainanalarana Caralarania	
🙎 Erkan T	Kullanıcılar, Bilgisayarlar veya Gruplar seçin	
🕵 Internet	Bu nesnenin türünü seçin:	
<u> </u>	Kullanıcılar, Gruplar ya da Yerleşik güvenlik ilkeleri	Nesne Türleri
	Bu konumdan:	
Administrators	firat.edu.tr	Konumlar
Tam Dene	Secilecek nesne adlarını girin (örnek):	
Değiştir		Adlan Danalla
Okuma ve		
Klasör İçer		
Oku		
Yaz Özəl İsinlər	Gelişmiş	amam İptal ;
Özel izin ya da tıklatın.	a gelişmiş ayarlar için Gelişmiş'i Gelişmiş	
	Tamam iptal Uygula	

- Dosya Upload (Server Dosya Gönderme işlemi)
- 6-Kullanıcılar ve Grupları seçin penceresinde Gelişmiş sekmesine girerek ASP.Net hesabını seçiniz. Seçimden sonra ekranda KullanıcıAdı\ASP.NET

oluşacak	Kullanıcılar veya Gruplar seçin		
5	<u>B</u> u nesnenin türünü seçin: Kullanıcılar, Gruplar ya da Yerleşik güvenlik ilkeleri Bu konumdan:	Nesne türleri:	
	ETANYILDIZI Ortak Sorgular	Konumlar	
	A <u>d</u> r. Başlar 💉	Siùtunlar Simdi B <u>u</u> l	
	Devre dışı hesa <u>p</u> lar Sür <u>e</u> siz parola	Kullanıcılar veya Gruplar seçin	? 🛛
	Son oturumdan beri geçen süre (gü <u>n)</u> :	By nesnenin türünü seçin: Kullanıcılar, Gruplar ya da Yerleşik güvenlik ilkeleri	Nesne Türleri
		Bu k <u>o</u> numdan: ETANYILDIZI	Konumlar
	Ad (HDN) Nasor Administrator ETANYILDIZI Administrators ETANYILDIZI ANDNYMOU	<u>S</u> eçilecek nesne adlarını girin (<u>örnek</u>):	
	ASPNET ETANYILDIZI Authenticated Backup Oper ETANYILDIZI		<u>A</u> dları Denetle
	BATCH CREATOR G CREATOR G DIALUP Ferronne	Geliş <u>m</u> iş Tamam	iptal

- Dosya Upload (Server Dosya Gönderme işlemi)
- 7- ASP.NET hesabı için tam denetim veriniz. Artık Data dizinine ekleme yapılabilecek

Data Özellikleri	?	×
Genel Paylaşım Güvenlik Web Paylaşımı Özelleştir		_
<u>G</u> rup ya da kullanici adlari:		
Administrators (ETANYILDIZI\Administrators)	^	
SP.NET Machine Account (ETANYILDIZIVASPNET)		
CREATOR OWNER	-	
🖸 Erkan TANYILDIZI (etanyildizi@firat.edu.tr)		
Internet Konuk Hesabi (ETANYILDIZI\IUSR_ETANYILDIZI)) 💌	
<u>Ekle</u>		
ASP.NET Machine Account için İzin Ver İzin Ver	me	
Tam Denetim	^	
Değiştir 🔽 📃		
Okuma ve Yürütme		
Klasör Içeriğini Listele 🗹 📃		
Daal jainlar	~	
Üzel izin ya da gelişmiş ayarlar için Gelişmiş'i Geliş <u>m</u> iş tıklatın.		,
Tamam iptal <u>U</u> y	gula	

- Default.aspx sayfasını değiştirme
- 1-Form tagı içerisine ilave edelim
- <form id="form3" runat="server" enctype="multipart/form-data">
- 2-Açılan Form tagından sonra aşağıdaki kodları ilave edelim
- <form id="form2" runat="server" enctype="multipart/form-data">
- div>
- Dosyayı Seçin:<asp:FileUpload ID="uplDosya" runat="server"/>

- <asp:Button ID="bntGonder" runat="server" Text="Gönder"
 OnClick="btnGonder_Click" />
- </div></form>

VS Ultimate 2013'ten sonra **enctype="multipart/form-data"** yazmaya gerek kalmadı, varsayılan durumda zaten aktif.

• Default.aspx sayfasını değiştirme

	<pre></pre>	
	asp:FileUpload#uplDosya	
Dosyayı Gönde	n Seçin : Browse	

- Default.aspx.cs dosyasına aşağıdaki kodları ekleyiniz.
- public partial class _Default : System.Web.UI.Page
- 1
 protected void btnGonder Click(object sender, EventArgs e)
- {
- if (uplDosya.HasFile)

Gönder

- {
- uplDosya.SaveAs(Server.MapPath("Data") + "\\" + uplDosya.FileName);
- }
- else Response.Write("Upload edilecek dosya yok");
- }

```
Dosyayı Seçin : ′ildizi\Desktop\bitirme.td Gözət...
```



- Upload işlemleri sırasında dikkat edilmesi gereken kritik bir sayı vardır.
 4096 byte. Yani 4 megabyte.
- Boyutu bu değerin üzerindeki bir dosyayı Upload etmek istediğimizde Asp.Net ortamı bir hata üretir ve dosyanın sunucuya gönderilmesine izin vermez.
- Eğer 4 megabyte üzerinde dosyaların upload edilebilmesini başka bir deyişle izin verilen limite kadar olan dosyaların gönderilebilmesini istiyorsak web.config dosyası içerisinde httpRuntime elementinin ayarlanması gerekmektedir.

- <?xml version="1.0"?>
- <configuration>
- <appSettings/>
- <connectionStrings/>
- <system.web>
- <httpRuntime maxRequestLength="51200"/>
- <compilation debug="true"/>
- <sessionstate cookieless="true" timeout="3" />
- <authentication mode="Windows"/>
- </system.web>
- </configuration>
- Burada yapılan ayarlamaya göre istemciler
- 50 megabyte' a kadar dosyaları sunucuya gönderebilecektir.



• Uygulama:

- Alınan dosyanın uzantısının docx veya pdf, dosya boyutunun ise 7 mb olduğunu kontrol eden bir arabirim oluşturunuz.
- Belirtilen klasörün boyutunu byte, kb, mb, veya gb cinsinden bulunuz.
- Diğer özellik ve komutları ders notlarından çalışarak deneyiniz.

VERİTABANINA BAĞLANMAK

VERİTABANI UYGULAMALARI

- Veri Merkezli Uygulamalar:
 - Veri Depolama,
 - Bağlantılı (Connected) ve Bağlantısız (Disconnected) Veri Ortamları,
 - Veri Erişim Yöntemleri, olarak ele alınmaktadır.

Veri Depolama

- Veriye erişmek için çeşitli veri depolama yöntemleri geliştirilmiştir.
 - Yapısal Olmayan Yöntem: Not defteri
 - Yapısal Yöntem: Excel
 - Hiyerarşik Yöntem: Xml
 - İlişkisel Veritabanı Yöntemi: Sql, Access, vs.
 - Nesne Yönelimli Veri Tabanı Yöntemi: Ado.Net

Veri Depolama

- Yapısal Olmayan: Bu yöntem ile depolanan veriler için belirli bir sınıflandırma ve sıralama yoktur. <u>Veriler düz bir şekilde kaydedilir</u>. Örneğin basit not dosyaları.
- Yapısal: Bu yöntem ile depolanan <u>veriler çeşitli gruplara ayrılarak saklanır</u> fakat bu gruplar arasında bir alt-üst ayrımı yapılmaz. Örneğin virgülle ayrılmış dosyalar (csv), Excel belgeleri.
- Hiyerarşik: Hiyerarşik depolama yöntemini ağaç yapısına benzetebiliriz. Bu yöntemde <u>veriler çeşitli kategorilere bölünerek depolanır</u>. Her bir kategorinin içerisinde alt kategorilerde olabilir. Örneğin XML (eXtensible Markup Language) dosyalar.
- İlişkisel Veritabanı: İlişkisel veritabanlarında veriler tablolar üzerinde depolanır. Tablo içerisindeki her bir satır kaydı, her bir sütun ise veriyi ifade eder. Örneğin SQL Server, Oracle, Access.
- Nesne Yönelimli Veritabanı: En gelişmiş veri depolama yöntemidir. Bu yöntemde veriler; ihtiyaca göre gruplandırılarak, nesneler içerisinde saklanır. Örneğin Versant, AOL.
- ADO.NET (ADO: Activex Data Object) bu depolama tekniklerinin tümünü destekler.

Bağlantılı ve Bağlantısız Veri Ortamları

- Bağlantılı (Connected) Veri Ortamları
- Bağlantılı veri ortamları, uygulamaların veri kaynağına sürekli bağlı kaldığı ortamlardır. Bu ortamlarda veri alma ve değiştirme işlemleri uygulama ile veri kaynağı arasında bağlantı kurulduktan sonra gerçekleştirilir. Bağlantılı veri ortamlarında, veri işlemleri gerçekleştiği sürece bağlantı açık kalır.

• Avantajları:

- En güvenli veri ortamı.
- Erişimler eş zamanlı.
- Dezavantajları:
 - Sabit bir ağ bağlantısının olması gerekir.
 - Ağ trafiğinin yoğunluğunu artırır.

Bağlantılı ve Bağlantısız Veri Ortamları

- Bağlantısız (Disconnected) Veri Ortamları
- Bağlantısız veri ortamı, uygulamanın veri kaynağına sürekli bağlı kalmadığı veri ortamıdır. Bağlantı, veri alış verişi yapılırken açılır, işlem bittikten sonra kapatılır.

• Avantajları:

- Taşınabilen aygıtlarla (Laptop, Pocket PC) girilen veriler, istenilen zamanda veri ortamlarına aktarılabilir.
- Uygulama performansını arttırır.

Dezavantajları:

- Verinin güncelliği sağlanmalıdır.
- Veri çakışması önlenmelidir.
Bağlantılı ve Bağlantısız Veri Ortamları



ADO.NET

- ADO (ActiveX Data Objects), farklı veri kaynaklarına hızlı ve güvenli erişim için Microsoft tarafından geliştirilen nesne modelidir.
- ADO.NET ise ADO teknolojisinin en yeni versiyonudur. ADO ile aynı programlama modelini kullanmamakla birlikte, ADO modelinden gelen pek çok çözüm yolunu da beraberinde getirir.
- ADO.NET nesne modeli iki ana bölümden oluşmaktadır.
 - INET Veri Sağlayıcı Sınıfları
 - DataSet Sınıfları

ADO.NET Nesne Modeli

• .NET Veri Sağlayıcı Sınıfları:

 <u>.NET veri sağlayıcı sınıfları, farklı türdeki veritabanlarına bağlanmak</u> <u>için kullanılır.</u> Bu sınıflar sayesinde istenilen türdeki veri kaynağına kolayca bağlantı kurulabilir, veri çekilebilir ve gerekli güncelleme işlemleri yapılabilir. ADO.NET nesne modeli, aşağıdaki veri sağlayıcı sınıflarını içerir: <u>SQL Server</u> .NET Veri Sağlayıcısı, <u>OLE DB</u> .NET Veri Sağlayıcısı, Diğer .NET Veri Sağlayıcıları

DataSet Sınıfları:

 <u>Çevrimdışı ortamlar için veri depolama ve yönetme işlemlerini sağlar.</u> DataSet sınıfları veri kaynağından bağımsız her tür uygulama ve veritabanı için kullanılabilir. Özellikle İlişkisel Veritabanı, XML ve XML Web Servisleri üzerinden veri çekmek için kullanılır.

ADO.NET Veri Sağlayıcıları

- NET veri sağlayıcıları, System.Data isim alanı içinde tanımlanmıştır.
 - SQL Server .NET: SQL Server 7.0 ve SQL Server 2000 veritabanlarına hızlı bağlantı sağlar. SQL Server bağlantı nesneleri System.Data.SqlClient isim alanında bulunur.
 - OLE DB .NET: SQL Server 6.5 ve daha öncesi sürümlerine, Oracle, Sybase, DB2/400 ve Microsoft Access veri tabanlarına bağlantı kurmayı sağlar. OLE DB bağlantı nesneleri System.Data.OleDb isim alanında bulunur.
 - ORACLE .NET: Oracle veritabanlarına bağlantı için tasarlanmış veri sağlayıcısıdır. Oracle bağlantı nesneleri System.Data.OracleClient isim alanında bulunur.
 - **ODBC** .NET: Diğer veritabanlarını destekleyen genel bir veri sağlayıcıdır. ODBC bağlantı nesneleri **System.Data.ODBC** isim alanında bulunur.

ADO.NET Veri Sağlayıcıları

- Her bir veri sağlayıcısı içerisinde, birçok bağlantı nesnesi bulunur.
 - Connection: Bağlantı kurmak için kullanılır.
 - Command : Veritabanına sorgu yollamak için kullanılır.
 - DataReader : Çevrim içi bağlantı ile sadece veri okuma.
 - DataAdapter : Çevrim dışı bağlantılarda veri işleme nesnesi.
- Veri Kaynaklarına Bağlanmak
 - Veri Sağlayıcı Seçmek
 - Bağlantı oluşturmak
 - Bağlantı Yönetimini seçmek

- System.Data.SqlClient isim alanı içerisinden <u>çevrimiçi</u> bağlantılar geliştirmek için SqlConnection, SqlCommand, SqlDataReader sınıfları kullanılır.
- SqlConnection; MS SQL Server üzerinde bağlantı açmak ve kapatmak için kullanılan sınıfdır.
- SqlCommand; MS SQL Server üzerinde Stored Procedure (Saklı Yordamlar) veya SQL Cümleleri çalıştırmak için kullanılan sınıftır.
- **SqlDataReader;** MS SQL Server üzerinde SqlCommand ile çalıştırılan SELECT sorguların sonuçlarını geri döndürmek için kullanılan sınıftır.

- System.Data.SqlClient isim alanı içerisinden <u>çevrimdışı</u> bağlantılar geliştirmek için SqlConnection, SqlDataAdapter, DataSet sınıfları kullanılır.
- •
- SqlConnection; MS SQL Server üzerinde bağlantı açmak ve kapatmak için kullanılan sınıftır.
- SqlDataAdapter; MS SQL Server'dan çekilen verileri DataSet içerisine ve DataSet'e çevrimdışı eklenmiş verileri MS SQL Server'a aktarmak için kullanılan sınıftır.
- DataSet; SQLDataAdapter nesnesinden gelen kayıtları çevrimdışı depolamak ve yönetmek için kullanılan sınıftır. DataSet tüm veri sağlayıcı sınıflar için ortaktır.

- System.Data.OleDb isim alanı içerisinden <u>çevrimiçi</u> bağlantılar geliştirmek için OleDbConnection, OleDbCommand, OleDbDataReader sınıfları kullanılır.
- •
- OleDbConnection; Access veya diğer veritabanları üzerinde bağlantı açmak ve kapatmak için kullanılan sınıftır.
- OleDbCommand; Access veya diğer veritabanları üzerinde Stored Procedure (Saklı Yordamlar) veya SQL Cümleleri çalıştırmak için kullanılan sınıftır.
- OleDbDataReader; Access veya diğer veritabanları üzerinde
 OleDbCommand ile çalıştırılan SELECT sorguların sonuçlarını geri
 döndürmek için kullanılan sınıftır.

- System.Data.OleDb isim alanı içerisinden <u>çevrimdışı</u> bağlantılar geliştirmek için OleDbConnection, OleDbDataAdapter, DataSet sınıfları kullanılır.
- OleDbConnection; Access veya diğer veritabanları üzerinde bağlantı açmak ve kapatmak için kullanılan sınıftır.
- OleDbDataAdapter; Access veya diğer veritabanlarından çekilen verileri DataSet içerisine ve DataSet'e çevrimdışı eklenmiş verileri ilgili veritabanına aktarmak için kullanılan sınıftır.

Bağlantı Oluşturmak

- Ms Access ile OLEDB Bağlantı Cümleleri
- Access'e Bağlantı:
- "Provider=Microsoft.Jet.OLEDB.4.0; Data Source=DB_Name.mdb;"
- Access'e Çalışma Grubu dosyası üzerinden Bağlantı:
- "Provider=Microsoft.Jet.OLEDB.4.0; Data Source=Db_Name.mdb; Jet OLEDB:System Database=Db_Name.mdw"
- Access'e Parola Korumalı Bağlantı: "Provider=Microsoft.Jet.OLEDB.4.0; Data Source=Db_Name.mdb; Jet OLEDB:Database Password=sifreniz"
- Network'teki Access'e Bağlantı: "Provider=Microsoft.Jet.OLEDB.4.0; Data Source=\\Server_Name\Share_Name\Share_Path\Db_Name.mdb"
- Remote Server(UzakServer) üzerindeki bir Access'e Bağlantı: "Provider=MS Remote; Remote Server=http://Your-Remote-Server-IP; Remote Provider=Microsoft.Jet.OLEDB.4.0; Data Source=Db_Name.mdb"

- Ms Access 2007 (ve sonrası) ile OLEDB Bağlantı Cümleleri
- Standard güvenlik
- Provider=Microsoft.ACE.OLEDB.12.0;Data Source=C:\myFolder\ myAccess2007file.accdb;Persist Security Info=False;
- Data Source=Uzak sunucunun domain adı;Initial Catalog=<u>database</u> <u>name</u>;User ID=<u>user name</u>;Password=<u>surround</u>;Persist Security Info=True;" />
 - Database password
- Provider=Microsoft.ACE.OLEDB.12.0;Data
 Source=C:\myFolder\myAccess2007file.accdb;Jet OLEDB:Database
 Password=MyDbPassword;

Bağlantı Oluşturmak

- SQL Server ile ODBC Bağlantı Cümleleri
- SQL Server sunucusuna SQL Authentication (Kimlik Doğrulama) ile bağlanmak: "Driver={SQL Server}; Server= Server_Name; Database=Db_Name; Uid=Username; Pwd= sifreniz;"
- SQL Server sunucusuna Windows Authentication ile bağlanmak:
- "Driver={SQL Server}; Server= Server_Name; Database=DB_Name; Trusted_Connection=yes;"

Bağlantı Oluşturmak

- SQL Server ile Sql Server Bağlantı Cümleleri
- SQL Server sunucusuna Authentication ile bağlanmak
- "Data Source=_Server_Name; Initial Catalog=Db _Name; User SQL Id= Username; Password=sifreniz;"
- SQL Server sunucusuna SQL Authentication ile bağlanmak
- "Server= Server_Name;Database=Db_Name;User ID= Username; Password=sifreniz;Trusted_Connection=False"
- SQL Server sunucusuna Windows Authentication ile bağlanmak
- "Data Source= Server_Name;Initial Catalog=Db_Name;Integrated Security=SSPI;"
- SQL Server sunucusuna SQL Authentication ile bağlanmak
- "Server=Server_Name;Database=Db_Name;Trusted_Connection=True;"

Bağlantı Yönetimi

- Bağlantı cümlesini oluşturduktan sonra, bağlantıyı açmak ve kapamak için Connection sınıfının iki önemli metodu kullanılır.
- Open ()
- Close ()
- (Bağlantı nesnesinin Dispose () metodu da bağlantıyı kapatmak için kullanılabilir.)

Bağlantı Durumlarını Kontrol Etmek

Bağlantı sınıfının durumu hakkında bilgi almak için, bağlantı sınıfının State özelliği kullanılır.

•	İsim	Açıklama	Değeri					
•	Broken	Yalnızca, açık bir bağlantının kopup tekrar bağlanıldığı durum	16					
•	Closed	Bağlantı kapalı	0					
•	Connectin	g Veri kaynağına bağlanma aşamasında	2					
•	Executing	Bağlantı üzerinden bir komutu çalıştırılıyor	4					
•	Fetching	Bağlantı üzerinden veri çekiliyor	8					
•	Open	Bağlantı açık	1					
	 private void BaglantiAc(System.Data.OleDb.OleDbConnection c 							
	• { /	//Connection, sadece kapalı ise açılacak						
	• i	<pre>if (con.State == ConnectionState.Closed)</pre>						
	•	<pre>con.Open();</pre>						
	• }							

 Win7 (64 bit) kullanıyorsanız mdb dosyalarında hata alabilirsiniz. Bunu gidermek için Solution Explorer da projenizde sağ tıklayıp Properties ten Build x86 yapınız.



- OleDb Üzerinden Access Veritabanına Erişim
- protected void Page_Load(object sender, EventArgs e)
- string connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data
 Source=" + Server.MapPath("fakulte.mdb");
- //bağlantı nesnesi kaynak ile oluşturulur.
- OleDbConnection conn = new OleDbConnection(connStr);
- conn.Open();
- Response.Write("Bağlantı Gerçekleştirildi");
- conn.Close();
- }

• veya

```
protected void Page_Load(object sender, EventArgs e)
```

```
• {
```

```
string connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=" + Server.MapPath("fakulte1.mdb");
```

- OleDbConnection conn = new OleDbConnection(connStr);
- try
- { conn.Open();

```
Response.Write("Bağlantı Gerçekleştirildi");
```

- }
- catch (Exception ex) { Response.Write(ex.Message); }
- conn.Close();



- OleDbCommand ve OleDbDataReader
- OleDbCommand nesnesi bir veri kaynağında ki tablodan SQL cümleciği kullanarak sorgulama yapmamızı sağlar (Seçme, Güncelleme, Ekleme ve Silme sorguları)
- OleDbDataReader sınıfı ise OleDbCommand nesnesinin çalıştırılması ile elde edilen sonuçları okumak için kullanılır. DataReader nesnesinde bulunan veriler sadece okunabilirdir, değiştirilemez. DataReader nesnesi herhangi bir konumda sadece bir kaydı tutar. Kayıtları elde etmek için DataReader'ın Read metodu kullanılır.

OleDbCommand Nesnesi

- string connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source="
 + Server.MapPath("fakulte.mdb");
- OleDbConnection conn = new OleDbConnection(connStr);
- conn.Open();
- Response.Write("Bağlantı Gerçekleştirildi");
- String sorgu = "Select Adı, Soyadı FROM Bilgiler";
- OleDbCommand komut = new OleDbCommand(sorgu, conn);

conn.Close();

- OleDbCommand Nesnesinin Çalıştırılması:
- ExecuteNonQuery() Metodu:
- Command Nesnesi üzerinde veri güncelleme, değiştirme ve silme işlemleri yapmak için kullanılır. Bu işlemin sonucunda etkilenen kayıt sayısı geriye döndürür.

- OleDbCommand Nesnesinin Çalıştırılması:
- ExecuteNonQuery() Metodu:



OleDbCommand Nesnesinin Çalıştırılması:

- <%@ Page Language="C#" CodeFile="Default.aspx.cs" %>
- <html ><head runat="server"><title>Basit Kayıt Ekleme</title></head>
- <body>
- Kayıt Ekleme için aşağıdaki formu doldurunuz...
- <form id="form2" runat="server">
- <asp:Label ID="Label2" style="..." runat="server" Text="Numaras1:" </asp:Label>
- <asp:Label ID="Label3" style="..." runat="server" Text="Ad1:" ></asp:Label>
- <asp:Label ID="Label4" style="..." runat="server" Text="Soyad1:" ></asp:Label>
- <asp:Label ID="Label5" style="..." runat="server" Text="Bölümü:"></asp:Label>
- <asp:Label ID="Label6" style="..." runat="server" Text="Dan1\$man1:"</asp:Label>
- <asp:label ID="Label7" style="..." runat="server" ></asp:label>
- <asp:TextBox ID="TextBox1" style="..." runat="server" ></asp:TextBox>
- <asp:TextBox ID="TextBox2" style="..." runat="server" ></asp:TextBox>
- <asp:TextBox ID="TextBox3" style="..." runat="server" ></asp:TextBox>
- <asp:TextBox ID="TextBox4" style="..." runat="server" ></asp:TextBox>
- <asp:TextBox ID="TextBox5" style="..." runat="server" ></asp:TextBox>
- <asp:Button ID="Button4" style="..." runat="server" Text="Kayıt" OnClick="Button1_Click" />
- <asp:Button ID="Button5" style="..." runat="server" Text="Formu Temizle"
 OnClick="Button2_Click" />
- </form> </body> </html>

• OleDbCommand Nesnesinin Çalıştırılması: Default.aspx.cs

- using System;
- using System.Collections.Generic;
- using System.Web;
- using System.Data;
- using System.Data.OleDb;
- using System.Web.UI;
- using System.Web.UI.WebControls;
- public partial class _Default : System.Web.UI.Page
- { protected void Page_Load(object sender, EventArgs e)
- { Label6.Text = "";
- protected void Button1_Click(object sender, EventArgs e)
- { string connStr = "Provider=Microsoft.Jet.OLEDB.4.0; Data
 Source=" + Server.MapPath("fakulte.mdb");
- OleDbConnection conn = new OleDbConnection(connStr);
- try
- { conn.Open(); }
- catch (Exception ex)
- { Response.Write(ex.Message); }

- OleDbCommand Nesnesinin Çalıştırılması:
- string sorgu = @"INSERT INTO Bilgiler(Numarası, Adı,Soyadı,Bölümü,Danışmanı) VALUES(' " + TextBox1.Text + " ',' " + TextBox2.Text + " ',' " + TextBox3.Text + " ',' " + TextBox4.Text + " ',' " + TextBox5.Text + " ')";
- OleDbCommand komut = new OleDbCommand(sorgu, conn);

```
komut.ExecuteNonQuery();
```

```
conn.Close(); Label6.Text = "Kayıt Eklendi«
```

```
• }
```

```
• protected void Button2_Click(object sender, EventArgs e)
• { TextBox1.Text = "" ; TextBox2.Text = "";
• TextBox3.Text = "" ; TextBox4.Text = "";
• TextBox5.Text = "";
• }
• }
```

- OleDbCommand Nesnesinin Çalıştırılması:
- ExecuteScalar() Metodu:
- Çalıştırılan Command nesnesinden geriye tek değer döndürmek için kullanılır. Bu değer nesne (Object) türündedir.
- Örneğin bir tablodaki kayıt sayısını elde etmek için kullanılan Count komutunun kullanıldığı SQL sorgularının çalıştırılması ExecuteScalar() metodu ile yapılabilir.
- Aynı şekilde tablodaki sayısal bir alanda bulunan değerlerin MAX ("alan") ve MIN("alan") özelliği ile maksimum ve minimum değerleri elde edilir.
- Kayıt eklenebilir.

- OleDbCommand Nesnesinin Çalıştırılması:
- protected void Button1_Click(object sender, EventArgs e)
- { string connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data
 Source=" + Server.MapPath("fakulte.mdb");
- OleDbConnection conn = new OleDbConnection(connStr);
- try { conn.Open(); }
- catch (Exception ex) { Response.Write(ex.Message); }
- string sorgu1 = "SELECT COUNT(*) FROM Bilgiler";
- OleDbCommand komut1 = new OleDbCommand(sorgu1, conn);
- int kayıt = (int)komut1.ExecuteScalar();

```
conn.Close();
```

```
Label1.Text = "Toplam Kayıt Sayısı:" + kayıt;
```

```
• }
```

```
Çıktı: Toplam Kayıt Sayısı : 10
```

- OleDbCommand Nesnesinin Çalıştırılması:
- protected void Button1_Click(object sender, EventArgs e)
- { string connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" + Server.MapPath("fakulte.mdb");
- OleDbConnection conn = new OleDbConnection(connStr);
- try { conn.Open(); }
- catch (Exception ex) { Response.Write(ex.Message); }
- string sorgu = @"INSERT INTO Bilgiler(Numarası,Adı,Soyadı,Bölümü,Danışmanı) VALUES(' " + TextBox1.Text + " ',' " + TextBox2.Text + " ','" + TextBox3.Text + " ','
 - " + TextBox4.Text + " ',' " + TextBox5.Text + " ')";
- OleDbCommand komut = new OleDbCommand(sorgu, conn);
- if (komut.ExecuteScalar()==System.DBNull.Value)

```
Label6.Text = "Kayıt Eklenemedi";
```

- else Label6.Text = "Kayıt Eklendi";
- conn.Close();

Çıktı: Kayıt Eklendi

• }

OleDbCommand Nesnesinin Çalıştırılması:

```
    protected void Button1_Click(object sender, EventArgs e)
```

- {
- string connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" +
 Server.MapPath("fakulte.mdb");
- OleDbConnection conn = new OleDbConnection(connStr);
- try { conn.Open(); }
- catch (Exception ex) { Response.Write(ex.Message); }
- string sorgu1 = "SELECT MAX(Notu) FROM Bilgiler";
- OleDbCommand komut1 = new OleDbCommand(sorgu1, conn);
- int notu = (int)komut1.ExecuteScalar();
- conn.Close();

```
Label1.Text = "En yüksek not:" + notu;
```

```
• }
```

• Çıktı: En yüksek not: 95

- OleDbCommand Nesnesinin Çalıştırılması:
- protected void Button1_Click(object sender, EventArgs e)

```
• {
```

- string connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" +
 Server.MapPath("fakulte.mdb");
- OleDbConnection conn = new OleDbConnection(connStr);
- try { conn.Open(); }
- catch (Exception ex) { Response.Write(ex.Message); }
- string sorgu1 = "SELECT MIN(Notu) FROM Bilgiler";
- OleDbCommand komut1 = new OleDbCommand(sorgu1, conn);
- int notu = (int)komut1.ExecuteScalar();
- conn.Close();

```
Label1.Text = "En düşük not:" + notu;
```

```
• }
```

```
Çıktı: En düşük not: 25
```

- OleDbCommand Nesnesinin Çalıştırılması:
- ExecuteReader() Metodu:
- Çalıştırılan Command nesnesinden geriye kayıt kümesi döndürmek için kullanılır. DataReader ile tanımlanan değişkene bu kayıt kümesi atanır.

Örnek: Veri tabanından veri elde etmek,

```
protected void Page Load(object sender, EventArgs e)
{
    string connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source="
  + Server.MapPath("fakulte.mdb");
          OleDbConnection conn = new OleDbConnection(connStr);
          try
              conn.Open();
          }
          catch (Exception ex)
          {
              Response.Write(ex.Message);
          }
```

- OleDbCommand Nesnesinin Çalıştırılması:
- string sorgu = "SELECT * FROM Bilgiler";
- OleDbCommand komut = new OleDbCommand(sorgu, conn);
- OleDbDataReader ord = komut.ExecuteReader();
- string [] a={"Numarası","Adı","Soyadı","Bölümü","Danışmanı"};
- TableRow r1 = new TableRow();
- **for(int** j=0;j<5;j++)
- { TableCell c = new TableCell();
- c.Controls.Add(new LiteralControl(a[j]));
- r1.Cells.Add(c);
- }
- Table1.Rows.Add(r1);

OleDbCommand Nesnesinin Çalıştırılması:

- while (ord.Read())
- { TableRow r = new TableRow();
- for (int i = 0; i <5; i++)</pre>
- { TableCell c = new TableCell();
- c.Controls.Add(new LiteralControl(ord[i].ToString()));

```
r.Cells.Add(c);
```

```
• }
```

```
Table1.Rows.Add(r);
```

```
• }
```

ord.Close(); conn.Close();

- }
- }

OleDbCommand Nesnesinin Çalıştırılması:

🖄 Untitled Page - Microsoft Internet Explorer										
<u>D</u> osya Dü <u>z</u> en <u>G</u> ö	rünüm <u>S</u> ik Kullan	ilanlar <u>A</u> raçlar	<u>Y</u> ardım			A*				
Geri 🔹 🕑	- 💌 🖻 🦿	🏠 🔎 Ara	Sik Kullanılar	nlar 🧭 🔗 🕇	🎍 o 🛛 🗖 🎇	🗾 🔇 🚳 👘				
Adres 🙆 http://10.6	🖌 🄁 Git	Bağlantılar 🎽 📆 🗸								
	Numarası	Adı	Soyadı	Bölümü	Danışmanı					
	02531042	İBRAHİM	öz	BİLGİSAYAR	ERKAN TANYILDIZI					
	02531029	ŞABAN	KAYABEY	BİLGİSAYAR	ERKAN TANYILDIZI					
	02500035	MEHMET	KÖKMEN	ELEKTRONİK	ENGİN AVCI					
	02500047	АНМЕТ	CAN	ELEKTRONİK	FERHAT BAĞÇACI					
	02531010	MEHMET	YILDIRIM	BİLGİSAYAR	MURAT KARABATAK					
	07531501	Ahmet	Deniz	Bilgisayar	Cafer BAL					
	07531001	Rabia	SANAÇ	Bilgisayar	Erkan TANYILDIZI					
E Bitti										

- Örnek: Veritabanından veri silmek,
- protected void Button1_Click(object sender, EventArgs e)
- {
- string connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source="
 - + Server.MapPath("fakulte.mdb");
- OleDbConnection conn = new OleDbConnection(connStr);
- try { conn.Open(); }
- catch (Exception ex) { Response.Write(ex.Message); }
- string a = TextBox1.Text;
- string sorgu1 = @"delete from Bilgiler WHERE Numarası='" + a +
 "'";
- OleDbCommand db_komut = new OleDbCommand(sorgu1, conn);
- int sayi = db_komut.ExecuteNonQuery();
- Response.Write("kayıtlar silindi.." + sayi.ToString());
- conn.Close();
- }
- Örnek: Veritabanında güncelleme yapmak
- protected void Button2_Click(object sender, EventArgs e)
- {
- string connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source="
 - + Server.MapPath("fakulte.mdb");
- OleDbConnection conn = new OleDbConnection(connStr);
- try { conn.Open(); }
- catch (Exception ex) { Response.Write(ex.Message); }
- string a = TextBox1.Text; string b = TextBox2.Text;
- string sorgu2 = @"UPDATE Bilgiler SET Numarası='" + b + "' WHERE
 Numarası='" + a + "'";
- OleDbCommand db_update_komut = new OleDbCommand(sorgu2, conn);
- int sayi = db_update_komut.ExecuteNonQuery();
- Response.Write("kayıt güncellendi.." + sayi.ToString());
- conn.Close();

• }

Bağlantısız (Disconnected) Veritabanı İşlemleri

- Bağlantısız veri ortamları, uygulamaların veri tabanından bağımsız çalıştığı ortamlardır.
- Veritabanı sunucusunun uzak olması, veri işlemlerinin uzun sürmesi ve mobil çalışma ihtiyacı, bağlantısız veri ortamlarına olan ihtiyacı artırmıştır.
 - DataAdapter nesnesi, DataSet nesne modeli, DataTable nesne modeli
- DataAdapter nesnesi: Connection, Command ve DataReader sınıflarını kullanarak, verilerin DataSet'e doldurulmasını ve DataSet de yapılan değişikliklerin veri tabanına kaydedilmesini sağlar.
- Veri kaynağından DataAdapter ile çekilen verilerin, çekirdek belleğe atılan kopyası DataSet içerisinde saklanır. DataSet ile bu veriler üzerinde gerekli düzenlemeler yapıldıktan sonra, veriler aynı DataAdapter ile veritabanına aktarılır.

Bağlantısız (Disconnected) Veritabanı İşlemleri

DataSet Nesne Modeli

 DataSet, Sanal bir veritabanı yapısını temsil eder. DataTable nesnelerinden oluşur. Bu tablolar arasında ilişkiler tanımlanabilir. DataSet'i oluşturan nesneler: DataTable, DataColumn, DataRow, DataRelation nesneleridir.



DataSet Nesne Modeli

- DataSet, DataAdapter, DataTable, DataRow, DataColumn
- DataAdapter , DataSet nesnelerine veri kaynağından bilgi alıp doldurmak için kullanılır. Verilerin getirilmesi ve iletilmesi görevlerini otomatik olarak yapar. Bunun için update, select, insert, ve delete komutlarını kullanır.
- DataSet, veri kaynağından bağımsız olarak veriler üzerinde işlem yapmamızı ve gerektiğinde DataAdapter yardımıyla DataSet üzerindeki verileri tekrar veri kaynağına aktarmamızı sağlar. Bir DataSet ile veri kaynağından bağlantı kesilse bile DataSet içerisindeki verilerle çalışabiliriz.
- DataTable sınıfı, bir veri kaynağındaki tablolara benzer. Yani bir DataTable nesnesinde sütunlar ve verileri içeren kayıt satırları bulunmaktadır. Bir DataSet içinde bir veya daha fazla DataTable bulunabilir. Ado.Net 2.0 ve sonraki sürümlerde DataSet'e ihtiyaç duymadan veriler DataTable atanabilir.
- DataRow sınıfı, DataTable nesneleri için veri satırlarını temsil eder.
- DataColumn sınıfı, bir tablodaki sütunları temsil eder. DataColumn nesneleri ile bir sütunun çeşitli özellikleri değiştirilebilir. Aşağıda bir DataColumn nesnesinin oluşturulması ve bazı özellikleri verilmiştir.

- using System.Data;
- using System.Data.OleDb;
- public partial class Data_set : System.Web.UI.Page
- { protected void Page_Load(object sender, EventArgs e)
- {
- string connStr = "Provider=Microsoft.Jet.OLEDB.4.0;Data Source="
 + Server.MapPath("fakulte.mdb");
- OleDbConnection baglan = new OleDbConnection(connStr);
- try { baglan.Open(); }
- catch (Exception ex) { Response.Write(ex.Message); }
- string sorgu = "SELECT * FROM Bilgiler";
- OleDbDataAdapter adp = new OleDbDataAdapter(sorgu, baglan);
- DataSet ds = new DataSet();
- adp.Fill(ds, "Bilgiler");

- •
- foreach (DataColumn dc in ds.Tables["Bilgiler"].Columns)
- Response.Write("<center>"+"Sütun Ad1 : " + dc.Caption + ", Veri Tipi : " + dc.DataType.ToString() +"</center>"+ "
");
- Response.Write("<center>-----Bilgiler Tablosu-----</center>" + "
";

```
    TableRow r1 = new TableRow();
```

- •
- foreach (DataColumn dc in ds.Tables["Bilgiler"].Columns) TableCell c1 = new TableCell();
- c1.Controls.Add(new LiteralControl(dc.Caption));
- r1.Cells.Add(c1); }
- Table1.Rows.Add(r1);

```
foreach (DataRow dr in ds.Tables["Bilgiler"].Rows)
      TableRow r = new TableRow();
   {
        object[] dizi = dr.ItemArray;
        foreach (object i in dizi)
                                              {
            TableCell c = new TableCell();
            c.Controls.Add(new LiteralControl(i.ToString()));
            r.Cells.Add(c);
                            }
       Table1.Rows.Add(r);
    }
   baglan.Close();
}
```

🚰 Untitled Page - Microsoft In	ternet Explorer									
Dosya Düzen <u>G</u> örünüm Sık Kull	lanılanlar <u>A</u> raçlar <u>Y</u> ardım									
🕞 Geri 👻 🕑 👻 📓	🏠 🔎 Ara tk Kullanı	lanlar 🧭 🍃	• 질 🧕 •	📃 🛍 🚺 🔇	3 🚳					
Adr <u>e</u> s 餐 http://10.6.2.77/WebSite/V	/tabani/Data_set.aspx					💌 🄁 Git	Bağlantılar »	1		
Sütun Adı : Numarası, Veri Tipi : System String										
		Sutu	n Adı. Adı, ven	. ripi . System.Sunig	5					
		Sütun	Adı : Soyadı, Ve	eri Tipi : System Stri	ng					
		Sütun .	Adı : Bölümü, V	eri Tipi : System.Stri	ng					
		Sütun A	.dı : Danışmanı, V	/eri Tipi : System.St	ring					
			Bilgiler	Tablosu						
	Numarası	Adı	Soyadı	Bölümü	Danışmanı					
	02531042	İBRAHİM	öz	BİLGİSAYAR	ERKAN TANYILDIZI					
	02531029	ŞABAN	KAYABEY	BİLGİSAYAR	ERKAN TANYILDIZI					
	02500035	MEHMET	KÖKMEN	ELEKTRONİK	ENGİN AVCI					
	02500047	AHMET	CAN	ELEKTRONİK	FERHAT BAĞÇACI					
	02531010	MEHMET	YILDIRIM	BİLGİSAYAR	MURAT KARABATAK					
	07531501	Ahmet	Deniz	Bilgisayar	Cafer BAL					
	07531001	Rabia	SANAÇ	Bilgisayar	Erkan TANYILDIZI					

- Veri Gösterim Kontrolleri
- GridView Kontrolü (DataSet, DataBind(), Select, Insert, Update, Delete)
- GridView kontrolünü veri güncelleme ve silme işlemleri için kullanacağız. Bu örneği geliştirebilmemiz için, GridView kontrolünü Access veritabanımıza bağlayacak bir DataSource kontrolüne ihtiyacımız var.



- Öncelikli olarak, AccessDataSource kontrolümüzü aşağıdaki aspx kodları ile sayfaya ekleyelim.
- <asp:AccessDataSource ID="AccessDataSource1" runat="server"
 DataFile="~/fakulte.mdb"</pre>
- SelectCommand="SELECT [Numarası], Adı, Soyadı, Bölümü, Danışmanı FROM Bilgiler"
- UpdateCommand="UPDATE Bilgiler Set Ad1=?, Soyad1=?,Bölümü=?, Dan1şman1=?
 WHERE (Numaras1=?) " >
- <UpdateParameters>
- <asp:Parameter Name="Ad1" Type="String" ></asp:Parameter>
- <asp:Parameter Name="Soyad1" Type="String" ></asp:Parameter>
- <asp:Parameter Name="Numarası" Type="Int32" ></asp:Parameter>
- <asp:Parameter Name="Bölümü" Type="String" ></asp:Parameter>
- <asp:Parameter Name="Danışmanı" Type="String" ></asp:Parameter>
- </UpdateParameters>
- </asp:AccessDataSource>

 SelectCommand ve UpdateCommand komutları dışında kullanılabilecek diğer komutlarda, DeleteCommand ve InsertCommand' dir. Bunların herbirinin parametre koleksiyonları farklıdır. Yani DeleteCommand için parametre koleksiyonu DeleteParameters iken, InsertCommand için InsertParameters' dır.



- <form id="form2" runat="server">
- <asp:GridView ID="GridView1" runat="server" AllowPaging="True" AllowSorting="True" AutoGenerateColumns="False" DataSourceID="AccessDataSource1" DataKeyNames="Numaras1" > <!-birincil anahtar -->
- Columns>
- <asp:BoundField DataField="Numarası" HeaderText="Numarası"
 SortExpression="Numarası" ReadOnly="True" />
- <asp:BoundField DataField="Ad1" HeaderText="Ad1" SortExpression="Ad1" />
- <asp:BoundField DataField="Soyad1" HeaderText="Soyad1"
 SortExpression="Soyad1" />
- <asp:BoundField DataField="Bölümü" HeaderText="Bölümü"
 SortExpression="Bölümü" />
- <asp:BoundField DataField="Danışmanı" HeaderText="Danışmanı"
 SortExpression="Danışmanı" />
- <asp:CommandField ShowEditButton="True" />
- <asp:CommandField ShowDeleteButton="True" />
- </Columns>
- </asp:GridView>

🕫 Vta	abani	- Micr	osoft Vis	ual Sti	udio					
Eile	<u>E</u> dit	⊻iew	Web <u>s</u> ite	<u>B</u> uild	<u>D</u> ebug	Format	La <u>v</u> out	<u>T</u> ools	Te <u>s</u> t	ļ
1	-	· 🔁 🛛	3 🥵 👌	6 🗈	2 9	- (21 -	F - E		Debug	
None	1	▼ Tim	es New Ror	man 👻	12pt	- B.	<u>z</u> u	A 🖉	≣ -	•
Data_grid.aspx.cs Data_grid.aspx Start Page										

oolboy

Numarası	Adı	Soyadı	Bölümü	Danışmanı					
abc	abc	abc	abc	abc	Edit	Delete			
abc	abc	abc	abc	abc	Edit	Delete			
abc	abc	abc	abc	abc	<u>Edit</u>	Delete			
abc	abc	abc	abc	abc	<u>Edit</u>	Delete			
abc	abc	abc	abc	abc	<u>Edit</u>	Delete			
abc	abc	abc	abc	abc	<u>Edit</u>	Delete			
abc	abc	abc	abc	abc	Edit	Delete			
abc	abc	abc	abc	abc	<u>Edit</u>	Delete			
abc	abc	abc	abc	abc	<u>Edit</u>	Delete			
abc	abc	abc	abc	abc	Edit	Delete			
12									
AccessData	P AccessDataSource - AccessDataSource1								

			elete plateField
Fields			
<u>A</u> vailable fields:	BoundFie	eld properties:	
(All Fields)	∎ Az		
	NullDi	isplayText	
	Read	Only	True
Soyadı	Show	Header	True
Bölümü	SortE	xpression	Numarası
🔄 🔄 Danışmanı 💽	Visible	в	True
	🗆 Data	1	
Add	Dataf	Field	Numarası
Selected fields:	Dataf	FormatString	
	🗆 Style	25	
I Adi	🗄 Contr	rolStyle	
Sovadı	E Foote	erStyle	
Bölümü Danışmanı	ReadOr Whether bound fi	nly r the field does ield.	not permit editing of its
Auto-generate fields	Convert	this field into a	TemplateField
Refresh Schema			OK Cancel

Fields

Available fields: 🐻 ImageField

📲 ButtonField CommandField

Select

🔏 Edit, Update, Cancel

.

E

Add

~

 Properties ekranında bulunan Auto Format ile GridView inize istediğiniz formatı verebilirsiniz. Biz Autumn Formatını seçiyoruz.

abani -	Microsoft Visu	ial Studio	0			*	Vtabani ·	Microsoft	Visual St	udio			
Edit	View Website	Build D	ebug F	orma	at Layout Tools Test	Window Ei	le <u>E</u> dit	⊻iew Web	<u>s</u> ite <u>B</u> uild	<u>D</u> ebug	F <u>o</u> rmat La <u>v</u> o	ut <u>T</u> ools Te	<u>st W</u> in
• 📖 •		5 49 03	• • •	61	- La P Debug	.Nt	7 - 🔛 -		<u>ا</u> کر ا	B 9	• (* - 🖓 -	🖳 🕨 Debu	ıg
6	*		3	В				-			- B <i>I</i> <u>U</u>	A 🔏 🗏	- 1:=
Data_g	rid.aspx* Dat	a_grid.asp;	x.cs S	tart i	age		Data_	grid.aspx*	Data_grid	.aspx.cs	Start Page		
Ð						Toolb	(AL)						
Numa	rası <u>Adı Soya</u>	<u>dı Bölün</u>	<u>nü Dan</u>	ışm	am	<u>×</u>	Num	arası Adı	Sovadı	Bölümi	i Danısmanı		
ibc	abc abc	abc	abc	¥	Cut		aho	aha	aho	aho	aha	Düzenle	ເຊິ
0C	abc abc	abc	abc	C)	Сору		auc	auc	auc	auc	auc	Duzenie	
bc	abc abc	abc	abc	(H	Paste		abc	abc	abc	abc	abc	Düzenle) [Sil]
bc	abc abc	abc	abc	~	Paste Alternate		abc	abc	abc	abc	abc	Düzenle	Sil
oc	abc abc	abc	abc	^	Delete	-	-1	-1	-1	-1	-1	Düzenle	ເພ
)C	abc abc	abc	abc		Style	-	abc	abc	abc	abc	abc	Duzenie	
DC	abc abc	abc	abc		View Code View Component Designer View in Browser		abc	abc	abc	abc	abc	Düzenle) Sil
bc	abc abc	abc	abc	8			abc	abc	abc	abc	abc	Düzenle) [Sil]
abc	abc abc	abc	abc	_	Edit Image		_1_		-1	-1	-1 -	Düzenle	ເພ
abc	abc abc	abc	abc		Show Smart Tag		abc	abc	abc	abc	abc	Duzenie	
12			8		Auto Format		abc	abc	abc	abc	abc	Düzenle) [Sil]
Access	DataSource - Ad	cessDataS	o ource1		Edit Template		abc	abc	abc	abc	abc	Düzenle	Sil
					End Template Editing		-1	-1	-1	-t	-1	Düzenle	່
				\$	Refresh		abc	abc	abc	abc	aoc	Duzenie	J
				-	Properties					1 2	2		
							Acces	5DataSourc	e - AccessD	ataSource1			Ó

🕙 Untitled P	age - Microso	oft Internet Ex	cplorer							
<u>D</u> osya Dü <u>z</u> er	n <u>G</u> örünüm	<u>S</u> ik Kullanılanlar	<u>A</u> raçlar <u>Y</u> ardım			- 1				
🔇 Geri 🔹 🕥 - 💌 🛃 🏠 🔎 Ara 🌟 Sik Kullanilanlar 🚱 😥 - 嫨 🖸 - 🛄 鑬 🎽										
Adres 🕘 http://10.6.2.77/WebSite/Vtabani/Data_grid.aspx 🛛 🍷 Git 🛛 Bağlantılar 🌺 🏹 🗸										
<u>Numarası</u>	Adı	<u>Soyadı</u>	<u>Bölümü</u>	<u>Danışmanı</u>						
02531042	İBRAHİM	ÖZ	BİLGİSAYAR	ERKAN TANYILDIZI	Düzenle Sil					
02531029	ŞABAN	KAYABEY	BİLGİSAYAR	ERKAN TANYILDIZI	Düzenle Sil					
02500035	MEHMET	KÖKMEN	ELEKTRONİK	ENGİN AVCI	Düzenle Sil	=				
02500047	AHMET	CAN	ELEKTRONİK	FERHAT BAĞÇACI	Düzenle Sil					
02531010	MEHMET	YILDIRIM	BİLGİSAYAR	MURAT KARABATAK	Düzenle Sil					
08531501	Rabia	SANAÇ	Bilgisayar	Erkan TAN YILDIZI	Düzenle Sil					
ど Bitti					🥝 Internet	×				

🕘 Untitled P	🖻 Untitled Page - Microsoft Internet Explorer										
<u>D</u> osya Dü <u>z</u> en	Dosya Düzen Görünüm Sık kullanılanlar Araçlar Yardım 🧗										
🔇 Geri 👻 💿 🔹 👔 🏠 🔎 Ara 🧙 Sik Kullanilanilar 🤣 🎅 + 🌺 💿 🔹 🛄 🕵 🥸											
Adres 🙆 http:	//10.6.2.77/WebSite/Vtaba		💌 🔁 Git 🛛 E	Bağlantılar »	📆 •						
							~				
<u>Numarası</u>	Adı	<u>Soyadı</u>	<u>Bölümü</u>	<u>Danışmanı</u>							
02531042	İBRAHİM	ÖZ	BİLGİSAYAR	ERKAN TANYILDIZI	Düzenle	Sil					
02531029	ŞABAN	KAYABEY	BİLGİSAYAR	ERKAN TANYILDIZI	Düzenle	Sil					
02500035	MEHMET	KÖKMEN	ELEKTRONİK	ENGİN AVCI	Düzenle	Sil					
02500047	AHMET	CAN	ELEKTRONİK	FERHAT BAĞÇACI	Düzenle	Sil					
02531010	MEHMET	YILDIRIM	BİLGİSAYAR	MURAT KARABATAK	Düzenle	Sil					
08531501	Rabia	SANAÇ	Bilgisayar	Erkan TANYILDIZI	Güncelle	İptal					
i 🛃 Bitti					🔮 Int	ernet					

- Silme işlemi de güncelleme(Update) işleminin benzeridir.
- <asp:AccessDataSource ID="AccessDataSource1" runat="server"
 DataFile="~/fakulte.mdb"</pre>
- SelectCommand="SELECT [Numarası], Adı, Soyadı, Bölümü, Danışmanı FROM Bilgiler"
- UpdateCommand="UPDATE Bilgiler Set Ad1=?, Soyad1=?,Bölümü=?, Dan1şman1=?
 WHERE (Numaras1=?) " >
- DeleteCommand ="DELETE FROM Bilgiler WHERE ([Numarası]=?)" >
- <UpdateParameters>
 - <asp:Parameter Name="Numaras1" Type="Int32" />
 - <asp:Parameter Name="Ad1" Type="String" />
 - <asp:Parameter Name="Soyad1" Type="String" />
 - <asp:Parameter Name="Bölümü" Type="String" />
 - <asp:Parameter Name="Danışmanı" Type="String" />
- </UpdateParameters>
- <DeleteParameters>
 - <asp:Parameter Name="Numarası" /> </DeleteParameters>
- <InsertParameters>
 - <asp:Parameter Name="Numaras1" Type="Int32" /> <asp:Parameter Name="Ad1" Type="String" />
 - <asp:Parameter Name="Soyad1" Type="String" /> <asp:Parameter Name="Bölümü"
 Type="String" />
 - <asp:Parameter Name="Danışmanı" Type="String" />
- </InsertParameters>
- </asp:AccessDataSource>

Untitled Page 1	age - Microso	oft Internet Ex	cplorer								
<u>D</u> osya Dü <u>z</u> en	<u>G</u> örünüm	<u>S</u> ik Kullanılarlar	<u>A</u> raçlar <u>Y</u> ardım			A					
Geri 🝷	😋 Geri 🔻 💿 🔹 📓 🏠 🔎 Ara 🤺 Sik Kullanilanilar 🤣 🙆 + 嫨 🔟 + 🛄 🎇 🎽										
Adres 🕘 http://10.6.2.77/Web5ite/Vtabani/Data_grid.aspx 💽 🔂 Git 🛛 Bağlantılar 🎽 📆 🖛											
<u>Numarası</u>	<u>Adı</u>	<u>Soyadı</u>	<u>Bölümü</u>	<u>Danışmanı</u>							
02531042	İBRAHİM	ÖZ	BİLGİSAYAR	ERKAN TANYILDIZI	Düzenle	Sil					
02531029	ŞABAN	KAYABEY	BİLGİSAYAR	ERKAN TANYILDIZI	Düzenle	Sil					
02500035	MEHMET	KÖKMEN	ELEKTRONİK	ENGİN AVCI	Düzenle	Sil					
02500047	AHMET	CAN	ELEKTRONİK	FERHAT BAĞÇACI	Düzenle	Sil					
02531010	MEHMET	YILDIRIM	BİLGİSAYAR	MURAT KARABATAK	Düzenle	Sil					
08531501	Ramazan	SANAÇ	Bilgisayar	Erkan TAN YILDIZI	Düzenle	Sil					
					1.0	~					
🖲 Bitti					🥝 Internet	.:!					

Ę	Untitled P	age - Microso	oft Internet Ex	cplorer							
!	<u>D</u> osya Dü <u>z</u> en	<u>G</u> örünüm	<u>S</u> ik Kullanılarlar	<u>A</u> raçlar <u>Y</u> ardım							
1	😋 Geri 🔹 📀 🔹 🛃 🏠 🔎 Ara 🜟 Sik Kullanilanlar 🤣 🖾 - چ 🖸 - 🛄 🛍 🎽										
Adres 🕘 http://10.6.2.77/WebSite/Vtabani/Data_grid.aspx 💽 🈏 Git 🛛 Bağlantılar 🌺 📆 🖛											
							~				
	<u>Numarası</u>	<u>Adı</u>	<u>Soyadı</u>	<u>Bölümü</u>	<u>Danışmanı</u>						
	02531042	İBRAHİM	ÖZ	BİLGİSAYAR	ERKAN TANYILDIZI	Düzenle	Sil				
	02531029	ŞABAN	KAYABEY	BİLGİSAYAR	ERKAN TANYILDIZI	Düzenle	Sil				
	02500035	MEHMET	KÖKMEN	ELEKTRONİK	ENGİN AVCI	Düzenle	Sil				
	02500047	AHMET	CAN	ELEKTRONİK	FERHAT BAĞÇACI	Düzenle	Sil				
	02531010	MEHMET	YILDIRIM	BİLGİSAYAR	MURAT KARABATAK	Düzenle	Sil				
							~				
1	Bitti					🥝 Internet					

- Web.Config Dosyasından Veri Tabanına Bağlanma
- Asp.Net'te veri tabanına bağlanma genelde her sayfada ayrı ayrı yapılan bir işlemdir. Büyük ve geniş çaplı projeler, uygulamalar üzerinde çalışıldığında bağlantı da yapılan bir değişikliği her bağlantı oluşturduğumuz sayfada tek tek bütün sayfalarda yapılması gerekir.
- Bu durumu önlemek adına farklı yollar mevcuttur. Bu yollardan birisi de bağlantıyı web.config dosyasında tanımlayıp kullanmaktır.
 - Bağlantıyı oluşturmak için web.config dosyasında configuration etiketinin altına aşağıdaki gibi yol tanımlanabilir.

- Web.Config Dosyasından Veri Tabanına Bağlanma
- <?xml version="1.0"?>
- <configuration>
- connectionStrings>
- <add name="veritaban1" connectionString="Provider=Microsoft.Jet.oledb.4.0; Data Source=C:/Users/info-pc/Desktop/deneme.mdb" providerName="System.Data.Oledb"/>
 - </connectionStrings>
- </configuration>

Web.Config Dosyasından Veri Tabanına Bağlanma

- using System;
- using System.Data;
- using System.Data.OleDb;
- public partial class _Default : System.Web.UI.Page
- { protected void Page_Load(object sender, EventArgs e)
- { string baglanti =
 - WebConfigurationManager.ConnectionStrings["veritaban1"].ConnectionString;
- /*burda web.configdeki bağlantı değişken olarak alınıp kullanılıyor. */
- OleDbDataAdapter da = new OleDbDataAdapter("select * from Tablo1", baglanti);
- DataSet ds = new DataSet();
- da.Fill(ds, "Tablo1");
- GridView1.DataSource = ds;
- GridView1.DataBind(); //Methodu ile veriyi datagrid'imizle birleştirmiş oluyoruz
- •

}

- Veri kaynakları sadece Access ve ya SQL ile sınırlı değildir. Veri kaynaklarından bir tanesi de XML dir. Günümüzde farklı platformdaki sistemler birbirleri ile haberleşirken XML dilini kullanırlar.
- XML bir programlama dili değildir. XML belgelerini, biçimlendirilmiş metin tabanlı verileri, depolama işine yarayan belgeler şeklinde nitelendirebiliriz.
- XML ile metin tabanlı veri saklama sistemine büyük kolaylık gelmiştir. Normal veri tabanlarındaki gibi verilere çeşitli isimler verilir ve gerektiğinde bu verinin ismi kullanılarak asıl veri dosyadan çekilir.
- XML'in standart olmasından dolayı XML tabanlı birçok uygulama alanı gelişmiştir. Örneğin farklı sistemler arasında haberleşmeyi sağlayan Web servisleri bu uygulama alanlarından en önemlisidir. Web servisleri sayesinde internet protokolü dediğimiz HTTP üzerinden, XML yardımıyla bir sunucudaki metodu çalıştırıp metodun geri dönüş değerini, aradaki çeşitli katmanları kullanarak XML formatında elde edebiliriz.

- XML Belgeleri
- Visual Studio .Net içerisinde XML dosyası oluşturmak için;

New File			? 🛛
Categories: General Performance • Web Script	Iemplates: Visual Studio installed templates Text File Style Sheet XML Schema Bitmap File Cursor File Cursor File Visual C# Class Visual Basic Class	HTML Page KML File XSLT File Icon File Native Resource Template Visual J# Class	
A blank XML file.			Cancel

- <?xml version="1.0" encoding="utf-8" ?>
- <yemekler>
- <menu adı="Seçenek 1">
- corba>Mercimek</corba>
- ana_yemek>Döner</ana_yemek>
- <tatl>Kadayıf</tatl>
- /menu>
- <menu adı="Seçenek 2">
- corba>Domates Çorbası</corba>
- <ana_yemek>Balık</ana_yemek>
- tatli>Künefe</tatli>
- </menu>
- </yemekler>

Xml dosyasının browser üzerinde görüntülenmesi



- Belgeden de görüldüğü gibi her veri, kapanan ve açılan etiketler arasına yazılmıştır. Belgenin ilk satırında, çeşitli bildirimler yapılmaktadır. Bu bildirimler versiyon bilgisi ve karakter kodlamasıdır. Bu bildirimler bu kadar ile sınırlı değildir. Her XML belgesinin mutlaka bir kök elemanı olmalıdır. Bu örnekte kök eleman
- <yemekler>
-
- </yemekler>
- etiketi ile bildirilen elemandır. Kök elemanı olmayan bir XML belgesi geçerli değildir.
- Tıpkı bir ağaç yapısı gibi, yemekler elemanında iki adet menü elemanı bulunmaktadır.
 Menü elemanı ise farklı yemek çeşitlerinden oluşan 3 ayrı elemandan oluşmaktadır. XML belgelerindeki veriler açılan ve kapanan etiketler arasına yazılır. Örneğin
- <ana_yemek>Döner</ana_yemek>
- Etiketinde "Döner" bilgisinin bir ana yemek olduğu söylenmektedir. Ayrıca her etiket için farklı niteliklerde tanımlanabilir. Örneğin
- <menu adı="Seçenek 2">
- İfadesinde menü elemanının "adı" isminde bir niteliği vardır. Bu niteliğin değeri ise "Seçenek 2" şeklindedir.

- XmlDocument
- Load() Metodu: ASP.Net sayfasına Xml sayfasını yüklemek için kullanılır.
- OuterXml: Xml dosyasının içeriğine erişilir.
- Örnek: Kitaplar.xml

🚰 C:\Documents and Settings\İbrahim\Desktop\Bitirme Ödevi\XML\Uygulamal 🔳 🗖 🔀								
Dosya Düzen Görünüm Sık Kullanılanlar Araçlar Yardım 🥂								
🕞 Geri 🔹 🕞 🔹 🛃 🎾 Ara 🤺 Sik Kullanılanlar 🚱 🔗 🌺 💙								
Adres 🖭 C:\Documents and Settings\İbrahim\Desktop\Bitirme Ödevi\XML\Uygulamal 🗠 🔁 Git 🛛 Bağlantılar 🌺								
- <kitaplar> - <kitap> <baslik>C# Developer's Guide To ASP.NET, XML and ADO.NET</baslik> <yazar yer="San Francisco">Jeffrey P. McManus</yazar> <yazar yer="Seattle">Chris Kinsman</yazar> </kitap> </kitaplar>								
😂 Bilti 🚽 🚽 Bilgisayarım								

- using System.Xml; //veya <%@ import Namespace="System.Xml" %>
- <html xmlns="http://www.w3.org/1999/xhtml" >
- <SCRIPT runat="server">
- void Page_Load(Object Sender, EventArgs e)
- {
- XmlDocument xd = new XmlDocument();
- xd.Load (Server.MapPath("kitaplar.xml"));
- Response.Write (xd.OuterXml);
- xd = null;
- }
- </SCRIPT>

🖆 Untitled Page - Microsoft Internet Explorer										
<u>D</u> osya Dü <u>z</u> en <u>G</u> örünüm <u>S</u> ik Kullanılanlar <u>A</u> raçlar <u>Y</u> ardım										
🕞 Geri 👻 💿 👻 😰 🏠 🔎 Ara tk Kullanılanlar	@ • 🎍	o - 🧾	۳ ۲							
Adres 🚳 http://10.6.2.77/WebSite/Vtabani/Default3.aspx	💙 🄁 Git	Bağlantılar 🎽	🔁 🔹							
C# Developer's Guide To ASP.NET, XML and ADO.NETJeffrey P. McManusChris Kinsman										
🗃 Bitti	0	internet								

• System.Xml İsim Alanı

- XML belgeleri üzerinde işlem yapmak için geliştirilmiş sınıflar System.XML isim alanında bulunur. Bu isim alanından en çok kullanılanları aşağıda verilmiştir.
- **XmlReader:** XML belgelerinden verileri okumamız için tasarlanmış bir sınıftır. Tamponlama mekanizması ile çalışmamaktadır. Yani veriler direkt dosya sisteminden alınır.
- **XmlWriter:** Bir akıma yada dosya sistemine XML verilerini kaydetmek için kullanılır.
- XmlTextReader: XML belgelerinden sadece ileri yönde okuma yapabilir ve XML verilerine bir akım (stream) gibi erişmemizi sağlar.
- XmlTextWriter: XML akımlarını dosya sisteminde oluşturmak için kullanılır.
- XmlNode: XML belgesindeki tek bir düğümü temsil etmek için kullanılır.
- XmlNodeReader: Bu sınıf okumak için bir düğüm bilgisini kaynak olarak kullanır.
- **XmlDocument:** Bir XML belgesinin belleğe bir ağaç yapısı şeklinde yüklenmesini ve bu ağaç üzerinde çeşitli işlemler yapılabilmesini sağlar.
- XmlTextWriter ve XmlTextReader sınıfları XmlWriter ve XmlReader sınıfından türemiştir. XmlTextWriter ve XmlTextReader sınıfları XML belgesi ile bir akım gibi çalıştığı için daha yüksek performansa sağlanır. Bu iki grup sınıfın yaptığı işler tamamen aynıdır.

- <?xml version="1.0" encoding="utf-8" ?>
- <yemekler>
- <menu adı="Seçenek 1">
- <corba>Mercimek</corba>
- <yemek>
- <ana_yemek>Döner</ana_yemek>
- <ana_yemek>Adana</ana_yemek>
- </yemek>
- <tatl:>Kaday:f</tatl:>
- </menu>
- <menu adı="Seçenek 2">
- <corba>Domates Çorbası</corba>
- <yemek>
- <ana_yemek>Balık</ana_yemek>
- <ana_yemek>Tavuk</ana_yemek>
- </yemek>
- <tatli>Künefe</tatli>
- </menu>
- </yemekler>

ASP.NET İLE XML OLUŞTURMA VE OKUMA XmlTextReader

```
using System.Xml;
protected void Page Load(object sender, EventArgs e)
          string yol = "XMLFile.xml";
          XmlTextReader xtr = new XmlTextReader(yol);
          while (xtr.Read())
              if (xtr.NodeType == XmlNodeType.Element)
          {
              £
                  Response.Write(xtr.Name);
                   if (xtr.HasAttributes)
                   ł
                   Response.Write(xtr.AttributeCount+" niteliği var," );
                       while (xtr.MoveToNextAttribute())
                           Response.Write(xtr.Name + " " + xtr.Value);
                       Response.Write("<br>");
                   }
                  else
                       Response.Write("\tNitelik Yok..."+"<br>");
               }
          }
```

🚰 Untitled Page - Microsoft Internet Explorer 📃	
<u>D</u> osya Düzen <u>G</u> örünüm <u>Sık Kullanılanlar A</u> raçlar <u>Y</u> ardım	1
🕝 Geri 👻 💿 👻 😰 🏠 🔎 Ara Kullanılanlar	»
Adres 🗃 http://10.6.2.77/WebSite/Vtabani/D 🔽 🔁 Git 🛛 Bağlantılar 🎽	🔁 🔹
yemekler Nitelik Yok menul niteliği var,adı Seçenek 1 corba Nitelik Yok yemek Nitelik Yok ana_yemek Nitelik Yok tatlı Nitelik Yok menul niteliği var,adı Seçenek 2 corba Nitelik Yok yemek Nitelik Yok ana_yemek Nitelik Yok tatlı Nitelik Yok tatlı Nitelik Yok	
🕘 Bitti 🤍 🔮 Internet	

XmlTextWriter

- using System.Xml;
- using System.Collections;
- protected void Button1_Click(object
 sender, EventArgs e) {
- Hashtable sozluk = new Hashtable();
- sozluk.Add("Araba", "Car");
- sozluk.Add("Masa", "Table");
- sozluk.Add("Okul", "School");
- sozluk.Add("Bilgisayar", "Computer");
- •
- XmlTextWriter xw = new XmlTextWriter (Server.MapPath("uygulama.xml"), System.Text.Encoding.UTF8);
- xw.Formatting=Formatting.Indented;
- // Xml dosyasını grintili oluşturuyor.
- // Bu özellik olmasa yan yana yazar
- xw.WriteStartDocument();
- xw.WriteComment("TR-EN Sözlük");
- ww.WriteStartElement("Sozluk");
- int i=1;

- foreach (DictionaryEntry d in sozluk)
- •
- www.writeStartElement("kelime");
- xw.WriteAttributeString("No",i.ToString());
- xw.WriteStartElement("Türkçe");
- xw.WriteString(d.Key.ToString());
- xw.WriteEndElement();
- xw.WriteStartElement("ingilizce");
- xw.WriteString(d.Value.ToString());
- ww.WriteEndElement();
- www.writeEndElement();
- i++;
- }
- xw.WriteEndElement();
- xw.WriteEndDocument();
- xw.Close();
- }

- Uygulama.Xml
- <?xml version="1.0" encoding="utf-8"?>
- <!--TR-EN Sözlük-->
- <Sozluk>
- <kelime No="1">
- <Türkçe>Okul</Türkçe>
- <İngilizce>School</İngilizce>
- </kelime>
- <kelime No="2">
- <Türkçe>Bilgisayar</Türkçe>
- <İngilizce>Computer</İngilizce>
- </kelime>
- <kelime No="3">
- <Türkçe>Masa</Türkçe>
- <İngilizce>Table</İngilizce>
- </kelime>
- <kelime No="4">
- <Türkçe>Araba</Türkçe>
- <İngilizce>Car</İngilizce>
- </Sozluk>

ASP.NET İLE XML OLUŞTURMA VE OKUMA XmlDocument

- Attributes: İlgili düğüme ilişkin bir nitelik koleksiyonu döndürür. Bu koleksiyonun türü XmlAttributeCollection' dır.
- ParentNode: Bir düğüme ilişkin XmlNode sınıfı ile temsil edilmiş ana düğümü verir.
- HasChildNodes: Eğer ilgili düğüm bir başka yavru düğüm (child node) içeriyorsa true değeri döndürür.
- FirstChild: İlgili düğüme ilişkin ilk yavru düğümü XmlNode sınıfı türünden verir.
- LastChild: İlgili düğüme ilişkin son yavru düğümü XmlNode sınıfı türünden verir.
- ChildNodes: İlgili düğüme ilişkin yavru düğümleri bir XmlNodeList koleksiyonu şeklinde verir.
- NextSibling: Üzerinde çalışılan düğümden hemen sonra gelen düğümü verir.
- **PreviousSibling:** Üzerinde çalışılan düğümden bir önce gelen düğümü verir.
- InnerText: Bir düğümün içinde yavru düğümleri ve varsa düğümlerin değerlerini verir.
- InnerXml: Bir düğüme ilişkin bilgileri ve yavru düğümleri XML etiketleri ile beraber string türünden verir.
- **OuterXml:** Bir düğüme ilişkin bilgileri XML etiketleri ile beraber string türünden verir. Bu özellikte yavru düğümler alınmaz.
- Name: Bir düğümün ismini verir.
- Value: Bir düğüme ilişkin değeri verir.
- NodeType: Düğümün hangi tipte olduğunu XmlNodeType numaralandırması ile verir.

- Düğüm değerlerini elde etme
- protected void Page_Load(object sender, EventArgs e)
- {
- XmlDocument xd = new XmlDocument();
- xd.Load(Server.MapPath("uygulama.xml"));
- XmlNode ndkelime = xd.DocumentElement;
- foreach(XmlNode nd in ndkelime.ChildNodes)
- { if(nd.Name == "kelime")
- Response.Write(" " + nd.InnerText + "
");
- // Response.Write(" " + nd.OuterXml + "
");
- //Okul<İngilizce>School
- // Response.Write(" " + nd.InnerXml + "
");
- //Okul<İngilizce>School
- •
- }
- }

OkulSchool BilgisayarComputer MasaTable ArabaCar

- Nitelik değerini elde etme
- protected void Page_Load(object sender, EventArgs e)

```
• {
```

• XmlTextReader xtr = new XmlTextReader(Server.MapPath("uygulama.xml"));

```
while (xtr.Read())
```

{

}

}

}

{

```
if (xtr.NodeType == XmlNodeType.Element)
```

```
• if (xtr.HasAttributes)
```

```
    Response.Write(xtr.Name+" "+xtr.AttributeCount + " niteliği var,");
    while (xtr.MoveToNextAttribute())
```

```
Response.Write(xtr.Name + " " + xtr.Value);
```

```
Response.Write("<br>");
```

kelime 1 niteliği var,No 1 kelime 1 niteliği var,No 2 kelime 1 niteliği var,No 3 kelime 1 niteliği var,No 4
ASP.NET İLE XML OLUŞTURMA VE OKUMA XmlNode

- Yavru düğümleri elde etme
- protected void Page_Load(object sender, EventArgs e)
- {
- XmlTextReader xr = new XmlTextReader(Server.MapPath("uygulama.xml"));
- XmlDocument doc=new XmlDocument();
- doc.Load(xr);
- XmlNodeList xnl=doc.ChildNodes;
- Response.Write("Toplam Yavru Düğüm:"+xnl.Count+"\n");
- foreach(XmlNode xn in xnl)
- Response.Write("Yavru Düğüm Adı: "+ xn.Name+" Türü: "+xn.NodeType+"
);
- XmlElement xe=doc.DocumentElement;
- if(xe.HasChildNodes)
- { Response.Write("ilk Yavru Düğüm: "+xe.FirstChild.Name); }
- XmlNodeList xnl2=xe.FirstChild.ChildNodes;
- foreach(XmlNode xn in xnl2)
- Response.Write("Yavru Düğüm Adı: "+ xn.Name+" Türü: "+xn.NodeType+"
");
- }

Toplam Yavru Düğüm:3 Yavru Düğüm Adı: xml Türü: XmlDeclaration Yavru Düğüm Adı: #comment Türü: Comment Yavru Düğüm Adı: Sozluk Türü: Element İlk Yavru Düğüm: kelimeYavru Düğüm Adı: Türkçe Türü: Element Yavru Düğüm Adı: İngilizce Türü: Element

• Veritabanından alınan bilgiler ile xml dosyası oluşturma ve xls ile stil yapısı uygulama.

id	menu	yemek	icecek	
1	Menu1	Musakka	Cacık	
2	Menu2	İmam Bayıldı	Çorba	
3	Menu3	Karnı Yarık	Kola	

- OleDbConnection baglanti = new OleDbConnection("provider=microsoft.jet.oledb.4.0; data source=" + Server.MapPath("~/App_Data/xml.mdb"));
- OleDbCommand komut = new OleDbCommand("select menu,yemek,icecek from xml", baglanti);
- XmlTextWriter xr = new XmlTextWriter(Server.MapPath("xmlFile.xml"), Encoding.UTF8);
- xr.Formatting = Formatting.Indented;
- xr.WriteStartDocument(); // xsl dosyamızı burada(aşağıda) bildiriyoruz.
- String xsl = "type='text/xsl' href='xml.xsl'";
- xr.WriteProcessingInstruction("xml-stylesheet", xsl);
- // xml dosyamızı oluşturmaya başlıyoruz
- xr.WriteStartElement("yemekler");
- baglanti.Open();
- OleDbDataReader oku = komut.ExecuteReader();

- while (oku.Read())
- {
- xr.WriteStartElement("menu");
- xr.WriteAttributeString("menu", oku.GetString(0));
- xr.WriteElementString("yemek", oku.GetString(1));
- xr.WriteElementString("icecek", oku.GetString(2));
- xr.WriteEndElement();
- }
- baglanti.Close();
- xr.WriteEndDocument();
- xr.Close();
- }

- Xml'i oluşturduktan sonra projemizde xml'i şekillendirmek için eklediğimiz xsl dosyamızın kodları. Oluşturulan Xml dosyası html sayfası içeriğinden çalıştırılırsa xsl de belirtilen şablon uygulanarak son sayfadaki görüntü oluşur.
- <?xml version="1.0" encoding="utf-8"?>
- <xsl:stylesheet version="1.0" xmlns:xsl= "http://www.w3.org/1999
 /XSL/Transform">
- <xsl:output method="html" indent="yes"/>
- <xsl:template match="/">
- <html>
- <head>
- <style type="text/css">
 - body{ font-family:trebuchet ms; font-size:11pt;}
- .menu{ padding-left:10px; background-color:#f0ffff; width:200px;}
- .yemek{ width:200px; padding-left:30px;}
- .icecek{ width:200px; padding-left:30px;}
- .yazi{ color:#900;}
- </style>
- </head>

Yemek • Musakka

		rement i masarata		
•	<body></body>	İçecek : Cacık		
•	<div></div>	Menu2		
•	<pre><xsl:for-each select="yemekler/menu"></xsl:for-each></pre>	Yemek : İmam Bayıldı İçecek : Corba		
•	<div class="menu"></div>	Menu3		
•	<pre><xsl:value-of select="menu"></xsl:value-of></pre>	Yemek : Karnı Yarık		
•		İçecek : Kola		
•	<pre><div class="yemek"></div></pre>			
•	Yemek : <xsl:value-of sele<="" td=""><td>ect="yemek"/></td></xsl:value-of>	ect="yemek"/>		
•				
•	<pre><div class="icecek"></div></pre>			
•	İçecek : <xsl:value-of sele<="" td=""><td>ect="icecek"/></td></xsl:value-of>	ect="icecek"/>		

- </div>
- </xsl:for-each>
- </div>
- </body>
- </html>
- </xsl:template>
- </xsl:stylesheet>

XML belgesini Xsl ile Şekillendirme xml07_kitaplar.xml

- <?xml version="1.0"?>
- <?xml-stylesheet type="text/xsl" href="xml07_kitaplar.xsl"?>
- KITAP>
- <KOD>1</KOD>
- <KITAPADI>Stratejik İttifak</KITAPADI>
- <YAZAR>Alptekin Dursunoğlu</YAZAR>
- </KITAP>
- KITAP>
- KOD>2</KOD>
- <KITAPADI>Kılıç Yarası Gibi</KITAPADI>

- </KITAP>
- KOD>3</KOD>
- <KITAPADI>Şehir Romantiğinin Günlüğü</KITAPADI>
- <YAZAR>Buket Uzuner</YAZAR>
- <YAYINEVI>Remzi</YAYINEVI>
- </KITAP>
- </KITAPLAR>

Xsl belgesi

- <?xml version="1.0" encoding="utf-8"?>
- <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"</pre>
- xmlns:msxsl="urn:schemas-microsoft-com:xslt" exclude-result-prefixes="msxsl" >
- <xsl:output method="html" indent="yes"/>
- <xsl:template match="/">
- <html> <head>
- <style type="text/css">
 - body{ font-family:arial; font-size:12pt;}
- .kitap{background-color:#00ffff;}
- .yazi{ color:#0000FF;}
- </style>
- </head>
- ody>
- <center>
- <h1 class="yazi">Kitaplarım</h1>
- <TABLE >
- THEAD>
 - <TH WIDTH="50">Kod</TH>
 - <TH WIDTH="200">Kitap Adi</TH>
 - <TH WIDTH="200">Yazar</TH>
 - <TH WIDTH="150">Yayinevi</TH>
- <TH WIDTH="150">Ceviren</TH>
 - </THEAD>

Xsl belgesi

•	<xsl:for-each select="KITAPLAR/KITAP"></xsl:for-each>											
•	<pre><xsl:sort order="ascending" select="@KITAPADI"></xsl:sort></pre>											
•	<tr class="kitap</th><th>"></tr>											
•	<td <="" td="" width="50"><td><pre>> <xsl:value< pre=""></xsl:value<></pre></td><td>e-of select='</td><td>"KOD"/></td><td></td></td> <td></td> <td></td>	<td><pre>> <xsl:value< pre=""></xsl:value<></pre></td> <td>e-of select='</td> <td>"KOD"/></td> <td></td>	<pre>> <xsl:value< pre=""></xsl:value<></pre>	e-of select='	"KOD"/>							
	<td width="200</td><td><pre>"> <xsl:value< pre=""></xsl:value<></td> <td>e-of select='</td> <td>"KITAPADI"</td> <td>/> </td>	<xsl:value< pre=""></xsl:value<>	e-of select='	"KITAPADI"	/>							
	<td wtdth="200</td><td>"> <xsl:valu< td=""><td>e-of select='</td><td>"YAZAR"/></td><td></td></xsl:valu<></td> <td></td> <td></td>	<xsl:valu< td=""><td>e-of select='</td><td>"YAZAR"/></td><td></td></xsl:valu<>	e-of select='	"YAZAR"/>								
	<td wtdth="150</td><td>"> <xsl:value< td=""><td>e-of select='</td><td>"VAVTNEVT"</td><td></td><td></td><td></td></xsl:value<></td>	<xsl:value< td=""><td>e-of select='</td><td>"VAVTNEVT"</td><td></td><td></td><td></td></xsl:value<>	e-of select='	"VAVTNEVT"								
	<pre></pre>	"> <xsl:valu< td=""><td>e of select-</td><td>"CEVTREN" /</td><td></td><td></td><td></td></xsl:valu<>	e of select-	"CEVTREN" /								
			e-or serect-	CLVIREN /	///////////////////////////////////////							
•												
•							×					
•												

 C:\Users\etanyildiz\Desktop @ C:\Users\etanyildiz\Desktop @ C:\Users\etanyildiz\Desktop > | | | | | || • | | | | Кітаріа | rim | | |
		Ko	d Kitap Adi	Yazar	Yayinevi	Ceviren	
•		21	Bozkurt	H. C. Amstrong	Arba	Gül Çağalı Güven	
	(html)	6	Cevdet Bey ve Oğulları	Orhan Pamuk	İletişim		
		15	Davud	Gerald Messadie	Doğan Kitap	Ali Cevat Akkoyunlu	
•		19	Entellektüel	Edward Said	Ayrıntı	Tuncay Birkan	
	() NOT COMPENSION	24	Hatasız Kodlama	Steve Magiure	Arkadaş	Ceyhan Temürcü	
•		4	Hayvan Çiftliği	George Orwell	MEB	Halide Edib Adıvar	
	-	10	İletişim Çatışmaları ve Empati	Üstün Dökmen	Sistem		
		14	Katledilen Piramit	Christian Jacq	Doğan Kitap	Aysel Bora	
		2	Kılıç Yarası Gibi	Ahmet Altan	Can		
		26	Kimse Kizmasin Kendimi	Hasan Cemal	Doğan Kitap		~

Microsoft Windows Internet Information Services

IIS (İnternet Bilgi Servisleri -Internet Information Services) KURULUM Kaynaklar <u>https://bidb.itu.edu.tr/sevir-defteri/blog/2013/09/06/iis-(internet-bilgi-</u> servisleri---internet-information-services)

IIS Kurulumu

- İnternet Bilgi Servisleri olarak adlandırılan IIS (Internet Information Services), Web sayfaları yayınlamak, Web uygulamaları çalıştırmak ve Web sunucuları kurup yayınlamak için kullanılan servislerdir.
- IIS, Microsoft Windows sunucu tabanlı işletim sistemleriyle birlikte çalışmaktadır. Microsoft Windows işletim sistemi yüklü sunucularda istemcilerden HTTP, HTTPS, FTP gibi protokollerle gelen istekleri karşılar ve yanıt verir. Buna ek olarak .NET ortamında yazılım geliştirecek ve Web servisleri kullanacak yazılım geliştiriciler de IIS kurulu bir Windows işletim sistemine ihtiyaç duymaktadırlar.

IIS Kurulumu

- IIS ilk olarak Microsoft Windows NT 3.51 işletim sistemine ek olarak piyasaya sürülmüştür ve günümüze kadar geliştirilerek gelmiştir. Bu işletim sisteminden sonra çıkan, Windows XP Home Edition dışındaki bütün Windows işletim sistemlerine IIS kurulumu yapılabilir. Son sürümü Windows Server 2008 R2 ile birlikte sunulan IIS 7.5 dir.
- Bu sürüm IIS 7.0 ın üzerine eklenmiş, yönetilebilirliği ve ölçeklenebilirliği arttıran güncellemeler içermektedir. IIS 7.0 ile internet bilgi servisleri önceki sürümlerine göre daha güvenli ve sunucunun modüler yapısı sayesinde daha verimli bir sisteme sahiptir. Ayrıca günümüzde en yaygın kullanılan IIS sürümü IIS 7.0'dır.
- Bu sürümde servislerin bileşenleri modülleştirilmiştir ve sunucu ilk kurulduğunda yalnızca temel bileşenler yüklendiğinden daha güvenli bir yapı söz konusudur. Bu modüler mimarideki amaç sunucuya yapılabilcek saldırı yüzeyini en aza indirmektir. Ek olarak, ihtiyaç duyulan ayarlara göre modüller eklenerek veya kaldırılarak IIS istenilen yapılandırmalar için uygun hale getirilebilir.
- Önceki sürümlerde yeni sunucu ayarları için baştan kurulum yapmak gerekmekteydi. Söz konusu yapı hem zaman kaybına hem de Web uygulamalarının servis dışı kalması yol açmaktaydı. Ancak modüllerin eklenip kaldırılabilmesi yoluyla bu durumun önüne geçilerek IIS daha esnek ve daha kolay yönetilebilen bir yapıya sahip olmuştur.

IIS Kurulumu

- IIS 7.0 ın temel bileşenlerini Protokol Dinleyiciler (Protocol Listeners) ve Servisler olarak ikiye ayırmak mümkündür.
- Protokol dinleyicilere örnek olarak HTTP.sys verilebilir. HTTP.sys; HTTP ve HTTPS protokolleri ile gelen isteklerle ilgilenmektedir. Servisler WWW (World Wide Web Publishing Service) ve WAS (Windows Process Activation Service) olmak üzere iki tanedirler. WWW servisi protokol dinleyici bağdaştırıcısı olarak HTTP.sys protokol dinleyicisinin güncelleştirmelerinden ve yapılandırmalarından sorumlu olmakla birlikte, istek kuyruğuna yeni bir istek eklendiğinde bu isteği WAS servisine bildirmektende sorumludur. WAS servisi ise uygulama havuzu yapılandırmasından ve çalışan işlemlerden sorumludur. WAS servisinin çalışabilmesi için WWW servisine ihtiyaç yoktur çünkü WAS servisi HTTP protokolü dışındaki isteklere de cevap vermektedir.

IIS 7.0 daki modüler yapı ise **Ana Modüller (Native Modules)** ve **Yönetilebilir Modüller (Managed Modules)** olarak ikiye ayrılırlar. Ana modüller IIS 7.0 ile birlikte yüklenirler ve daha sonra yapılandırılabilme özelliğine sahiptirler. Yönetilebilir modüller ise sonradan eklenerek sunucunun işlevselliğini arttırmak için kullanılırlar. Her modül kendine özgü bir göreve sahiptir.

 Start (Başlat) >> Control Panel (Denetim Masası) >> Programs and Features (Programlar ve Özellikler) yolu izlendikten sonra karşılaşılan ekranda Turn Windows Features on or off (Windows Özelliklerini Aç yada Kapat) bağlantısına tıklanır.



(İnternet Information Açılan ekranda Internet Bilgi Services Servisleri) bölümüne gelinir.

0



- Internet Information Services (İnternet Bilgi Servisleri) kutucuğu işaretlendikten sonra Okey (Tamam) düğmesine tıklanarak işlem tamamlanır. Bu işlem sonucunda İnternet Bilgi Servisi güvenlik için yalnızca temel bileşenleri yükler. Bu temel bileşenler HTTP protokolünün çalışmasını sağlayan, gelen istekleri filtreleyen ve görüntüleyen, HTTP protokolünün yeniden yönlendirilmesini sağlayan bileşenlerdir.
- Bütün bileşenleri yüklemek için Internet Information Services altındaki bütün başlıkların işaretli olması gerekmektedir. Ancak kullanılmayacak özelliklerin yüklenmemesi güvenlik açısından önemlidir.

- Sunucuya yeni özellikler kazandırmak için aynı bölümde Internet Information Services (İnternet Bilgi Servisleri) başlığı altındaki bağlantılar kullanılır.
- ASP.NET platformunda yazılmış dinamik Web sayfaları yayınlamak için World Wide Web Services (WWW Servisleri) başlığı altında Application Development Features (Uygulama Geliştirme Özellikleri) bölümünde ASP ve ASP.NET özelliklerinin işaretli olması gerekmektedir. Bu iki özellik seçildiğinde çalışabilmeleri için gerekli olan ISAPI Filtreleri, ISAPI Uzantıları ve .NET Uzantıları da otomatik olarak yüklenmektedir.

 Bu işlem bittikten sonra istediğiniz dizine bir klasör oluşturunuz ve notepad'i (veya Visual Studio)açınız ve aşağıdaki kodları ekleyiniz. sayfayı açtığınız klasörün içine index.aspx olarak kaydediniz.



- Sonra Denetim Masası>Yönetimsel Araçlar>Internet Information Services(IIS) dizinini açın. ve Sol taraftaki Bağlantılar penceresindeki Default Web Site>Uygulama Ekle dizinine gidin. Burada gelen pencereden Diğer Ad'a servisinizin adını, fiziksel yola da oluşturduğunuz klasörün yolunu verin.
- Tarayıcıya http://localhost/Servis/index.aspx yolu giriniz.
 - a. http://localhost /Dosya yolu ve adı
 - b. http://127.0.0.1 /Dosya yolu ve adı
 - c. http://MakinaAdi /Dosya yolu ve adi
 - d. http://IpNumarasi /Dosya yolu ve adı
- Localhost: Lokal makina adı.
- 127.0.0.1 : Lokal IP numarası.
- Makina Adı: Ağ içerisindeki bilgisayar adı.
- Ip Numarası: Ağ içerisindeki Ip Numarası.



 FTP ile dosya yayınlamak için FTP Publishing Service (FTP Yayınlama Servisi) özelliğinin yüklü olması gerekmektedir. Ek olarak FTP Server (FTP Sunucusu) ve FTP Management Console (FTP Yönetim Kumandası) özellikleri kullanılarak FTP sunucusu kurulabilir ve FTP yayınlarını yönetmek için arayüz yüklenebilir.

- Start (Başlat) >> Control Panel (Denetim Masası) >> Administrative Tools (Yönetimsel Araçlar) >> IIS Manager (İnternet Bilgi Servisleri Yöneticisi) yolu izlenerek IIS yönetim arayüzü açılır. Sites (Siteler) başlığına sağ tıklanarak Add Web Site (Web Sitesi Ekle) seçilir.
- Gelen pencerede Site Name (Site Adı) kutusuna eklemek istenilen Web sitesinin ismi girilir.
- Yayınlanmak istenen sitenin dosyalarının tutulduğu klasörü gösteren dizinin yolu Physical Path (Fiziksel Yol) kutusuna yazılır.

- Kimlik denetimi ayarları için **Connect As (Farklı Bağlan)** düğmesine tıklanır.
- Gelen pencerede isteğe bağlı olarak bireysel bir kullanıcı veya uygulama havuzuna dahil olan kullanıcılar için kimlik denetimi tanımlanabilir. Yalnızca yerel bilgisayar için Web yayını yapılacaksa Specific User (Özel Kullanıcı) bölümünden bilgisayardaki yerel kullanıcılar tanımlanmalıdır. Ayarlar yapıldıktan sonra Okey (Tamam) düğmesine tıklanarak bu işlem kaydedilir.

- Bu işlemlerden sonra yapılan ayarlar Test Settings (Ayarları Test Et) düğmesine tıklanarak kontrol edilebilir.
- Binding (Bağlama) bölümünde bağlantı tipi olarak ihtiyaca göre http veya https seçilir.
- IP adresi ve port bölümünde ilgili sitenin yayınlanacağı ip adresi ve port numarası girilir. HTTP protokolü 80. porttan yayın yaptığı için varsayılan olarak port kısmında 80 gözükmektedir.

- Eğer varsa Host Name (Sunucu Adı) kısmında sitenin ilgili ip adresine atanmış olan site adı girilir.
- Ayarları kaydettikten sonra Web sitesi yayınının hemen başlaması için Start Web Site Immediately (Web Sitesini Hemen Başlat) kutucuğu işaretlenir ve Okey (Tamam) düğmesine tıklanarak bu bölümdeki işlemler tamamlanır.
- Yayınlanan Web sitesinin dosyalarının tutulduğu dizinin tarayıcı üzerinde hiyerarşik bir şekilde görüntülenebilmesi için ana menüde Directory Browsing (Dizin Tarama) üzerine çift tıklanarak ilgili bölüm açılır. Burada ekranın sağ bölümündeki menüden Enable (Etkinleştir) bağlantısı tıklanarak bu özellik aktif hale getirilir.

- Bu işlemler sonunda yayınlanmak istenen site ile ilgili ayarlar tamamlanmış olur. Bu aşamadan sonra yönetimsel ihtiyaçlar dahilinde modüller eklenip çıkartılabilir. Bunun için ana menüde Modules (Modüller) üzerine çift tıklanarak ayarlama menüsü açılır. Modül eklemek için Add Managed Modules (Düzenlenmiş Modül Ekle) seçeneğine tıklanır.
- Ana modüllerin ayarlarını değiştirmek için Configure Native Modules (Ana Modülleri Yapılandır) seçeneğine tıklanarak ayarlar yapılır.