

# İlk Ders:

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü

YMH111 Algoritma ve Programlama-I

Doç. Dr. Yaman Akbulut

# YMH111 Algoritma ve Programlama-I

- obs.firat.edu.tr
- debsis.firat.edu.tr
- yaz.tf.firat.edu.tr
- tek.firat.edu.tr
- <http://www.kriptarium.com/algoritma.html>
- JAVA
- NetBeans

## 2. Ders: Algoritma, Akış Diyagramları

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü

YMH111 Algoritma ve Programlama-I

Doç. Dr. Yaman Akbulut

# YMH111 Algoritma ve Programlama-I

- <http://www.kriptarium.com/algoritma.html> (Yardımcı kaynak)
- Algoritmaları anlamak ilk 14 (1-14) video.
- Algoritma
- Problem Çözümü
- Akış Diyagramları
- Örnekler

# Algoritma

- **Algoritma:** Bir problemi çözmek için sonlu sayıda, bir birini takip eden, iyi tanımlı adımlar kümesidir.

# Problem Çözümü

**girdi:**

Klavyeden, dosyadan veya başka bir aygıttan veriyi alma.

**çıktı:**

Ekranda veriyi görüntüleme veya veriyi bir dosya ya da başka bir aygıta gönderme.

**matematik:**

Toplama, çarpma gibi bazı temel matematiksel işlemleri gerçekleştirme.

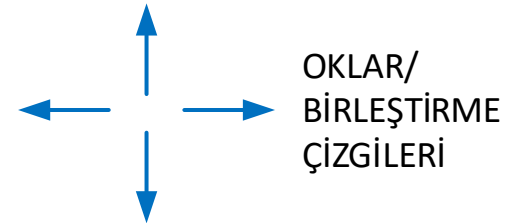
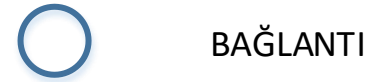
**koşullu yürütme:**

Belirli durumlar için sınaama yapma ve uygun cümle sırasını çalıştırma.

**tekrarlama:**

Bazı eylemleri genellikle ufak tefek bazı değişikliklerle tekrar tekrar yürütme.

# Akış Diyagramı Elemanları



# Örnek 1:

Bir öğrencinin öğrenci numarası, vize notu, final notu veriliyor. Öğrencinin bu bilgilerle ortalamasını hesaplayan algoritmayı ve akış diyagramını çiziniz.

# Örnek 1:

Bir öğrencinin  
öğrenci  
numarası, vize  
notu, final notu  
veriliyor.  
Öğrencinin bu  
bilgilerle  
ortalamasını  
hesaplayan  
algoritmayı ve  
akış diyagramını  
çiziniz.

## Örnek 2:

Bir öğrencinin öğrenci numarası, vize notu, final notu veriliyor. Vize notu ve final notunun ortalamaya olan katkısı sırasıyla %40 ve %60 şeklindedir. Öğrencinin bu bilgilerle ortalamasını hesaplayan algoritmayı ve akış diyagramını çiziniz.

# Örnek 2:

Bir öğrencinin öğrenci numarası, vize notu, final notu veriliyor.

Vize notu ve final notunun ortalamaya olan katkısı sırasıyla %40 ve %60 şeklindedir.

Öğrencinin bu bilgilerle ortalamasını hesaplayan algoritmayı ve akış diyagramını çiziniz.

## Örnek 3:

Bir öğrencinin öğrenci numarası, vize notu, final notu veriliyor. Vize notu ve final notunun ortalamaya olan katkısı sırasıyla %40 ve %60 şeklindedir. Ortalaması 50 ve üzerinde olan öğrenci o dersten BAŞARILI, diğer durumda BAŞARISIZ sayılacaktır. Öğrencinin bu bilgilerle ortalamasını hesaplayan ve başarı durumunu gösteren algoritmayı ve akış diyagramını çiziniz.

# Örnek 3:

Bir öğrencinin öğrenci numarası, vize notu, final notu veriliyor. Vize notu ve final notunun ortalamaya olan katkısı sırasıyla %40 ve %60 şeklindedir.

Ortalaması 50 ve üzerinde olan öğrenci o dersten BAŞARILI,

diğer durumda BAŞARISIZ sayılacaktır.

Öğrencinin bu bilgilerle ortalamasını hesaplayan ve başarı durumunu gösteren algoritmayı ve akış diyagramını çiziniz.

## Örnek 4:

Bir sınıftaki 150 öğrencinin Algoritma ve Prog.-I dersine ait dönem sonu ortalama notlarının sınıf ortalamasını hesaplayıp ekrana yazdıran algoritmayı ve akış diyagramını çiziniz.

# Örnek 4:

Bir sınıftaki 150  
öğrencinin  
Algoritma ve  
Prog.-I dersine ait  
dönem sonu  
ortalama  
notlarının sınıf  
ortalamasını  
hesaplayıp ekrana  
yazdıran  
algoritmayı ve akış  
diyagramını çiziniz.

# 3. Ders: Algoritma, Akış Diyagramları

(Ağıştırmalar ve Örnek Sınav Soruları)

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü

YMH111 Algoritma ve Programlama-I

Doç. Dr. Yaman Akbulut

# YMH111 Algoritma ve Programlama-I

- <http://www.kriptarium.com/algoritma.html> (Yardımcı kaynak)
- Algoritmaları anlamak (15-...) video.
- Problem Çözümü
- Akış Diyagramları
- Alıştırmalar
- Örnek Sorular

# Problem Çözümü

**girdi:**

Klavyeden, dosyadan veya başka bir aygıttan veriyi alma.

**çıktı:**

Ekranda veriyi görüntüleme veya veriyi bir dosya ya da başka bir aygıta gönderme.

**matematik:**

Toplama, çarpma gibi bazı temel matematiksel işlemleri gerçekleştirme.

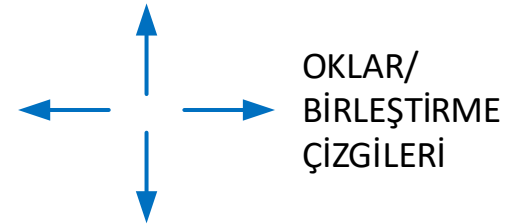
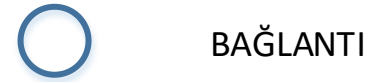
**koşullu yürütme:**

Belirli durumlar için sınaama yapma ve uygun cümle sırasını çalıştırma.

**tekrarlama:**

Bazı eylemleri genellikle ufak tefek bazı değişikliklerle tekrar tekrar yürütme.

# Akış Diyagramı Elemanları



# Alıştırma 1:

Döngü, başlangıç değeri, bitiş değeri, artırma/azaltma değeri ve karşılaştırma ifadesi kullanımı.

Ekrana 5 kez "Merhaba Dünya" yazan programın akış diyagramını çiziniz.

# Alıştırma 1:

Döngü,  
başlangıç  
değeri, bitiş  
değeri,  
artırma/azaltma  
değeri ve  
karşılaştırma  
ifadesi  
kullanımı.

Ekrana 5 kez  
"Merhaba  
Dünya" yazan  
programın akış  
diyagramını  
çiziniz.

# Alıştırma 2.a:

İndis (i, j, k, ... ) kullanımı.

Ekrana aşağıdaki çıktıyı yazan programın akış diyagramını çiziniz.

```
Merhaba Dünya 1  
Merhaba Dünya 2  
Merhaba Dünya 3  
Merhaba Dünya 4  
Merhaba Dünya 5
```

# Alıştırma 2.a:

İndis (i, j, k, ... )  
kullanımı.

Ekrana aşağıdaki  
çıktıyı yazan  
programın akış  
diyagramını çiziniz.

Merhaba Dünya 1  
Merhaba Dünya 2  
Merhaba Dünya 3  
Merhaba Dünya 4  
Merhaba Dünya 5

## Alıştırma 2.b:

İndis (i, j, k, ... ) kullanımı.

Ekrana aşağıdaki çıktıyı yazan programın akış diyagramını çiziniz.

```
Merhaba Dünya 10  
Merhaba Dünya 20  
Merhaba Dünya 30  
Merhaba Dünya 40  
Merhaba Dünya 50
```

# Alıştırma 2.b:

İndis (i, j, k, ... )  
kullanımı.

Ekrana aşağıdaki  
çıktıyı yazan  
programın akış  
diyagramını çiziniz.

```
Merhaba Dünya 10  
Merhaba Dünya 20  
Merhaba Dünya 30  
Merhaba Dünya 40  
Merhaba Dünya 50
```

# Soru 1:

Aşağıdaki sayıları döngü kullanarak ekrana yazan programın akış diyagramını çiziniz.

1, 2, 3, 4, 5, 6, 7, 8, 9, 10

# Soru 1:

Aşağıdaki  
sayıları  
döngü  
kullanarak  
ekrana  
yazan  
programın  
akış  
diyagramını  
çiziniz.

1, 2, 3, 4, 5,  
6, 7, 8, 9, 10

## Soru 2:

Aşağıdaki sayıları döngü kullanarak ekrana yazan programın akış diyagramını çiziniz.

$1/2, 2/5, 3/10, 4/17, 6/37, 7/50, 8/65, 9/82, 10/101$

## Soru 2:

Aşağıdaki  
sayıları döngü  
kullanarak  
ekrana yazan  
programın akış  
diyagramını  
çiziniz.

$1/2$ ,  $2/5$ ,  $3/10$ ,  
 $4/17$ ,  $6/37$ ,  
 $7/50$ ,  $8/65$ ,  
 $9/82$ ,  $10/101$

## Soru 3:

Boy, ağırlık ve cinsiyet bilgilerini dışarıdan parametre olarak alıp Beden Kitle İndeksini (BKİ) hesaplayan ve ekrana **KİLOLU** veya **KİLOLU DEĞİL** diye belirten bir program yazmanız istenmektedir.

Eğer cinsiyeti kadın ise BKİ 23 ve üstü kilolu olarak kabul edilmektedir. Eğer cinsiyeti erkek ise BKİ 25 ve üstü kilolu olarak kabul edilmektedir.

$$BKİ = Ağırlık \text{ (kg)} / (Boy \text{ (m)} * Boy \text{ (m)})$$

Örnek bir program çıktısı şu şekildedir:

Giriş

Cinsiyeti giriniz (E/K): E

Ağırlık (kg) bilgisini giriniz: 87

Boy (m) bilgisini giriniz: 1,78

Çıkış

BKİ: 27,5 KİLOLU

Bu işlemleri  
gerçekleştirecek  
programın akış  
diyagramını çizin.

# Soru 3:

Boy, ağırlık ve cinsiyet bilgilerini dışarıdan parametre olarak alıp Beden Kitle İndeksini (BKİ) hesaplayan ve ekrana **KİLOLU** veya **KİLOLU DEĞİL** diye belirten bir program yazmanız istenmektedir.

Eğer cinsiyeti kadın ise BKİ 23 ve üstü, eğer cinsiyeti erkek ise BKİ 25 ve üstü kilolu olarak kabul edilmektedir.

$$BKİ = \frac{Ağırlık(kg)}{(Boy(m) * Boy(m))}$$

Örnek bir program çıktısı şu şekildedir:

Giriş

Cinsiyeti giriniz (E/K): E  
Ağırlık (kg) bilgisini giriniz: 87  
Boy (m) bilgisini giriniz: 1,78

Çıkış

BKİ: 27,5 KİLOLU

Bu işlemleri gerçekleştirecek programın akış diyagramını çiziniz.

# 4. Ders: Algoritma, Akış Diyagramları (Çalışma Soruları)

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü

YMH111 Algoritma ve Programlama-I

Doç. Dr. Yaman Akbulut

# YMH111 Algoritma ve Programlama-I

- <http://www.kriptarium.com/algoritma.html> (Yardımcı kaynak)
- Algoritmaları anlamak (...) video.
- Problem Çözümü
- Akış Diyagramları
- Çalışma Soruları
- Örnekler

# Problem Çözümü

**girdi:**

Klavyeden, dosyadan veya başka bir aygıttan veriyi alma.

**çıktı:**

Ekranda veriyi görüntüleme veya veriyi bir dosya ya da başka bir aygıta gönderme.

**matematik:**

Toplama, çarpma gibi bazı temel matematiksel işlemleri gerçekleştirme.

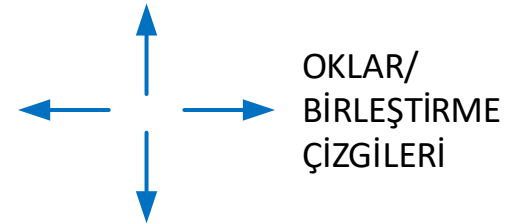
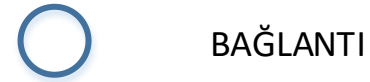
**koşullu yürütme:**

Belirli durumlar için sınaama yapma ve uygun cümle sırasını çalıştırma.

**tekrarlama:**

Bazı eylemleri genellikle ufak tefek bazı değişikliklerle tekrar tekrar yürütme.

# Akış Diyagramı Elemanları



# Koşul (Şart) ifadeleri

|    |                |
|----|----------------|
| >  | büyük mü?      |
| >= | büyük eşit mi? |
| <  | küçük mü?      |
| <= | küçük eşit mi? |
| == | eşit mi?       |
| != | eşit değil mi? |

# Artırma/Azaltma, Çarpma/Bölme, Mod

$i = i + 1$

i değişkenini 1 artır, i'ye (kaydet) atama yap.

$i = i + 2$

i değişkenini 2 artır, i'ye atama yap.

$i = i - 1$

i değişkenini 1 azalt, i'ye atama yap.

$i = i - 2$

i değişkenini 2 azalt, i'ye atama yap.

$i = i * 3$

i değişkenini 3 ile çarp, i'ye atama yap.

$i = i / 3$

i değişkenini 3 ile böl, i'ye atama yap.

$i = i \% 10$

i değişkene mod 10 işlemi uygula, kalanı i'ye kaydet.

# Çalışma Soruları 1:

1'den 100'e kadar olan tamsayıların toplamını hesaplayan programın akış diyagramını çiziniz.

$$1 + 2 + 3 + \dots + 100 = ?$$

## Çalışma Soruları 1:

1'den 100'e  
kadar olan  
tamsayıların  
toplamını  
hesaplayan  
programın  
akış  
diyagramını  
çiziniz.

$$1 + 2 + 3 + \dots + 100 = ?$$

## Çalışma Soruları 2:

1'den 100'e kadar olan tek tamsayıların toplamını hesaplayan programın akış diyagramını çiziniz.

$$1 + 3 + 5 + \dots + 99 = ?$$

## Çalışma Soruları 2:

1'den 100'e  
kadar olan  
tek  
tamsayıların  
toplamını  
hesaplayan  
programın  
akış  
diyagramını  
çiziniz.

$$1 + 3 + 5 + \dots \\ + 99 = ?$$

# Çalışma Soruları 3:

1'den 100'e kadar olan çift tamsayıların toplamını hesaplayan programın akış diyagramını çiziniz.

$$2 + 4 + 6 + \dots + 100 = ?$$

### Çalışma Soruları 3:

1'den 100'e  
kadar olan  
çift  
tamsayıların  
toplamını  
hesaplayan  
programın  
akış  
diyagramını  
çiziniz.

$$2 + 4 + 6 + \dots \\ + 100 = ?$$

## Çalışma Soruları 4:

Kullanıcı tarafından girilen  $n$  sayısına kadar olan tamsayıların toplamını hesaplayıp ekrana gösteren programın akış diyagramını çiziniz.

$$1 + 2 + 3 + \dots + n = ?$$

## Çalışma Soruları 4:

Kullanıcı  
tarafından  
girilen n sayısına  
kadar olan  
tamsayıların  
toplamını  
hesaplayıp  
ekrana gösteren  
programın akış  
diyagramını  
çiziniz.

$$1 + 2 + 3 + \dots + n \\ = ?$$

## Çalışma Soruları 5:

Kullanıcı tarafından girilen bir sayının pozitif ya da negatif olduğunu gösteren programın akış diyagramını çiziniz.

## Çalışma Soruları 5:

Kullanıcı  
tarafından  
girilen bir  
sayının pozitif  
ya da negatif  
olduğunu  
gösteren  
programın akış  
diyagramını  
çiziniz.

## Çalışma Soruları 6:

Kullanıcı tarafından girilen bir sayının tek ya da çift olduğunu gösteren programın akış diyagramını çiziniz.

## Çalışma Soruları 6:

Kullanıcı  
tarafından  
girilen bir  
sayının tek  
ya da çift  
olduğunu  
gösteren  
programın  
akış  
diyagramını  
çiziniz.

## Çalışma Soruları 7:

Kullanıcı tarafından girilen iki sayının birbirine göre durumunu (küçük, büyük, eşit) gösteren programın akış şemasını çiziniz.

## Çalışma Soruları 7:

Kullanıcı  
tarafından  
girilen iki  
sayının  
birbirine  
göre  
durumunu  
(küçük,  
büyük, eşit)  
gösteren  
programın  
akış  
şemasını  
çiziniz.

## Çalışma Soruları 8:

Kullanıcı tarafından girilen  $n$  sayısının faktöriyelini hesaplayıp ekrana gösteren programın akış diyagramını çiziniz.

$n! = ?$

## Çalışma Soruları 8:

Kullanıcı  
tarafından  
girilen  $n$   
sayısının  
faktöriyelini  
hesaplayıp  
ekrana  
gösteren  
programın akış  
diyagramını  
çiziniz.

$n! = ?$

# 5. Ders: Algoritma, Akış Diyagramları

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü

YMH111 Algoritma ve Programlama-I

Doç. Dr. Yaman Akbulut

# YMH111 Algoritma ve Programlama-I

- <http://www.kriptarium.com/algoritma.html> (Yardımcı kaynak)
- Algoritmaları anlamak (...) video.
- Problem Çözümü
- Akış Diyagramları
- Çalışma Soruları

# Problem Çözümü

**girdi:**

Klavyeden, dosyadan veya başka bir aygıttan veriyi alma.

**çıktı:**

Ekranda veriyi görüntüleme veya veriyi bir dosya ya da başka bir aygıta gönderme.

**matematik:**

Toplama, çarpma gibi bazı temel matematiksel işlemleri gerçekleştirme.

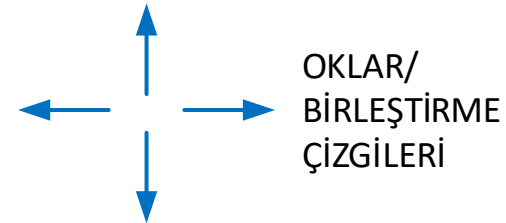
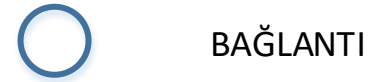
**koşullu yürütme:**

Belirli durumlar için sınama yapma ve uygun cümle sırasını çalıştırma.

**tekrarlama:**

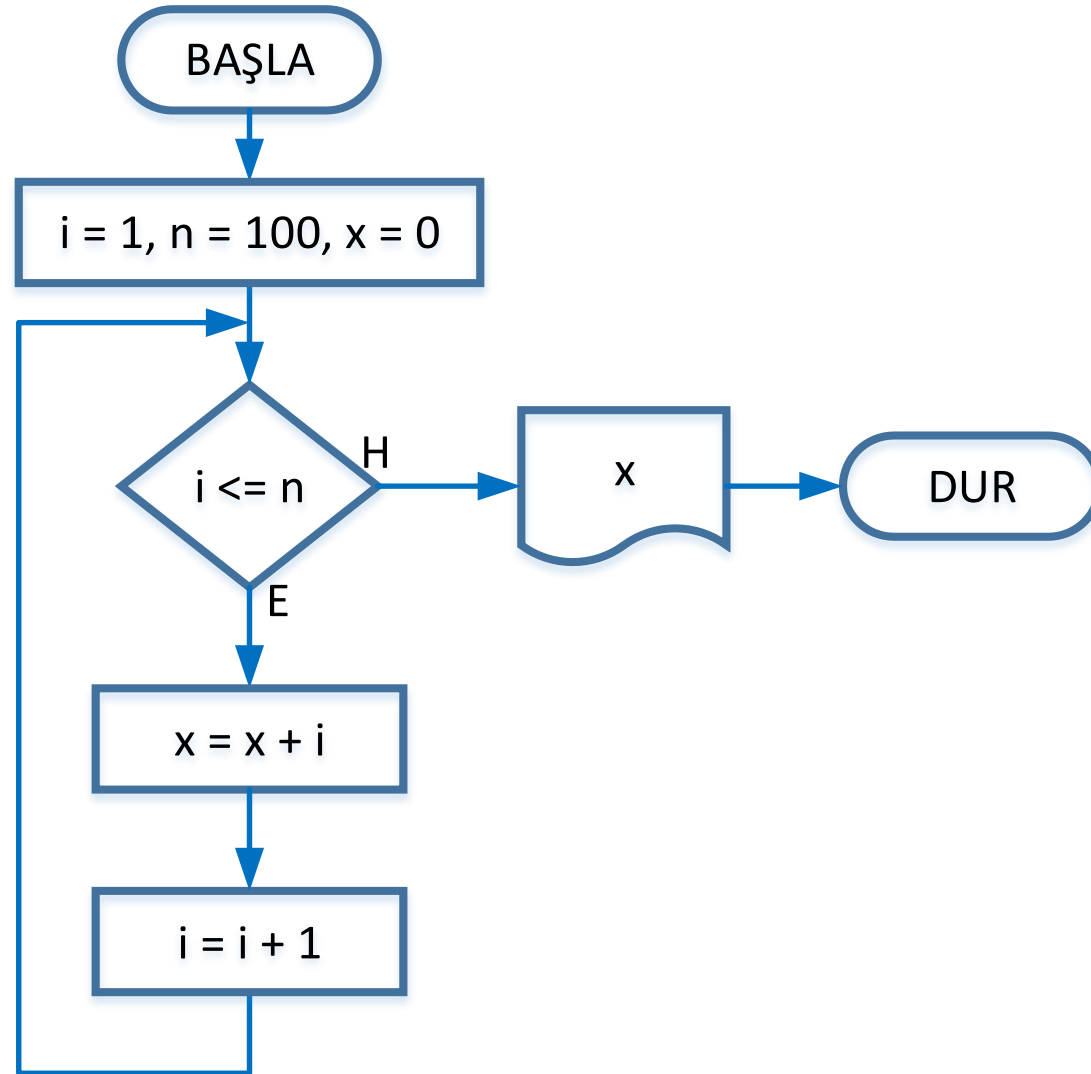
Bazı eylemleri genellikle ufak tefek bazı değişikliklerle tekrar tekrar yürütme.

# Akış Diyagramı Elemanları



## Soru 1:

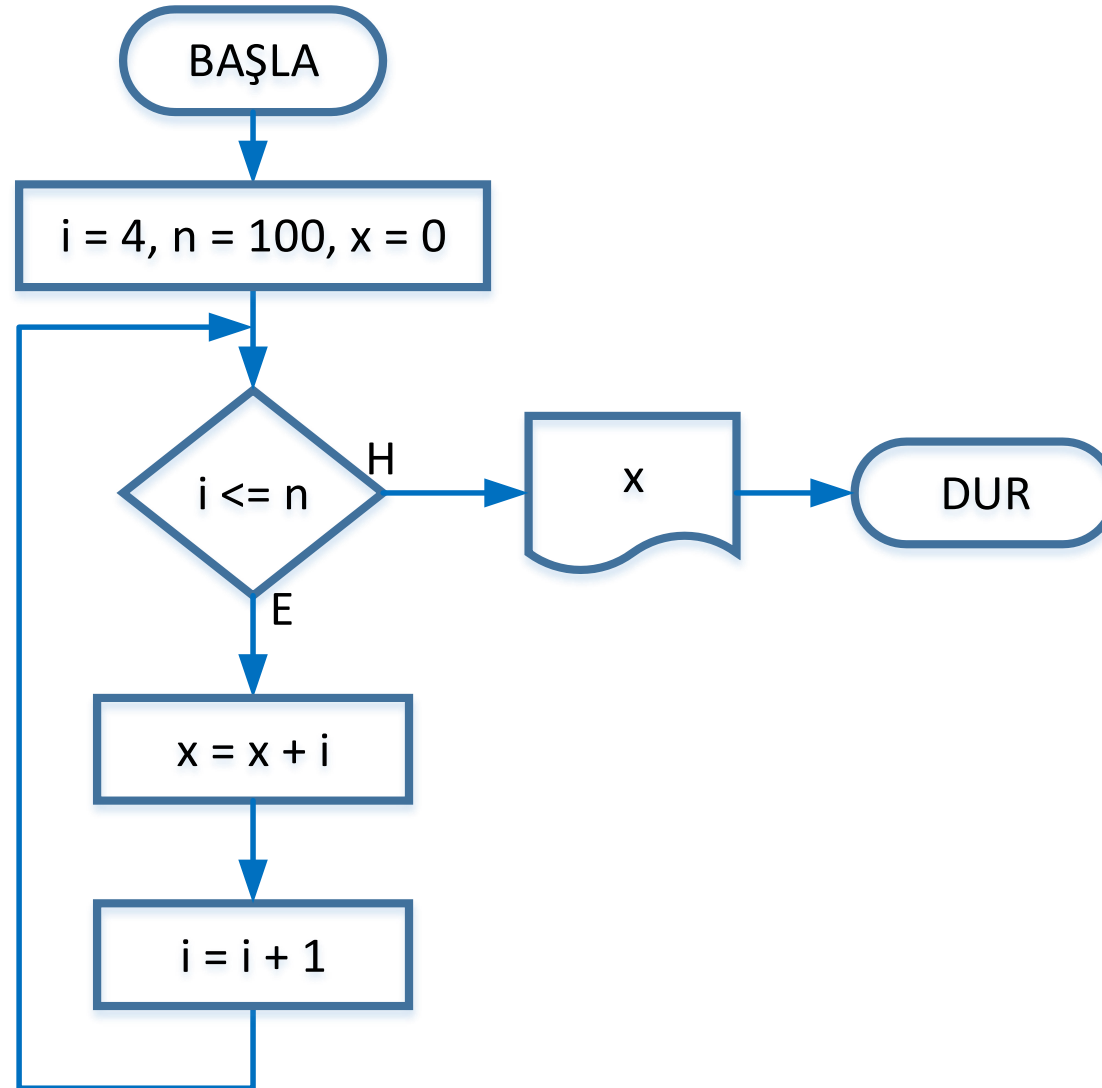
Akış diyagramı verilmiş olan algoritmanın çıktısı (sonucu) nedir, yazınız?



Yanıt:

## Soru 2:

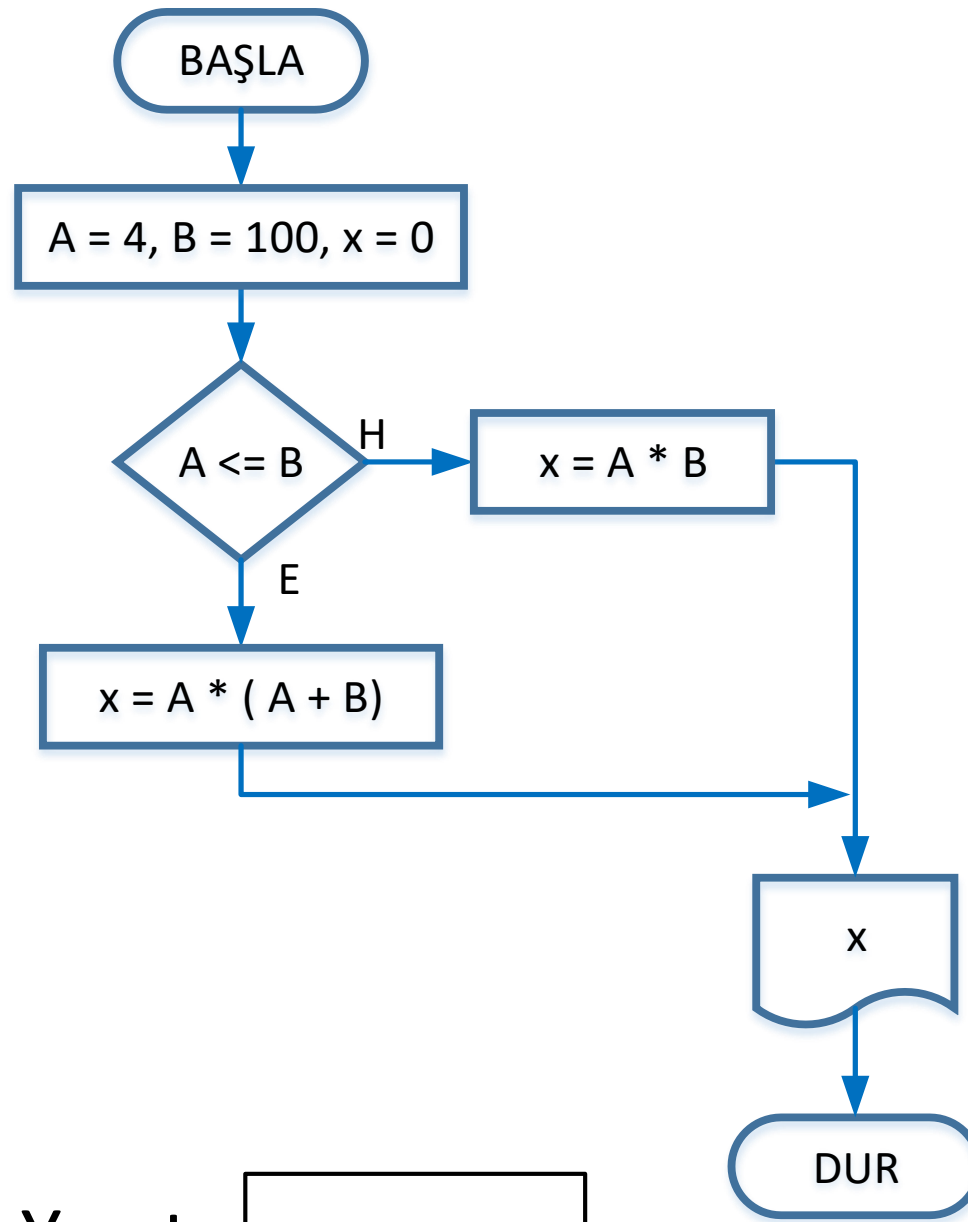
Akış diyagramı verilmiş olan algoritmanın çıktısı (sonucu) nedir, yazınız?



Yanıt:

### Soru 3:

Akış diyagramı verilmiş olan algoritmanın çıktısı (sonucu) nedir, yazınız?

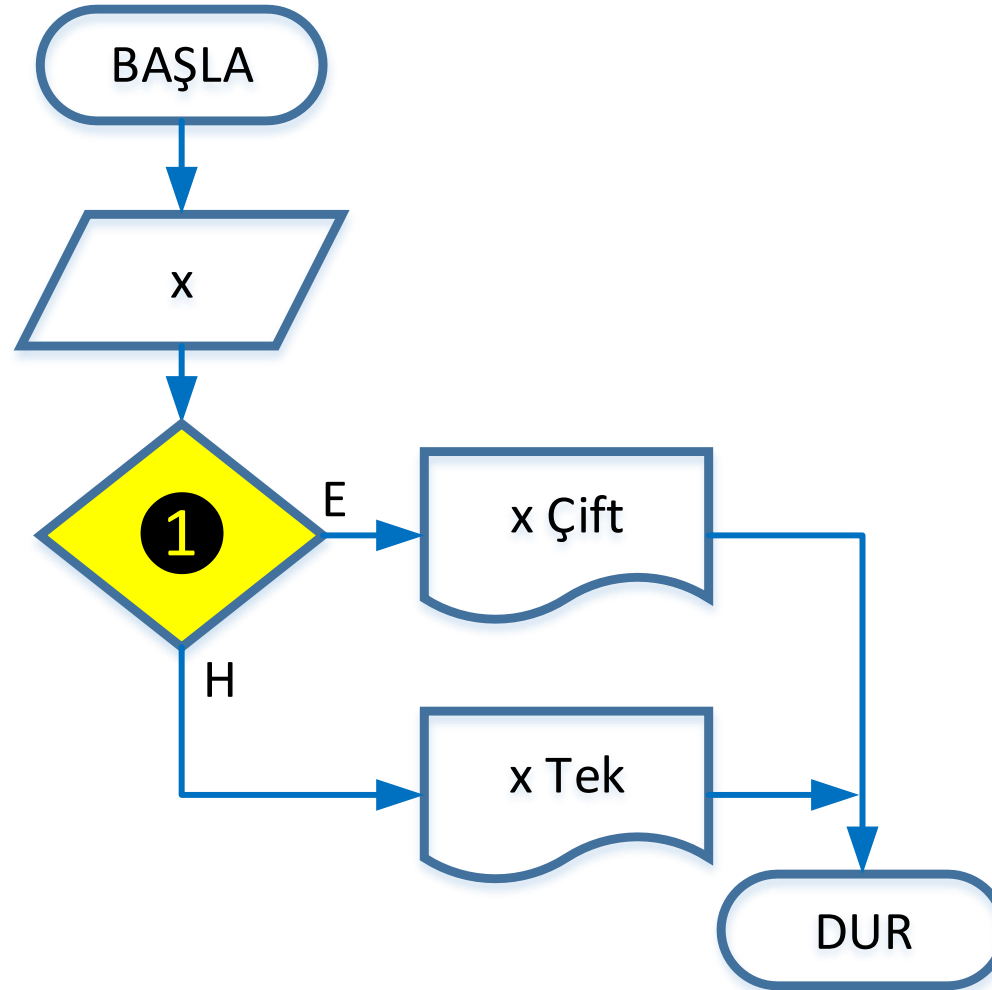


Yanıt:

## Soru 4:

Kullanıcı tarafından girilen bir sayının Çift ya da Tek olduğunu bulan programın akış diyagramı verilmiştir.

Etiketlenmiş ① numaralı bölüme gelmesi gereken ifadeyi yazınız.

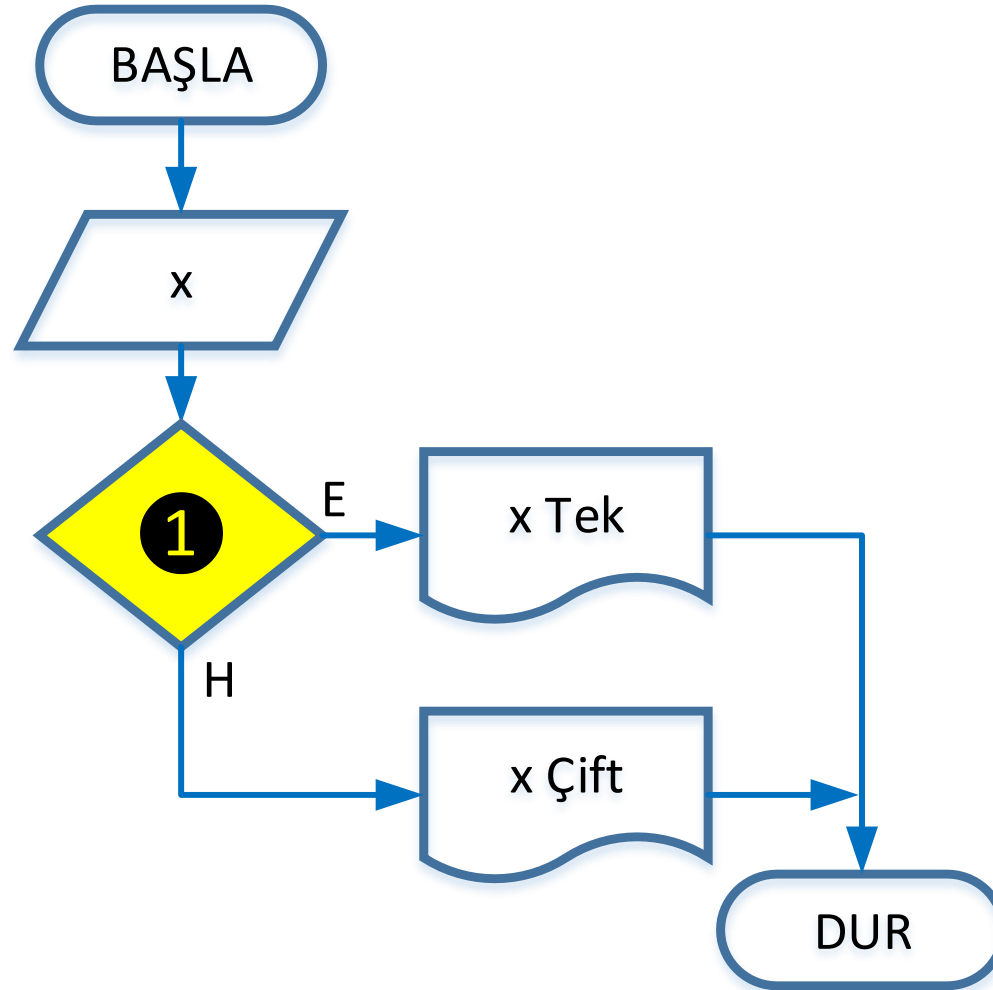


Yanıt:

## Soru 5:

Kullanıcı tarafından girilen bir sayının Çift ya da Tek olduğunu bulan programın akış diyagramı verilmiştir.

Etiketlenmiş ① numaralı bölüme gelmesi gereken ifadeyi yazınız.



Yanıt:

## Soru 6:

Brüt maaşı verilen bir çalışandan maaş miktarına göre vergi kesilmektedir. Brüt maaşı 5.000 TL'ye kadar olandan %15, brüt maaşı 5.000 TL ve üzeri olan çalışandan %20 vergi kesilmektedir.

**Net maaş = Brüt maaş - Vergi** denklemi kullanılarak çalışanların ne kadar net maaş alacağını hesaplayan programın akış diyagramı çizilmiştir.

Etiketlenmiş **1** numaralı bölüme gelmesi gereken ifadeyi yazınız.

## Soru 6:

Brüt maaşı verilen bir çalışandan maaş miktarına göre vergi kesilmektedir.

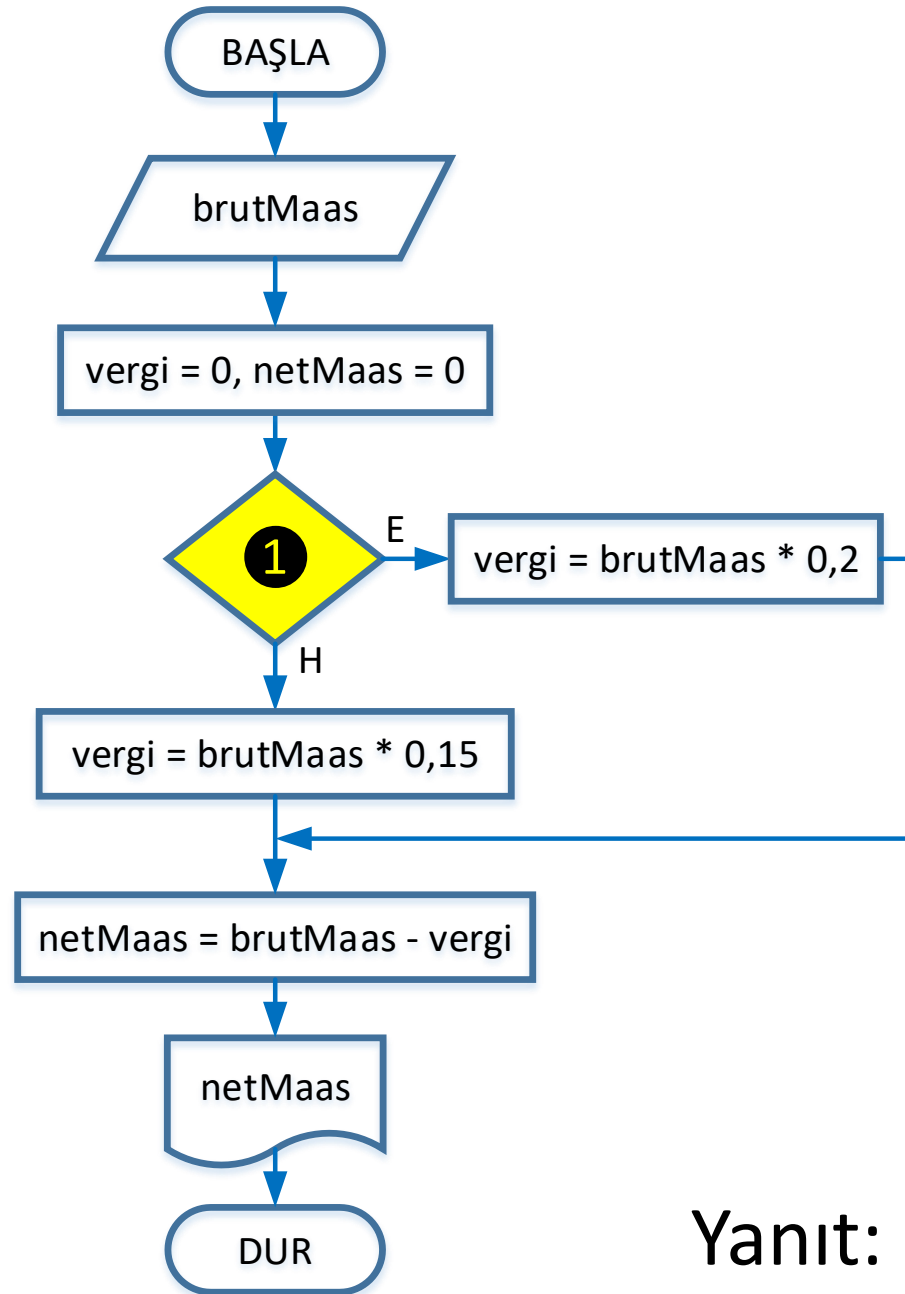
Brüt maaşı 5.000 TL'ye kadar olandan %15,

brüt maaşı 5.000 TL ve üzeri olan çalışandan %20 vergi kesilmektedir.

**Net maaş = Brüt maaş – Vergi**

denklemini kullanılarak çalışanların ne kadar net maaş alacağını hesaplayan programın akış diyagramı çizilmiştir.

Etiketlenmiş **1** numaralı bölüme gelmesi gereken ifadeyi yazınız.



Yanıt:

# Soru 7:

Brüt maaşı verilen bir çalışandan maaş miktarına göre vergi kesilmektedir.

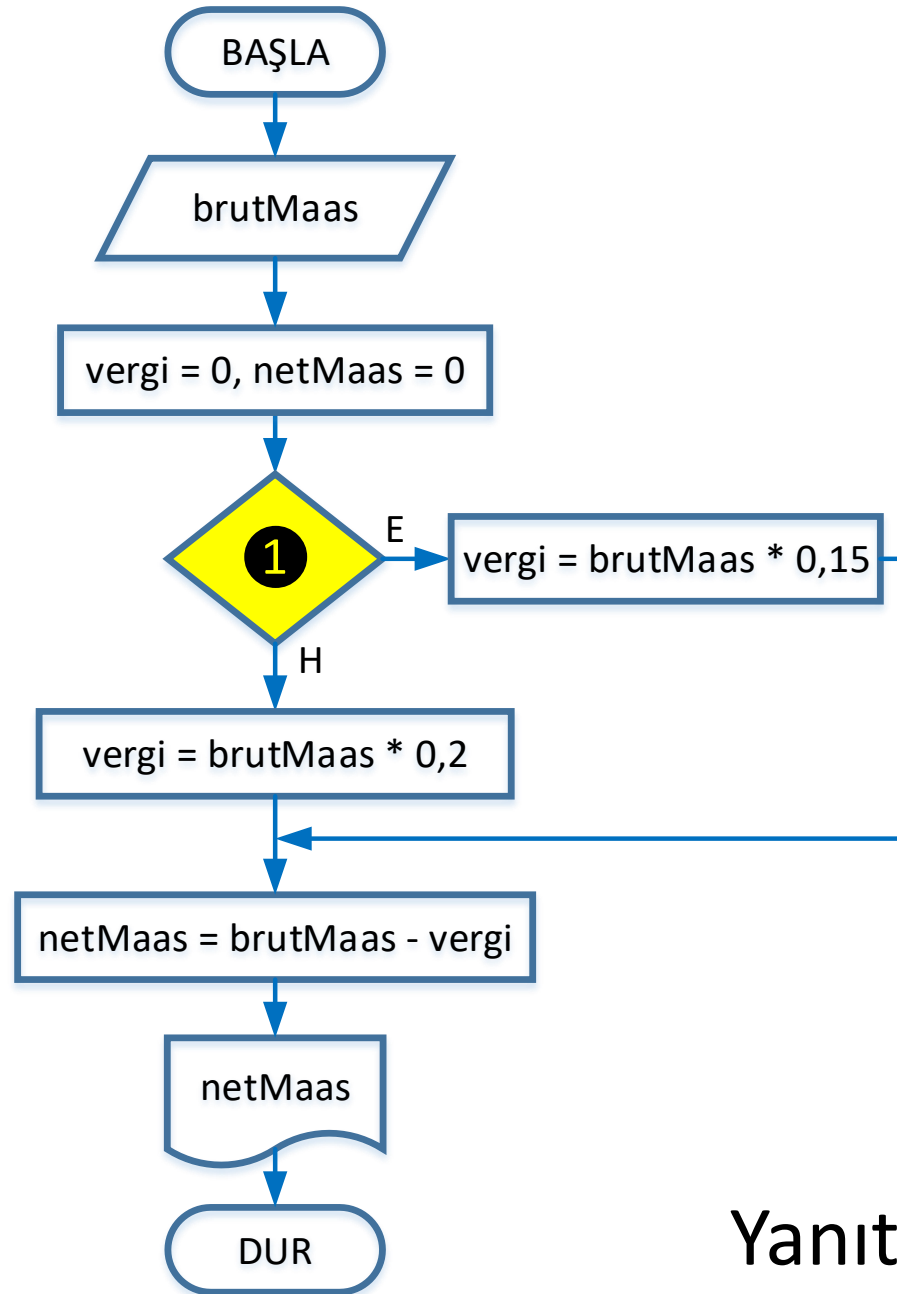
Brüt maaşı 5.000 TL'ye kadar olandan %15,

brüt maaşı 5.000 TL ve üzeri olan çalışandan %20 vergi kesilmektedir.

**Net maaş = Brüt maaş – Vergi**

denklemini kullanılarak çalışanların ne kadar net maaş alacağını hesaplayan programın akış diyagramı çizilmiştir.

Etiketlenmiş **1** numaralı bölüme gelmesi gereken ifadeyi yazınız.



Yanıt:

## Soru 8:

Brüt maaşı verilen bir çalışandan maaş miktarına göre vergi kesilmektedir. Brüt maaşı 5.000 TL'ye kadar olandan %15, brüt maaşı 5.000 TL ve üzeri olan çalışandan %20, **brüt maaşı 10.000 TL ve üzeri olandan %25 vergi kesilmektedir.**

**Net maaş = Brüt maaş - Vergi** denklemi kullanılarak çalışanların ne kadar net maaş alacağını hesaplayan programın akış diyagramı çizilmiştir.

Etiketlenmiş **1** numaralı bölüme gelmesi gereken ifadeyi yazınız.

# Soru 8:

Brüt maaşı verilen bir çalışandan maaş miktarına göre vergi kesilmektedir.

Brüt maaşı 5.000 TL'ye kadar olandan %15,

brüt maaşı 5.000 TL ve üzeri olan çalışandan %20,

brüt maaşı 10.000 TL ve üzeri olandan %25 vergi kesilmektedir.

**Net maaşı = Brüt maaş – Vergi**

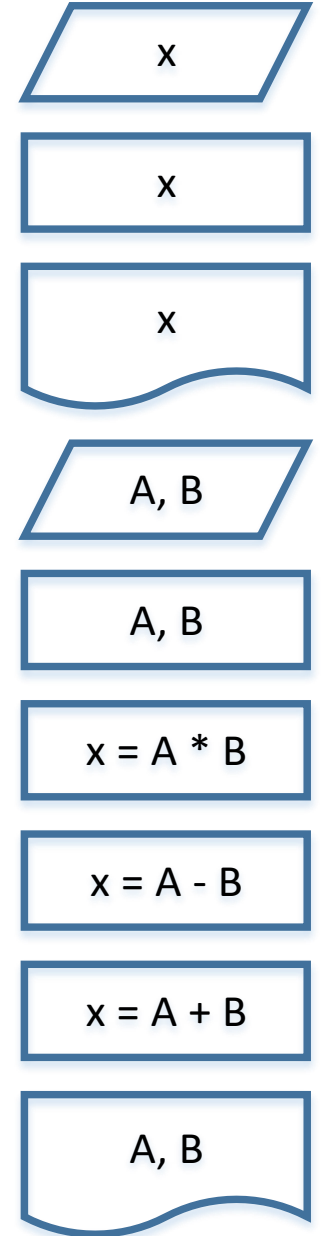
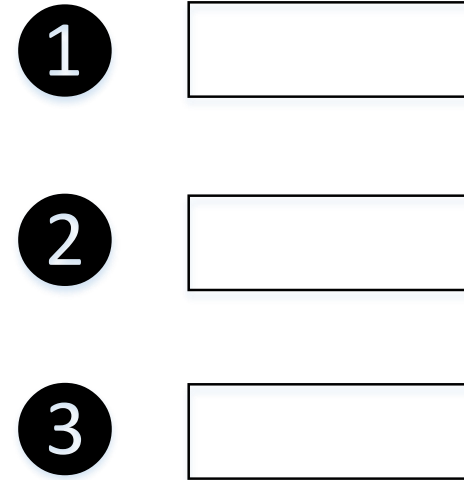
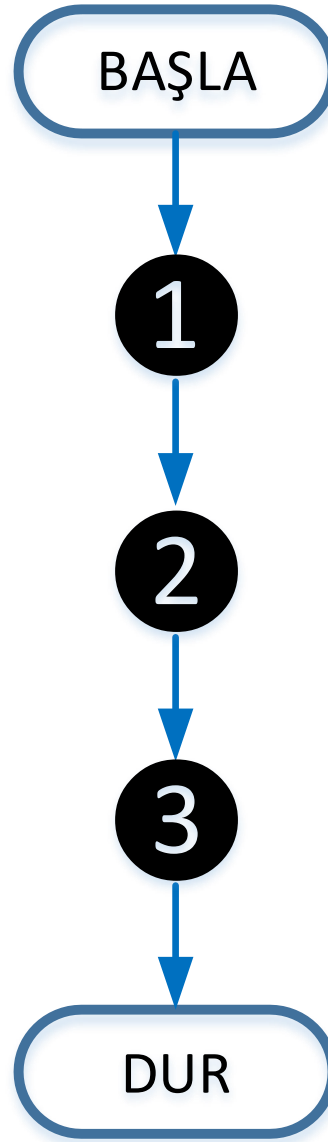
denklemini kullanarak çalışanların ne kadar net maaş alacağını hesaplayan programın akış diyagramı çizilmiştir.

Etiketlenmiş **1** numaralı bölüme gelmesi gereken ifadeyi yazınız.

## Soru 9:

Kullanıcı tarafından girilen iki sayıyı toplayıp sonucunu ekrana yazan programın akış diyagramını oluşturmanız istenmektedir.

Verilen akış diyagramında **1** **2** **3** numaralı yerlere gelecek parçaları eşleştiriniz.



# 6. Ders: JAVA Programlama Dili (JAVA'ya Giriş)

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü

YMH111 Algoritma ve Programlama-I

Doç. Dr. Yaman Akbulut

# JAVA Programlama Dili

- <http://www.kriptarium.com/algorithm.html> (Yardımcı kaynak)
- JAVA Programlama Dili (Ders: 1-10) video. (**Mutlaka İzleyiniz!**)
  - Ders 1: JAVA Giriş (izle)
  - Ders 2: JAVA Uygulama Alanları (izle)
  - Ders 3: Java Platformları / Sürümleri (izle)
  - Ders 4: Java Dilinin Tarihsel Gelişimi (izle)
  - Ders 5: JAVA İsmi Nereden Geliyor? (izle)
  - Ders 6: Java Dilinin Temel Özellikleri (NEDEN JAVA) (izle)
  - Ders 7: Java Neden Platformdan Bağımsızdır? (izle)
  - Ders 8: Java ve Güvenlik (izle)
  - Ders 9: Java dilinin diğer güçlü özellikleri (izle)
  - Ders 10: İlk JAVA Uygulamamız (izle)

# JAVA Programlama Dili

- JAVA bir programlama dili
- JAVA bir platform
- JAVA yüksek seviyeli bir dil
- JAVA nesne tabanlı programlama dili

# JAVA Kullanım Alanları

- Masaüstü uygulamaları
- Web uygulamaları
- Kurumsal uygulamalar
- Gömülü sistemler
- Mobil uygulamalar
- Akıllı kartlar
- Robotik
- Oyunlar

# JAVA Uygulama Tipleri

- Standalone application
- Web application
- Enterprise application
- Mobile application

# JAVA Platformları

1. JAVA Standard Edition (SE)
2. JAVA Enterprise Edition (EE)
3. JAVA Micro Edition (ME)
4. JAVAFX

# JAVA'nın Versiyonları

- ...

# JAVA'nın Özellikleri

1. Simple (Basit)
2. Object-Oriented (Nesne Tabanlı)
3. Portable (Taşınabilir)
4. Platform Independent (Platform bağımsız)
5. Secured (Güvenli)
6. Robust (Sağlam)
7. Architecture neutral (Mimari Nötr)
8. Interpreted (Yorumlanır)
9. High performance (Yüksek Performanslı)
10. Multithreaded (İş Parçacıklı)
11. Distributed (Dağıtık)
12. Dynamic (Dinamik)

# Object-Oriented Programming Concepts (Nesne Tabanlı Programlama kavramları)

- Object (Nesne)
- Class (Sınıf)
- Inheritance (Kalıtım)
- Polymorphism (Polimorfizm)
- Abstraction (Soyutlama)
- Encapsulation (Kapsülleme)

# Programlama Dili Seçim Kriterleri

1. Okunabilirlik
2. Yazılabilirlik
3. Güvenilirlik
4. Maliyet

# JAVA'da program yazma ortamı

1. Java Development Kit (JDK) + notepad (yazı editörü)

Java geliştirme kiti (seti)

2. Integrated Development Environment (IDE)

Entegre geliştirme ortamı

# İlk JAVA programı 1:

```
1 public class Merhaba
2 {
3     public static void main(String[] args)
4     {
5         //Konsol penceresinde selamlama
6
7         System.out.println("Merhaba, Dünya!");
8     }
9 }
```

Program çıktısı:

Merhaba, Dünya!

# İlk JAVA programı 2:

```
1 public class Merhaba{  
2     public static void main(String[] args){  
3         //Konsol penceresinde selamlama  
4  
5         System.out.println("Merhaba, Dünya!");  
6     }  
7 }
```

Program çıktısı:

Merhaba, Dünya!

# class (sınıf) yapısı

```
public class SinifAdi {
```

```
    ...
```

```
}
```

# method (metot, yöntem) yapısı

```
public static void main(String[] args) {  
  
    ...  
  
}
```

# parameter (değişken) yapısı

String[] args

**comment** (yorum) yapısı

//Konsol penceresinde selamlama

# Reserved Words

- class
- public
- private
- static
- void
- main

# İlk JAVA programı (Çalıştır)

```
1 public class Merhaba{
2     public static void main(String[] args){
3         //Konsol penceresinde selamlama
4         System.out.println("Merhaba, Dünya!");
5     }
6 }
```

Java Development Kit (JDK) +  
notepad ile program çalıştır

1. Java Development Kit (JDK) yükle
2. Not defterini (Notepad) aç
3. Merhaba.java kaynak dosyası oluştur (yandaki kodu yaz)
4. java Merhaba.java (Compile, Derle)
5. java Merhaba (Execute, Çalıştır)

Program çıktısı:  
Merhaba, Dünya!

# JAVA temel veri tipleri

# JAVA deęiřken tanımlama

# 7. Ders: JAVA Programlama Dili (Program çalıştırma, Değişkenler, Alistirmalar (Ödev))

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü

YMH111 Algoritma ve Programlama-I

Doç. Dr. Yaman Akbulut

# JAVA Programlama Dili

- <http://www.kriptarium.com/algorithm.html> (Yardımcı kaynak)
- JAVA Programlama Dili (Ders: 11-22) video. (Mutlaka İzleyiniz!)
  - Ders 11: JAVA programlarının derlenmesi ve yorumlanması (izle)
  - Ders 12: JVM, JRE ve JDK arasındaki temel farklar (izle)
  - Ders 13: Java Değişkenler (izle)
  - Ders 14: Değişkenler ile alakalı kod örnekleri (izle)
  - Ders 15: Veri Tipleri (izle)
  - Ders 16: Java İlkel/Primitive Veri Tipleri (izle)
  - Ders 17: Değişken Aralıkları (izle)
  - Ders 18: Java Operatörler (izle)
  - Ders 19: Operatör Örnekleri 1 (x++ ve ++x) (izle)
  - Ders 20: Operatör Örnekleri 2 (izle)
  - Ders 21: Mantıksal ve Aritmetiksel Operatörler (izle)
  - Ders 22: Kaydırma Operatörleri (izle)

# JAVA Programlama Dili

- Java Development Kit (JDK)
- Java Runtime Environment (JRE)
- Java Virtual Machine (JVM)

# JAVA'da program yazma ve çalıştırma ortamı

1. Java Development Kit (JDK) (Java geliştirme kiti (seti))  
+ Windows notepad (yazı editörü) (alternatif [notepad++](#))

2. Integrated Development Environment (IDE)

Entegre geliştirme ortamı

- NetBeans
- Eclipse
- IntelliJ IDEA
- VS Code

# İlk JAVA programı (Derle & Çalıştır)

```
1 public class Merhaba {  
2     public static void main(String[] args) {  
3         //Konsol penceresinde selamlama  
4  
5         System.out.println("Merhaba, Dünya!");  
6     }  
7 }
```

Java Development Kit (JDK) +  
notepad ile program çalıştır

1. Java Development Kit (JDK) yükle
2. Not defterini (Notepad) aç
3. Merhaba.java kaynak dosyası oluştur (yandaki kodu yaz)
4. **javac** Merhaba.java (Compile, Derle)
5. **java** Merhaba (Execute, Çalıştır)

Program çıktısı:  
**Merhaba, Dünya!**

# JDK kurulum (Java SE 15.0.1 JDK)

<https://www.oracle.com/java/technologies/javase-downloads.html>



jdk-15.0.1\_windo  
ws-x64\_bin.exe

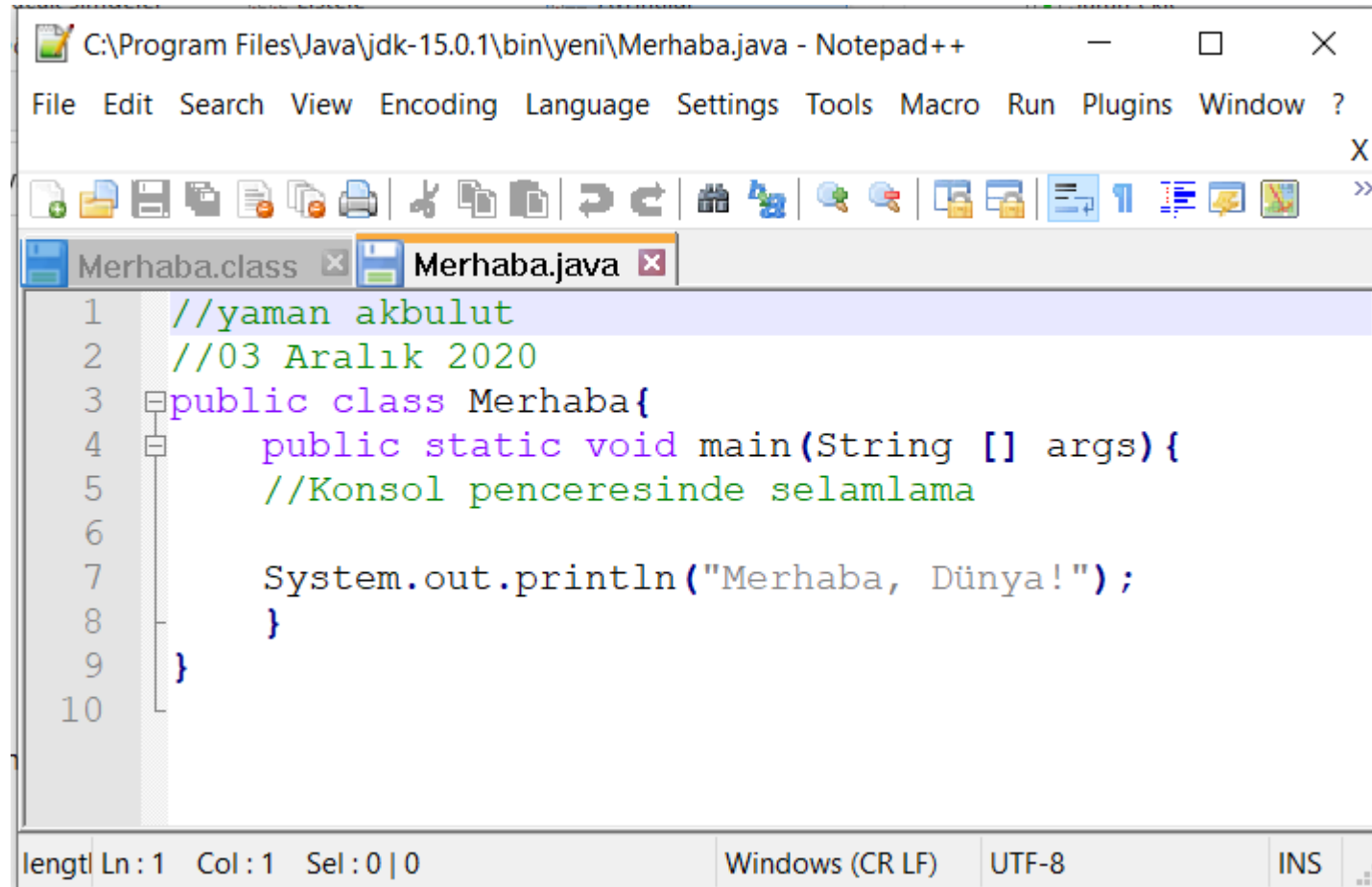
Yerel Disk (C:) > Program Files > Java > jdk-15.0.1 > bin

| Ad                                       | Değiştirme tarihi | Tür               | Boyut |
|------------------------------------------|-------------------|-------------------|-------|
| server                                   | 3.12.2020 22:29   | Dosya klasörü     |       |
| yeni                                     | 3.12.2020 22:37   | Dosya klasörü     |       |
| api-ms-win-core-console-l1-1-0.dll       | 3.12.2020 22:29   | Uygulama uzantısı | 12 KB |
| api-ms-win-core-console-l1-2-0.dll       | 3.12.2020 22:29   | Uygulama uzantısı | 12 KB |
| api-ms-win-core-datetime-l1-1-0.dll      | 3.12.2020 22:29   | Uygulama uzantısı | 12 KB |
| api-ms-win-core-debug-l1-1-0.dll         | 3.12.2020 22:29   | Uygulama uzantısı | 12 KB |
| api-ms-win-core-errorhandling-l1-1-0.dll | 3.12.2020 22:29   | Uygulama uzantısı | 12 KB |
| api-ms-win-core-file-l1-1-0.dll          | 3.12.2020 22:29   | Uygulama uzantısı | 15 KB |
| api-ms-win-core-file-l1-2-0.dll          | 3.12.2020 22:29   | Uygulama uzantısı | 12 KB |
| api-ms-win-core-file-l2-1-0.dll          | 3.12.2020 22:29   | Uygulama uzantısı | 12 KB |
| api-ms-win-core-handle-l1-1-0.dll        | 3.12.2020 22:29   | Uygulama uzantısı | 12 KB |
| api-ms-win-core-heap-l1-1-0.dll          | 3.12.2020 22:29   | Uygulama uzantısı | 12 KB |

path=C:\Program Files\Java\jdk-15.0.1\bin

# Merhaba.java

# (Notepad++)



The screenshot shows the Notepad++ application window. The title bar reads "C:\Program Files\Java\jdk-15.0.1\bin\yeni\Merhaba.java - Notepad++". The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, and Window. The toolbar contains various icons for file operations and editing. The tab bar shows two tabs: "Merhaba.class" and "Merhaba.java", with the latter being the active tab. The editor area displays the following Java code:

```
1 //yaman akbulut
2 //03 Aralık 2020
3 public class Merhaba{
4     public static void main(String [] args){
5         //Konsol penceresinde selamlama
6
7         System.out.println("Merhaba, Dünya!");
8     }
9 }
10
```

The status bar at the bottom shows "lengt Ln : 1 Col : 1 Sel : 0 | 0", "Windows (CR LF)", "UTF-8", and "INS".

# Özel Karakterler

{ } braces

( ) parentheses

[ ] brackets

// double slashes

" " quotation marks

; semicolon

/\*

for block comments

\*/

/\*\*

for documentation

\*/

# cmd, cd, dir, path...

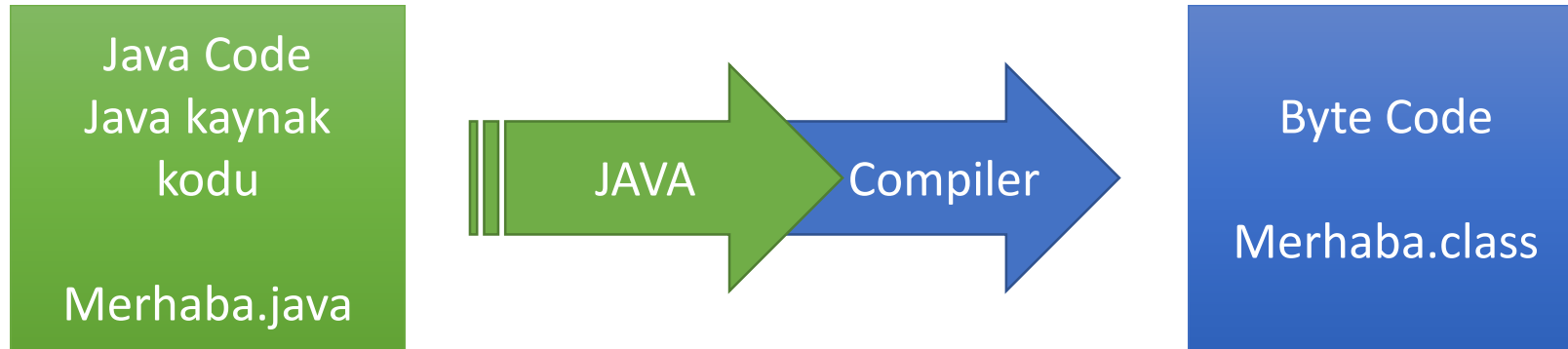
```
C:\Program Files\Java\jdk-15.0.1\bin>cd yeni

C:\Program Files\Java\jdk-15.0.1\bin\yeni>dir
Volume in drive C has no label.
Volume Serial Number is E6A7-B5BC

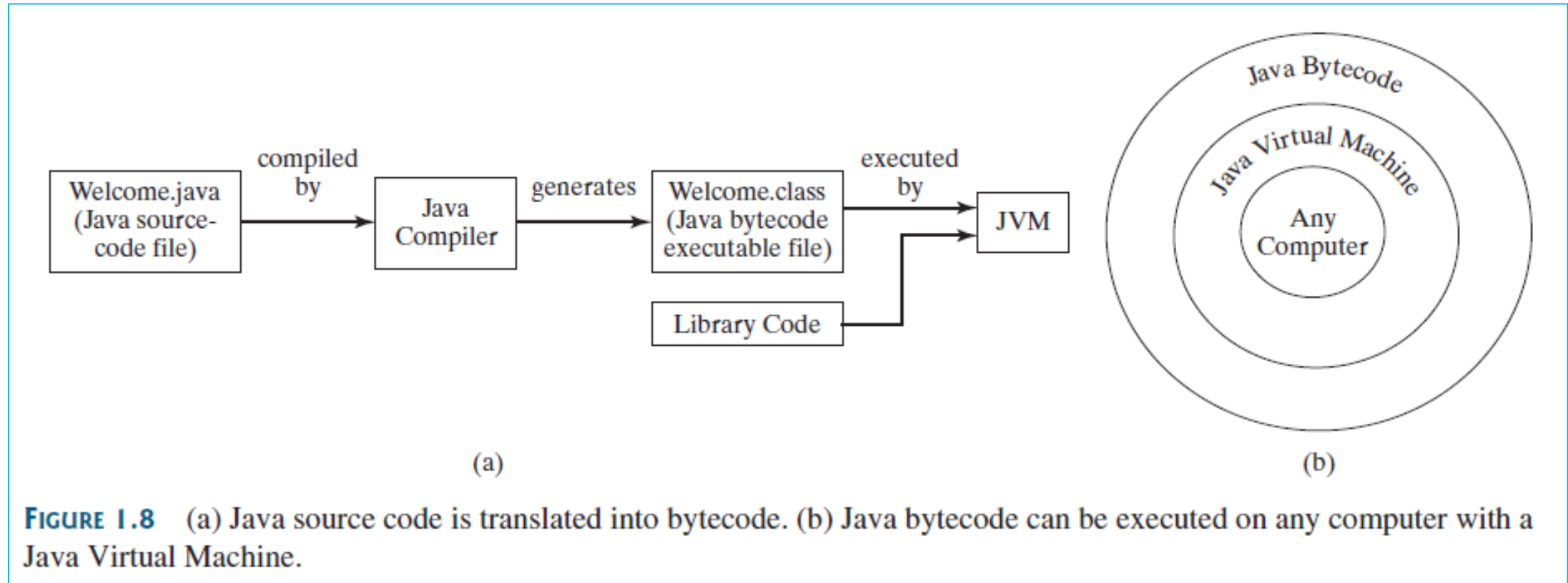
Directory of C:\Program Files\Java\jdk-15.0.1\bin\yeni

03.12.2020  22:34    <DIR>          .
03.12.2020  22:34    <DIR>          ..
03.12.2020  22:36                188 merhaba.java
               1 File(s)                188 bytes
               2 Dir(s)  93.748.731.904 bytes free
```

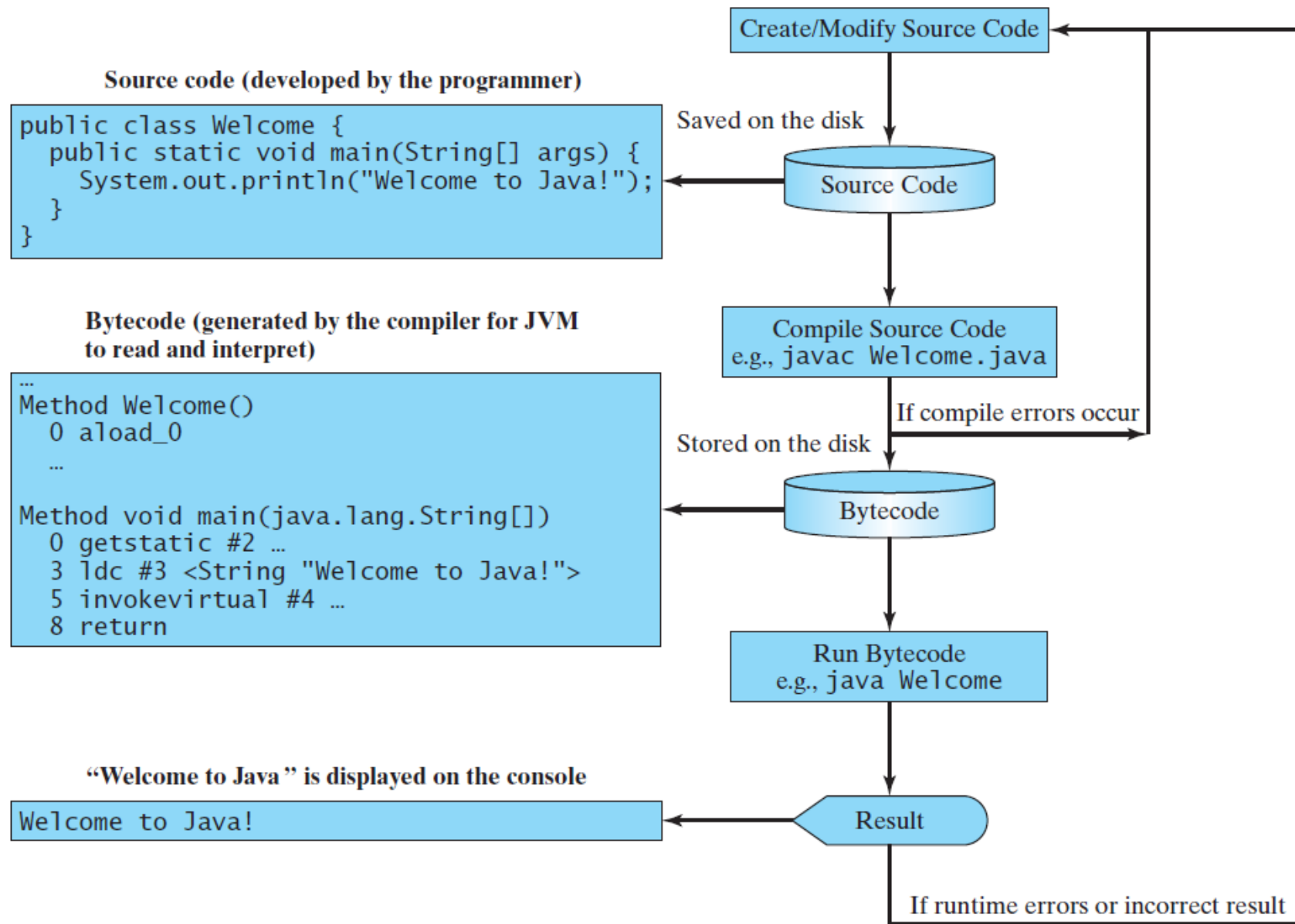
# JAVA compiler (Derleme)



# JAVA compiler (Derleme)



Kaynak: Introduction to JAVA Programming 10th Edition, Y. Daniel Liang



**FIGURE 1.6** The Java program-development process consists of repeatedly creating/modifying source code, compiling, and executing programs.

# javac Merhaba.java

BÜYÜK küçük Harfe DİKKAT!!!

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>javac merhaba.java
merhaba.java:3: error: class Merhaba is public, should be declared in a file named Merhaba.java
public class Merhaba{
      ^
1 error

C:\Program Files\Java\jdk-15.0.1\bin\yeni>javac Merhaba.java

C:\Program Files\Java\jdk-15.0.1\bin\yeni>java Merhaba
Merhaba, DÃ¼nya!

C:\Program Files\Java\jdk-15.0.1\bin\yeni>
```

java Merhaba  
Merhaba, DÃ¼nya!

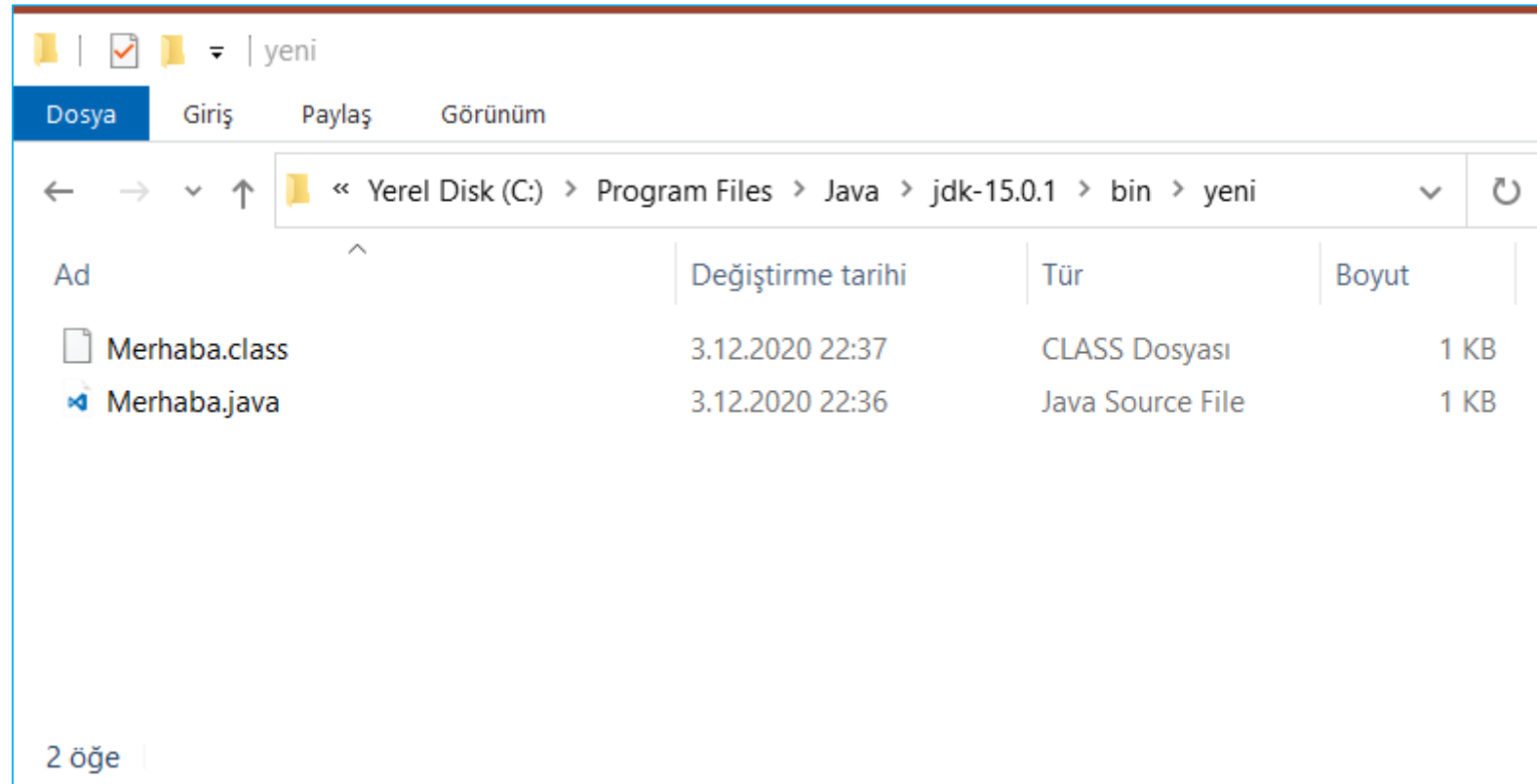
# «yeni» klasörü pencere görünümü

Kaynak dosya:

.java

Derlenmiş dosya:

.class



# Merhaba.class

```
Merhaba.class x Merhaba.java x
1 //yaman akbulut
2 //03 Aralık 2020
3 public class Merhaba{
4     public static void main(String [] args){
5         //Konsol penceresinde selamlama
6
7         System.out.println("Merhaba, Dünya!");
8     }
9 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni\Merhaba.class - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Merhaba.class x Merhaba.java x
1 Êÿ³NULNULNUL; NULGS
2 NULSTXNULETXBELNULEOTFFNULENONULACKSOHNULDLEjava/lang/ObjectSOHNULACK<
init>SOHNULETX()V NULBSNUL BELNUL
3 FFNULVTNULFFSOHNULDLEjava/lang/SystemSOHNULETXoutSOHNULNAKLjava/io/Prin
tStream;BSNULSOHNULDC2Merhaba, DÃfÃnya!
4 NULDLENULDC1BELNULDC2FFNULDC3NULDC4SOHNULDC3java/io/PrintStreamSOHNUL
BELprintlnSOHNULNAK(Ljava/lang/String;)VBELNULSYNSOHNULBELMerhabaSOHNUL
EOTCodeSOHNULSILineNumberTableSOHNULEOTmainSOHNULSYN([Ljava/lang/String;
)VSOHNUL
5 SourceFileSOHNULFFMerhaba.javaNUL!NULNAKNULSTXNULNULNULNULNULSTXNULSOH
NULENONULACKNULSOHNULETBNULNULNULGSNULSOHNULSOHNULNULNULENO*·NULSOH±
NULNULNULSOHNULCANNULNULNULACKNULSOHNULNULNULETXNUL
NULEMNULSUBNULSOHNULETBNULNULNUL%NULSTXNULSOHNULNULNUL²NULBELDC2
6 ¶NULSI±NULNULNULSOHNULCANNULNULNUL
7 NULSTXNULNULNULBELNULBSNULBSNULSOHNULESCNULNULNULSTXNULES
```

# Ödev 1: MerhabaUcMesajli.java

```
1 public class MerhabaUcMesajli {  
2     public static void main(String[] args) {  
3         System.out.println("Programming is fun!");  
4         System.out.println("Fundamentals First");  
5         System.out.println("Problem Driven");  
6     }  
7 }
```

Programming is fun!  
Fundamentals First  
Problem Driven

Kaynak: Introduction to JAVA Programming 10th Edition, Y. Daniel Liang

# Ödev 2: IfadeyiHesapla.java

```
1 public class IfadeyiHesapla {  
2     public static void main(String[] args) {  
3         System.out.println((10.5 + 2 * 3) / (45 - 3.5));  
4     }  
5 }
```

0.39759036144578314

Kaynak: Introduction to JAVA Programming 10th Edition, Y. Daniel Liang

# Ödev 3:

Aşağıdaki kodun çıktısını gösteriniz.

```
public class Test {  
    public static void main(String[] args) {  
        System.out.println("3.5 * 4 / 2 - 2.5 is ");  
        System.out.println(3.5 * 4 / 2 - 2.5);  
    }  
}
```

Kaynak: Introduction to JAVA Programming 10th Edition, Y. Daniel Liang

# Ödev 4:

Aşağıdaki kodun syntax (sözdizimi) hatasının (error) çıktısını gösteriniz.  
Daha sonra kodu düzeltip çalıştırınız.

## LISTING 1.4 ShowSyntaxErrors.java

```
1  public class ShowSyntaxErrors {  
2      public static main(String[] args) {  
3          System.out.println("Welcome to Java");  
4      }  
5  }
```

Kaynak: Introduction to JAVA Programming 10th Edition, Y. Daniel Liang

# Ödev 5:

Aşağıdaki kodun runtime (çalışma zamanı) hatasının (error) çıktısını gösteriniz.  
Daha sonra kodu düzeltip çalıştırınız.

## LISTING 1.5 ShowRuntimeErrors.java

```
1 public class ShowRuntimeErrors {  
2     public static void main(String[] args) {  
3         System.out.println(1 / 0);  
4     }  
5 }
```

# Ödev 6:

Aşağıdaki kodun mantıksal hatasının (error) çıktısını gösteriniz.

Daha sonra kodu düzeltip çalıştırınız. **Not: Doğru cevap 67 değil, 95.**

## LISTING 1.6 ShowLogicErrors.java

```
1 public class ShowLogicErrors {  
2     public static void main(String[] args) {  
3         System.out.println("Celsius 35 is Fahrenheit degree ");  
4         System.out.println((9 / 5) * 35 + 32);  
5     }  
6 }
```

```
Celsius 35 is Fahrenheit degree  
67
```

# Ödev 7:

"Java'ya Hoşgeldiniz!" mesajını 5 defa gösteren bir Java programı yazınız, çalıştırınız.

# Ödev 8:

Aşağıdaki örüntüyü (çıktıyı) veren bir Java programı yazınız.

```
      J      A      V      V      A
      J      A A      V      V      A A
J      J      AAAAA      V V      AAAAA
      J J      A      A      V      A      A
```

```
      J      A      V      V      A
      J      A A      V      V      A A
J      J      AAAAA      V V      AAAAA
      J J      A      A      V      A      A
```

# Ödev 9:

Aşağıdaki koddaki hataları bulunuz ve düzeltiniz.

```
1  public class Welcome {  
2      public void Main(String[] args) {  
3          System.out.println('Welcome to Java!);  
4      }  
5  }
```

# Ödev 10:

Java ve Bilişim terimleri olarak

JAVA SE, JAVA EE, JSP, JSF, JAVA ME, JAVA FX, JRE, JDK, JVM,  
compiler, interpreter, bytecode, class, method, block, bit, byte,  
IDE, API, HTML, XML, RIA, OS, CPU, ALU, RAM, ROM, EPROM, EEPROM,  
LAN, NIC, DSL, USB, CD, DVD, KB, MB, GB, TB,

nedir?

İngilizce ve Türkçe karşılıklarını yazınız.

# Reserved Words / Keywords

- class
- public
- private
- static
- void
- main
- int
- float
- char
- boolean
- byte
- double

# JAVA değişkenler, değişken tanımlama

- Bir Java programı yürütülürken değer tutan saklayıcılara değişken (variable) denir.
- Değişkenler bir veri tipi (datatype) ile atanmaktadırlar.

# JAVA Değişken Tipleri

- Yerel Değişkenler (Local Variables)
- Örnek Değişkenler (Instance Variables)
- Statik Değişkenler (Static Variables)

# 1-Yerel Değişkenler

- Metot gövdesi içinde tanımlanan bir değişken yerel (local) değişkendir.
- Bu değişkenleri sadece metot ile kullanabiliriz.
- Bir yerel değişken «static» anahtar sözcüğü ile tanımlanamaz.

## 2-Örnek Değişkenler

- Sınıf içerisinde ancak metod gövdesi dışında tanımlanan bir değişkendir.
- Bir örnek değişken «static» anahtar sözcüğü ile tanımlanamaz.

# 3-Statik Değişkenler

- «static» anahtar sözcüğü ile tanımlanan değişkenlerdir.
- Bu değişkenler yerel olmaz.
- Statik değişkenin tek bir kopyası oluşturulur ve sınıfın tüm örnekleri arasında paylaşılır.
- Statik değişkenler için bellek tahsisi sadece sınıf bellekte yüklendiği zaman gerçekleştirilir.

# Değişken tipleri

1. `class` `Abc`{
2.     `int` `sayi`=10; //örnek değişken (instance variable)
3.     `static int` `m`=100;//statik değişken (static variable)
4.     `void` `birinciMethod`(){\
5.         `int` `k`=50;// yerel değişken (local variable)
6.     }`//metotun sonu` (end of the method)
7. `}// sınıfın sonu` (end of the class)

# Değişkenler için kod örnekleri-1

```
1 class Temel{  
2     public static void main(String[] args){  
3         int a = 10;  
4         int b = 20;  
5         int c = a + b;  
6         System.out.println(c);  
7     }  
8 }
```

Program çıktısı:  
30

# Değişkenler için kod örnekleri-2 (Genişletme)

```
1 class Temel{
2     public static void main(String[] args){
3         int a = 20;
4         float b = a;
5         System.out.println(a);
6         System.out.println(b);
7     }
8 }
```

Program çıktısı:

20

20.0

# Değişkenler için kod örnekleri-3 (Tip dönüşümü)

```
1 class Temel{
2     public static void main(String[] args){
3         float b = 20.5f;
4         //int a=b; //derleme zamanı hatası
5         int a = (int)b;
6         System.out.println(b);
7         System.out.println(a);
8     }
9 }
```

Program çıktısı:

20.5

20

# Değişkenler için kod örnekleri-4 (Taşma)

```
1 class Temel{
2     public static void main(String[] args){
3         //Overflow
4         int a = 150;
5         byte b = (byte)a;
6         System.out.println(a);
7         System.out.println(b);
8     }
9 }
```

Program çıktısı:

150

-106

# JAVA Veri Tipleri

- İlkel Veri Tipleri (Primitive Data Types)
- İlkel Olmayan Veri Tipleri (Non-Primitive Data Types)

# 8. Ders: JAVA Programlama Dili (Veri Tipleri, Tanımlayıcılar, Operatörler)

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü

YMH111 Algoritma ve Programlama-I

Doç. Dr. Yaman Akbulut

# JAVA Programlama Dili

- <http://www.kriptarium.com/algorithm.html> (Yardımcı kaynak)
- JAVA Programlama Dili (Ders: 11-22) video. (Mutlaka İzleyiniz!)
  - Ders 11: JAVA programlarının derlenmesi ve yorumlanması (izle)
  - Ders 12: JVM, JRE ve JDK arasındaki temel farklar (izle)
  - Ders 13: Java Değişkenler (izle)
  - Ders 14: Değişkenler ile alakalı kod örnekleri (izle)
  - Ders 15: Veri Tipleri (izle)
  - Ders 16: Java İlkel/Primitive Veri Tipleri (izle)
  - Ders 17: Değişken Aralıkları (izle)
  - Ders 18: Java Operatörler (izle)
  - Ders 19: Operatör Örnekleri 1 (x++ ve ++x) (izle)
  - Ders 20: Operatör Örnekleri 2 (izle)
  - Ders 21: Mantıksal ve Aritmetiksel Operatörler (izle)
  - Ders 22: Kaydırma Operatörleri (izle)

# JAVA'da program yazma ve çalıştırma ortamı

1. Java Development Kit (JDK) (Java geliştirme kiti (seti))  
+ Windows notepad (yazı editörü) (alternatif [notepad++](#))

2. Integrated Development Environment (IDE)

Entegre geliştirme ortamı

- NetBeans
- Eclipse
- IntelliJ IDEA

# Özel Karakterler

{ } braces

( ) parentheses

[ ] brackets

// double slashes

" " quotation marks

; semicolon

/\*

for block comments

\*/

/\*\*

for documentation

\*/

# Java Keywords /Reserved Words

|                 |                   |                  |                     |
|-----------------|-------------------|------------------|---------------------|
| <b>abstract</b> | <b>double</b>     | <b>int</b>       | <b>super</b>        |
| <b>assert</b>   | <b>else</b>       | <b>interface</b> | <b>switch</b>       |
| <b>boolean</b>  | <b>enum</b>       | <b>long</b>      | <b>synchronized</b> |
| <b>break</b>    | <b>extends</b>    | <b>native</b>    | <b>this</b>         |
| <b>byte</b>     | <b>final</b>      | <b>new</b>       | <b>throw</b>        |
| <b>case</b>     | <b>finally</b>    | <b>package</b>   | <b>throws</b>       |
| <b>catch</b>    | <b>float</b>      | <b>private</b>   | <b>transient</b>    |
| <b>char</b>     | <b>for</b>        | <b>protected</b> | <b>try</b>          |
| <b>class</b>    | <b>goto</b>       | <b>public</b>    | <b>void</b>         |
| <b>const</b>    | <b>if</b>         | <b>return</b>    | <b>volatile</b>     |
| <b>continue</b> | <b>implements</b> | <b>short</b>     | <b>while</b>        |
| <b>default</b>  | <b>import</b>     | <b>static</b>    |                     |
| <b>do</b>       | <b>instanceof</b> | <b>strictfp*</b> |                     |

# Java Veri Tipleri

## Primitive data types:

- Boolean
- Character
- Integer
- Floating point

## Non-Primitive data types:

- Classes
- Interfaces
- Arrays

# Numerik Veri Tipleri

| tipi  | kapasitesi              | değer aralığı            |
|-------|-------------------------|--------------------------|
| byte  | 8-bit işaretli tamsayı  | $-2^7 \dots 2^7-1$       |
| short | 16-bit işaretli tamsayı | $-2^{15} \dots 2^{15}-1$ |
| int   | 32-bit işaretli tamsayı | $-2^{31} \dots 2^{31}-1$ |
| long  | 64-bit işaretli tamsayı | $-2^{63} \dots 2^{63}-1$ |

**float**                      32-bit sayı ondalıklı sayı (kesir)                      32-bit IEEE 754  
Negatif aralık:  $-3.4028235E + 38$  den  $-1.4E - 45$   
Pozitif aralık:  $1.4E - 45$  ten  $3.4028235E + 38$

**double**                      64-bit sayı ondalıklı sayı (kesir)                      64-bit IEEE 754  
Negatif aralık:  $-1.7976931348623157E + 308$  ten  $-4.9E - 324$   
Pozitif aralık :  $4.9E - 324$  ten  $1.7976931348623157E + 308$

# Klavyeden numerik değer okuma

| tipi     | metot        | açıklama                            |
|----------|--------------|-------------------------------------|
| • byte   | nextByte()   | byte tipinde tamsayı okur.          |
| • short  | nextShort()  | short tipinde tamsayı okur.         |
| • int    | nextInt()    | int tipinde tamsayı okur.           |
| • long   | nextLong()   | long tipinde tamsayı okur.          |
| • float  | nextFloat()  | float tipinde ondalıklı sayı okur.  |
| • double | nextDouble() | double tipinde ondalıklı sayı okur. |

# Örnek 1: SayiOkuma.java

```
1  //Nümerik Sayı Okuma
2  //
3  import java.util.Scanner;
4
5  public class SayiOkuma {
6      public static void main(String [] args) {
7          Scanner giriş = new Scanner(System.in);
8          System.out.print("Bir byte değeri giriniz:");
9          byte byteDeger = input.nextByte();
10
11          System.out.print("Bir short değeri giriniz:");
12          short shortDeger = input.nextShort();
13
14          System.out.print("Bir int değeri giriniz:");
15          int intDeger = input.nextInt();
16
17          System.out.print("Bir long değeri giriniz:");
18          long longDeger = input.nextLong();
19
20          System.out.print("Bir float değeri giriniz:");
21          float floatDeger = input.nextFloat();
22
23          System.out.print("Bir double değeri giriniz:");
24          double doubleDeger = input.nextDouble();
25      }
26 }
```

# Tanımlayıcılar

## Tanımlayıcılar

Classes (Sınıflar),  
Methods (Yöntemler),  
ve Variables (Değişkenleri) tanımlar.

Bir tanımlayıcı Harfler, rakamlar, \_ alt çizgi ve \$ işaretlerinden oluşabilir.

Bir tanımlayıcı rakam ile başlayamaz.

Bir tanımlayıcı keywords'den oluşamaz.

Bir tanımlayıcı true, false veya null olamaz.

Bir tanımlayıcı istenen uzunlukta olabilir.

# Tanımlayıcılar

Java büyük küçük harfe duyarlıdır.

yaman, Yaman, YAMAN farklı tanımlayıcıdır.

## Açıklayıcı isimlendirme

Tanımlayıcılar program içindeki değişkenlerin, metotların, sınıfların ve diğer şeylerin isimlendirilmesi için kullanılır.

İsimlendirmenin açıklayıcı olması program okunabilirliği açısından çok önemlidir. `ogrenciSayisi`; `ogrSay`, `ogrSayi`, `ogrenciSayi` dan daha iyi bir isimlendirmedir.

İsimlendirmelerde kısaltmalardan kaçınınız.

`i`, `j`, `k`, `x`, `y` benzeri değişken isimlendirmeleri genelleme açısından uygundur. Bunları çoğunlukla indis olarak kullanıyoruz.

\$ karakterini isimlendirmede kullanmayınız.

# Tanımlayıcıları isimlendirme

## Değişkenlerin ve yöntemlerin (metot) isimlendirilmesi

Değişken (variable) ve yöntemleri (method) isimlendirmeye küçük harf ile başlayın, küçük devam edin.

`yaricap` değişken, `alan` değişken, `print` yöntem

## Sınıfların isimlendirilmesi

Sınıfların (class) ilk harflerini Büyük yapın.

`AlanHesapla` sınıf, `System` sınıf

`System` sınıfı Java'da tanımlı olduğundan kendi sınıflarınıza bu ismi vermemelisiniz!

## Sabitlerin isimlendirilmesi

Sabitlerin (constant) bütün harflerini BÜYÜK yapın.

`PI` sabit, `ENBUYUK` sabit

# Değişken tanımlama

Veri tipleri: `int` `double` `byte` `short` `long` `float` `char` `boolean`

## Örnek tanımlamalar

```
veritipi degisken1, degisken2, ..., degiskenn;
```

```
int sayi;  
double yaricap;  
double donusumOrani;
```

```
int i, j, k;
```

```
int sayi = 1;
```

```
int sayi;  
sayi = 1;
```

```
int i = 1, j = 2;
```

# Atama Operatörü (=)

## Örnek atamalar

```
int y = 1;  
double yarıcap = 2.0;  
int x = 5 * (3 / 2);  
x = y + 1;  
double alan = yarıcap * yarıcap * 3.14159;
```

```
x = x + 1;
```

```
1 = x; //yanlış kullanım
```

degisken = ifade şeklinde kullanılır.

$x = 2 * x + 1$

matematikte bir denklem

$x = 2 * x + 1$

Java'da  $2 * x + 1$  ifadesi  $x$  değişkenine atanır.

# Sabit Tanımlama (final)

## LISTING 2.4 ComputeAreaWithConstant.java

```
1  import java.util.Scanner; // Scanner is in the java.util package
2
3  public class ComputeAreaWithConstant {
4      public static void main(String[] args) {
5          final double PI = 3.14159; // Declare a constant
6
7          // Create a Scanner object
8          Scanner input = new Scanner(System.in);
9
10         // Prompt the user to enter a radius
11         System.out.print("Enter a number for radius: ");
12         double radius = input.nextDouble();
13
14         // Compute area
15         double area = radius * radius * PI;
16
17         // Display result
18         System.out.println("The area for the circle of radius " +
19             radius + " is " + area);
20     }
21 }
```

# Aritmetik Operatörler

|   | anlamı      | örnek        | sonuç |
|---|-------------|--------------|-------|
| + | toplama     | $22 + 1$     | 23    |
| - | çıkarma     | $22.0 - 0.1$ | 21.9  |
| * | çarpma      | $200 * 30$   | 6000  |
| / | bölme       | $1.0 / 2.0$  | 0.5   |
| % | mod (kalan) | $20 \% 3$    | 2     |

Hem bölünen hem de bölen `int` ise:

5 / 2 sonuç 2 olur

5 / 2 sonuç 2.5 olmaz

% Mod işlemi kalanı verir:

5 % 2 sonuç 1 olur

10 % 2 sonuç 0 olur

20 % 3 sonuç 2 olur

# Artı (+) Operatörü

```
ArtiOperatoru.java
1  //Artı (+) operatörü
2
3  public class ArtiOperatoru {
4      public static void main(String [] args) {
5          byte top;
6          top = 3 + 5;
7          System.out.println("3+5=" + top);
8          System.out.println("Yazilim Muhedisligi Bolumu");
9          System.out.println("Yazilim " + "Muhedisligi " + "Bolumu");
10     }
11 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>javac ArtiOperatoru.java
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java ArtiOperatoru
```

```
3+5=8
```

```
Yazilim Muhedisligi Bolumu
```

```
Yazilim Muhedisligi Bolumu
```

# Ödev 1:

```
OrtalamaHesapla.java
1 //Ortalama Hesapla
2 import java.util.Scanner; //Scanner, java.util paketinde
3
4 public class OrtalamaHesapla {
5     public static void main(String [] args) {
6         //Scanner nesnesi oluşturur
7         Scanner input = new Scanner(System.in);
8
9         //Kullanıcıdan 3 sayı girmesi istenir
10        System.out.print("Üc sayi giriniz:");
11        double sayi1 = input.nextDouble();
12        double sayi2 = input.nextDouble();
13        double sayi3 = input.nextDouble();
14
15        //Ortalama hesaplanır
16        double ortalama = (sayi1 + sayi2 + sayi3) / 3;
17
18        //Sonuç gösterilir
19        System.out.println(sayi1 + " " + sayi2 + " "
20            + sayi3 + " " + "Ortalaması: " + ortalama);
21    }
22 }
```

## Ödev 2: `KardesYasi.java`

İki kardeşe ait yaş bilgisinin kullanıcı tarafından klavyeden girilmesi istenmiştir ve sonrasında kardeşlerin yaşlarının toplamı ve ortalaması hesaplanıp, sonucu çıktı alınmıştır.

Bu kodu Java programlama dilinde yazınız.

# Ödev 3:

Aşağıdaki algoritmayı Java koduna dönüştürünüz (yazınız)

Adım 1: Başlangıç değeri 100 olan `mil` isimli bir `double` değişken tanımlayınız.

Adım 2: Değeri 1.609 olan `KILOMETRE_MIL_KATSAYISI` isimli bir `double` sabit tanımlayınız.

Adım 3: `kilometre` isimli bir `double` değişken tanımlayınız, `mil` ve `KILOMETRE_MIL_KATSAYISI` nı çarpınız ve sonucu `kilometre`ye atayınız.

Adım 4: `kilometre`yi konsolda gösteriniz.

Adım 4'ten sonra `kilometre` nedir?

# Ödev 4:

Aşağıdaki kodu bir kaynak dosyasına yazınız. Çalışıp değer döndürebilmesi için ana metodu ve diğer gereken kodları ekleyiniz. Dosyanın ismi Kuvvet.java olsun.

`pow` metodu `Math` sınıfında tanımlanmıştır. Syntax (sözdizimi) `Math.pow(a, b)` şeklindedir. `a` üssü `b`'nin sonucunu geri döndürür. Örnek olarak `Math.pow(2, 3)` ifadesinde  $a^b$  ( $2^3$ ) sonuç 8 olarak gelir, burada `a` ve `b`; `pow` metodunun parametresidir.

```
System.out.println(Math.pow(2, 3)); // Displays 8.0
System.out.println(Math.pow(4, 0.5)); // Displays 2.0
System.out.println(Math.pow(2.5, 2)); // Displays 6.25
System.out.println(Math.pow(2.5, -2)); // Displays 0.16
```

Kaynak: Introduction to JAVA Programming 10th Edition, Y. Daniel Liang

# Ödev 5:

Aşağıdaki kod satırlarının bir dosyaya yazıp çalıştırınız. Dosyanın adı NoktaliSayi.java olsun.

Birinci satırın sonucu ile ikinci satırın sonucunu inceleyiniz, karşılaştırınız.

```
System.out.println("1.0 / 3.0 = " + 1.0 / 3.0);
```

```
System.out.println("1.0F / 3.0F = " + 1.0F / 3.0F);
```

# Ödev 6:

Java ve Bilişim terimleri olarak

multiprogramming, multithreading, multiprocessing,

runtime error, logic error, compile error,

pseudocode, variable, console,

identifier, suffix, prefix,

nedir?

İngilizce ve Türkçe karşılıklarını yazınız.

# 9. Ders: JAVA Programlama Dili (keywords, operatörler, kontrol yapıları)

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü

YMH111 Algoritma ve Programlama-I

Doç. Dr. Yaman Akbulut

# JAVA Programlama Dili

- <http://www.kriptarium.com/algorithm.html> (Yardımcı kaynak)
- JAVA Programlama Dili (Ders: 18-32) video. (Mutlaka İzleyiniz!)
  - Ders 18: Java Operatörler (izle)
  - Ders 19: Operatör Örnekleri 1 (x++ ve ++x) (izle)
  - Ders 20: Operatör Örnekleri 2 (izle)
  - Ders 21: Mantıksal ve Aritmetiksel Operatörler (izle)
  - Ders 22: Kaydırma Operatörleri (izle)
  - Ders 23: Koşul Kontrol ? Operatörleri (izle)
  - Ders 24: Kontrol Yapılarına Giriş (izle)
  - Ders 25: Kontrol Yapıları (if/else) İfadesinin Kullanımı (izle)
  - Ders 26: Kontrol Yapıları - Örnek Problem 1 (izle)
  - Ders 27: Kontrol Yapıları - Örnek Problem 2 (izle)
  - Ders 28: Kontrol Yapıları - Örnek Problem 3 (izle)
  - Ders 29: Kontrol Yapıları - İç içe Koşul Yapıları Oluşturmak (izle)
  - Ders 30: Kontrol Yapıları - Örnek Problem 4 (izle)
  - Ders 31: Kontrol Yapıları - ? operatörünün kullanımı (izle)
  - Ders 32: Kontrol Yapıları - Switch Kullanımı (izle)

# JAVA'da program yazma ve çalıştırma ortamı

1. JDK + `notepad++`

2. Integrated Development Environment (IDE)

Entegre geliştirme ortamı

- NetBeans
- Eclipse
- IntelliJ IDEA

# Özel Karakterler

{ } braces

( ) parentheses

[ ] brackets

// double slashes

" " quotation marks

; semicolon

/\*

for block comments

\*/

/\*\*

for documentation

\*/

# Java Keywords

|          |            |           |              |
|----------|------------|-----------|--------------|
| abstract | double     | int       | super        |
| assert   | else       | interface | switch       |
| boolean  | enum       | long      | synchronized |
| break    | extends    | native    | this         |
| byte     | final      | new       | throw        |
| case     | finally    | package   | throws       |
| catch    | float      | private   | transient    |
| char     | for        | protected | try          |
| class    | goto       | public    | void         |
| const    | if         | return    | volatile     |
| continue | implements | short     | while        |
| default  | import     | static    |              |
| do       | instanceof | strictfp  |              |

# Artırılmış Atama Operatörleri

| operatör        | anlamı      | örnek               | eşdeğeri               |
|-----------------|-------------|---------------------|------------------------|
| <code>+=</code> | toplama     | <code>i += 8</code> | <code>i = i + 8</code> |
| <code>-=</code> | çıkarma     | <code>i -= 8</code> | <code>i = i - 8</code> |
| <code>*=</code> | çarpma      | <code>i *= 8</code> | <code>i = i * 8</code> |
| <code>/=</code> | bölme       | <code>i /= 8</code> | <code>i = i / 8</code> |
| <code>%=</code> | mod (kalan) | <code>i %= 8</code> | <code>i = i % 8</code> |

```
sayi = sayi + 1;  
Şununla aynı  
sayi += 1;
```

# Artırılmış Atama Operatörleri

```
x /= 4 + 5.5 * 1.5;  
Şununla aynı  
x = x / (4 + 5.5 * 1.5);
```

```
x += 2; // ... (statement)  
System.out.println(x += 2); // ifade (expression)  
  
// += -= *= /= %= hem statement içinde hem de ifade  
içinde olarak kullanılabilir.
```

# Örnek 1:

```
ArtirilmisOperator.java
1  //Artirilmis Operatör
2  //
3  public class ArtirilmisOperator {
4      public static void main(String [] args) {
5          double a = 6.5;
6          a += a + 1;
7          System.out.println(a);
8          a = 6;
9          a /= 2;
10         System.out.println(a);
11     }
12 }
```

Komut İstemi

```
C:\Program Files\Java\jdk-16.0.1\bin\yeni>java ArtirilmisOperator
14.0
3.0
```

# Artırma (Inc.) Azaltma (Dec.) Operatörleri

| operatör | anlamı  | İng.      | eşdeğeri            |
|----------|---------|-----------|---------------------|
| ++       | artırma | increment | değişkeni 1 artırır |
| --       | azaltma | decrement | değişkeni 1 azaltır |

i++; // i artı artı şeklinde okunur, ++ yerine İng. plus plus  
j--; // j eksi eksi şeklinde okunur, -- yerine İng. minus minus

# Artırma (Inc.) Azaltma (Dec.) Operatörleri

```
int i = 3, j = 3;  
i++; // i 4 olur  
j--; // j 2 olur
```

++ postfix increment, postincrement  
-- postfix decrement, postdecrement

```
int i = 3, j = 3;  
++i; // i 4 olur  
--j; // j 2 olur
```

++ prefix increment, preincrement  
-- prefix decrement, predecrement

# Artırma (Inc.) Azaltma (Dec.) Operatörleri

| Operatör           | ismi          | açıklama                                         | örnek ( i = 1 kabul edilsin)                              |
|--------------------|---------------|--------------------------------------------------|-----------------------------------------------------------|
| <code>++var</code> | preincrement  | değişkeni 1 artırır,<br>yeni değerini kullanır   | <code>int j = ++i;</code><br><code>// j = 2, i = 2</code> |
| <code>var++</code> | postincrement | değişkeni 1 artırır,<br>önceki değerini kullanır | <code>int j = i++;</code><br><code>// j = 1, i = 2</code> |
| <code>--var</code> | predecrement  | değişkeni 1 azaltır,<br>yeni değerini kullanır   | <code>int j = --i;</code><br><code>// j = 0, i = 0</code> |
| <code>var--</code> | postdecrement | değişkeni 1 azaltır,<br>önceki değerini kullanır | <code>int j = i--;</code><br><code>// j = 1, i = 0</code> |

## Örnek 2:

```
int i = 10;  
int yeniSayi = 10 * i++;  
System.out.print(" i: " + i + ", yeni sayı: " + yeniSayi);
```

i: 11, yeni sayı: 100 //sonuç

```
int yeniSayi = 10 * i;  
i = i + 1;
```

```
int i = 10;  
int yeniSayi = 10 * (++i);  
System.out.print(" i: " + i + ", yeni sayı: " + yeniSayi);
```

i: 11, yeni sayı: 110 //sonuç

```
i = i + 1;  
int yeniSayi = 10 * i;
```

## Örnek 3:

```
1  //Artırılmış Operatör
2  //
3  public class ArtirilmisOperator2 {
4      public static void main(String [] args) {
5          int a = 6;
6          int b = a++;
7          System.out.println(a);
8          System.out.println(b);
9          a = 6;
10         b = ++a;
11         System.out.println(a);
12         System.out.println(b);
13     }
14 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java ArtirilmisOperator2
```

```
7
```

```
6
```

```
7
```

```
7
```

# boolean veri tipi

`boolean` veri tipi `true` ya da `false` değerlerini alan bir değişkeni tanımlar.

Herhangi iki değeri karşılaştırmanın sonucu ya `true` ya da `false` şeklinde bir sonuç verir.

# Karşılaştırma Operatörleri

| Java      | Mat.    | ismi                      | örnek                  | sonuç        |
|-----------|---------|---------------------------|------------------------|--------------|
| Operatörü | sembolü |                           | (yaricap 5 olsun)      | (boolean)    |
| <         | <       | küçük mü                  | <b>yaricap &lt; 0</b>  | <b>false</b> |
| <=        | ≤       | küçük eşit mi             | <b>yaricap &lt;= 0</b> | <b>false</b> |
| >         | >       | büyük mü                  | <b>yaricap &gt; 0</b>  | <b>true</b>  |
| >=        | ≥       | büyük eşit mi             | <b>yaricap &gt;= 0</b> | <b>true</b>  |
| ==        | =       | eşit mi                   | <b>yaricap == 0</b>    | <b>false</b> |
| !=        | ≠       | eşit değil mi (farklı mı) | <b>yaricap != 0</b>    | <b>true</b>  |

```
double yaricap = 1;  
System.out.println(yaricap > 1);  
// true false
```

## Örnek 4:

```
1 //Toplama Testi
2 //
3 import java.util.Scanner;
4
5 public class ToplamaTesti {
6     public static void main(String [] args) {
7         int sayi1 = (int) (System.currentTimeMillis() % 10);
8         int sayi2 = (int) (System.currentTimeMillis() / 7 % 10);
9
10        // Scanner oluşturma (klavyeden/konsoldan giriş için)
11        Scanner giris = new Scanner(System.in);
12
13        System.out.print(sayi1 + " + " + sayi2 + " sonucu nedir? ");
14
15        int cevap = giris.nextInt();
16
17        System.out.println(sayi1 + " + " + sayi2 + " = " + cevap
18        + " sonucu " + (sayi1 + sayi2 == cevap));
19    }
20 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java ToplamaTesti
3 + 3 sonucu nedir? 5
3 + 3 = 5 sonucu false
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java ToplamaTesti
2 + 8 sonucu nedir? 10
2 + 8 = 10 sonucu true
```

# Random sayı üretme

Random sayı üretmenin birçok farklı yolu vardır. Örnekte random sayı üretmek için kullanılan yöntemi inceleyiniz:

Bu vb. satırları bir Java kodu aracılığıyla konsolda gösteriniz.

```
(System.currentTimeMillis())
```

```
(System.currentTimeMillis() % 10)
```

```
(int)(System.currentTimeMillis() % 10)
```

```
(System.currentTimeMillis()/7 % 10)
```

```
(int)(System.currentTimeMillis()/7 % 10)
```

# Java Karar yapıları

1. if yapısı
2. if else yapısı
3. if  
else if  
else if  
...  
else yapısı
4. switch yapısı
5. Koşullu ifade yapısı

# Karar yapıları 1: if

```
if (boolean ifade/şart/koşul) {  
  
    // şart doğruysa (true)  
    // yapılacaklar 1  
    // yapılacaklar 2  
    // yapılacaklar 3 ...  
}  
// şart yanlışsa (false) buradan devam edecek...
```

# Karar yapıları 1: if

```
if i > 0 {  
System.out.println("i pozitif");  
} //yanlış kullanım
```

```
if (i > 0) {  
System.out.println("i pozitif");  
} //genel kullanım
```

```
if (cift == true)  
System.out.println("çift");
```

```
if (i > 0) {  
System.out.println("i pozitif");  
} //doğru kullanım
```

≡

```
if (i > 0)  
System.out.println("i pozitif");  
// if ten sonra tek satır varsa
```

≡

```
if (cift)  
System.out.println("çift");  
// daha iyi
```

# Örnek 5: if yapısı

```
1 //Temel if Demo
2 //
3 import java.util.Scanner;
4
5 public class TemelIfDemo {
6     public static void main(String [] args) {
7         Scanner giris = new Scanner(System.in);
8         System.out.print("Bir tam sayi giriniz:");
9         int sayi = giris.nextInt();
10
11         if (sayi % 5 == 0)
12             System.out.println("HiFive");
13
14         if (sayi % 2 == 0)
15             System.out.println("HiEven");
16     }
17 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java TemelIfDemo
```

```
Bir tam sayi giriniz:8
```

```
HiEven
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java TemelIfDemo
```

```
Bir tam sayi giriniz:40
```

```
HiFive
```

```
HiEven
```

# Karar yapıları 2: if-else

```
if (boolean ifade/şart/koşul) {  
    // şart  
    // doğruysa (true)  
    // yapılacaklar...  
}  
else {  
    // şart  
    // yanlışsa (false)  
    // yapılacaklar...  
}
```

# Karar yapıları 2: if else

```
1  //AlanHesapla
2  //
3  import java.util.Scanner;
4
5  public class AlanHesapla{
6      public static void main(String [] args){
7          Scanner giris = new Scanner(System.in);
8
9          System.out.print("Lutfen capi giriniz: ");
10         int cap = giris.nextInt();
11
12         if (cap<0){
13             System.out.println("Yanlış giriş!");
14         }
15         else {
16             double alan = cap * cap * 3.14159;
17             System.out.println("Alan: " + alan);
18         }
19     }
20 }
```

## Karar yapıları 2: if else

```
if (radius >= 0) {  
    area = radius * radius * PI;  
    System.out.println("The area for the circle of radius " +  
        radius + " is " + area);  
}  
else {  
    System.out.println("Negative input");  
}
```

# Karar yapıları 2: **if else**

```
if (number % 2 == 0)
    System.out.println(number + " is even.");
else
    System.out.println(number + " is odd.");
```

# Karar yapıları 2: if else

```
if (age < 16)
    System.out.println
        ("Cannot get a driver's license");
if (age >= 16)
    System.out.println
        ("Can get a driver's license");
```

(a)

```
if (age < 16)
    System.out.println
        ("Cannot get a driver's license");
else
    System.out.println
        ("Can get a driver's license");
```

(b)

# Karar yapıları 3: if else if ... else

```
if (koşul 1) {  
    // koşul 1 doğruysa (true) yapılacaklar...  
}  
else if (koşul 2){  
    // koşul 2 doğruysa (true) yapılacaklar...  
}  
else if (koşul 3){  
    // koşul 3 doğruysa (true) yapılacaklar...  
}  
...  
else {  
    // koşullar sağlanmazsa...  
}
```

# Karar yapıları 3: if else if ... else

```
if (puan >= 90.0)
    System.out.print("A");
else
    if (puan >= 80.0)
        System.out.print("B");
    else
        if (puan >= 70.0)
            System.out.print("C");
        else
            if (puan >= 60.0)
                System.out.print("D");
            else
                System.out.print("F");
```

≡

```
if (puan >= 90.0)
    System.out.print("A");
else if (puan >= 80.0)
    System.out.print("B");
else if (puan >= 70.0)
    System.out.print("C");
else if (puan >= 60.0)
    System.out.print("D");
else
    System.out.print("F");

// daha iyi
```

# İç içe if yapıları

```
int i = 1, j = 2, k = 3;

if (i > j) {
    if (i > k)
        System.out.println("A");
}
else
    System.out.print("B");
```

# Örnek 6: Random sayı üretme (CikarmaTesti)

```
1 //Cikarma Testi
2 import java.util.Scanner;
3
4 public class CikarmaTesti {
5     public static void main(String [] args) {
6         //1.İki tane random tam sayı üret
7         int sayi1 = (int)(Math.random() * 10);
8         int sayi2 = (int)(Math.random() * 10);
9
10        //2.sayi1<sayi ise sayi1 ve sayi2 yi yer
11        //değiştir
12        if (sayi1 < sayi2) {
13            int temp = sayi1;
14            sayi1 = sayi2;
15            sayi2 = temp;
16        }
```

```
17
18        //3.Kullanıcıya sor "sayi1 - sayi2 sonucu
19        //nedir?"
20        System.out.print(sayi1 + " - " + sayi2
21        + " nedir? ");
22
23        Scanner giris = new Scanner(System.in);
24        int cevap = giris.nextInt();
25
26        // 4. Cevabı değerlendir ve sonucu göster
27        if (sayi1 - sayi2 == cevap)
28            System.out.println("Sonuc dogru, tebrikler!");
29        else {
30            System.out.println("Cevabin yanlis!");
31            System.out.println(sayi1 + " - " + sayi2 +
32            " sonucu " + (sayi1 - sayi2) + " olmalı... ");
33        }
34    }
35 }
```

# Mantık(sal) Operatörleri

| Operatör | İsim (İng.)  | İsim      |
|----------|--------------|-----------|
| !        | not          | değil     |
| &&       | and          | ve        |
|          | or           | veya      |
| ^        | exclusive or | özel veya |

# Örnek 7:

```
1 //Mantik Operatoru
2
3 import java.util.Scanner;
4
5 public class MantikOperatoru {
6     public static void main(String [] args) {
7         //tarayıcı nesnesi oluştur
8         Scanner giris = new Scanner(System.in);
9
10        //giriş al
11        System.out.print("Bir tam sayi giriniz: ");
12        int sayi = giris.nextInt();
13
14        if (sayi % 2 == 0 && sayi % 3 == 0)
15            System.out.println(sayi + " sayisi 2 VE 3'e bolunebilir.");
16
17        if (sayi % 2 == 0 || sayi % 3 == 0)
18            System.out.println(sayi + " sayisi 2 VEYA 3'e bolunebilir.");
19
20        if (sayi % 2 == 0 ^ sayi % 3 == 0)
21            System.out.println(sayi + " sayisi 2 VEYA 3'e bolunebilir, "
22                               + "her ikisine bolunemez.");
23    }
24 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java MantikOperatoru
```

```
Bir tam sayi giriniz: 8
```

```
8 sayisi 2 VEYA 3'e bolunebilir.
```

```
8 sayisi 2 VEYA 3'e bolunebilir, her ikisine bolunemez.
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java MantikOperatoru
```

```
Bir tam sayi giriniz: 18
```

```
18 sayisi 2 VE 3'e bolunebilir.
```

```
18 sayisi 2 VEYA 3'e bolunebilir.
```

# Karar yapıları 4: `switch` yapısı

```
switch (durum) {  
    case deger0: ifade0;  
                break;  
    case deger1 : ifade1;  
                break;  
    case deger2: ifade2;  
                break;  
  
    ...  
    case degerN: ifadeN;  
                break;  
    default: default için ifade;  
            System.exit(1);  
}
```

# Karar yapıları 4: `switch` yapısı

```
switch (gun) {  
    case 0: System.out.println("Pazartesi");  
        break;  
    case 1: System.out.println("Sali");  
        break;  
    case 2: System.out.println("Carsamba");  
        break;  
    case 3: System.out.println("Persembe");  
        break;  
    case 4: System.out.println("Cuma");  
        break;  
    case 5: System.out.println("Cumartesi");  
        break;  
    case 6: System.out.println("Pazar");  
        break;  
    default: System.out.println("Lutfen geçerli bir sayi giriniz");  
        System.exit(1);  
}
```

# Örnek 8:

```
1 //Gunler
2 import java.util.Scanner;
3
4 public class Gunler {
5     public static void main(String [] args) {
6         Scanner giris = new Scanner(System.in);
7
8         System.out.print("Gun icin bir sayi giriniz (1-7): ");
9         int gun = giris.nextInt();
10
11         switch (gun) {
12             case 1: System.out.println("Bugun gunlerden Pazartesi");
13                     break;
14             case 2: System.out.println("Bugun gunlerden Sali");
15                     break;
16             case 3: System.out.println("Bugun gunlerden Carsamba");
17                     break;
18             case 4: System.out.println("Bugun gunlerden Persembe");
19                     break;
20             case 5: System.out.println("Bugun gunlerden Cuma");
21                     break;
22             case 6: System.out.println("Bugun gunlerden Cumartesi");
23                     break;
24             case 7: System.out.println("Bugun gunlerden Pazar");
25                     break;
26             default: System.out.println("Lutfen geçerli bir sayi giriniz");
27                     System.exit(1);
28         }
29     }
30 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java Gunler
Gun icin bir sayi giriniz (1-7): 5
Bugun gunlerden Cuma
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java Gunler
Gun icin bir sayi giriniz (1-7): 8
Lutfen geçerli bir sayi giriniz
```

## Örnek 9:

```
1 //Hafta İçi Sonu
2 import java.util.Scanner;
3
4 public class HaftaIciSonu {
5     public static void main(String [] args) {
6         Scanner giris = new Scanner(System.in);
7
8         System.out.print("Gun icin bir sayi giriniz (1-7): ");
9         int gun = giris.nextInt();
10
11         switch (gun) {
12             case 1:
13             case 2:
14             case 3:
15             case 4:
16             case 5: System.out.println("Hafta ici"); break;
17             case 6:
18             case 7: System.out.println("Hafta sonu");
19         }
20     }
21 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java HaftaIciSonu
Gun icin bir sayi giriniz (1-7): 3
Hafta ici
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java HaftaIciSonu
Gun icin bir sayi giriniz (1-7): 6
Hafta sonu
```

# Karar yapıları 5: Koşullu ifade

şart  
( mantıksal ifade ) ? ifade1 : ifade2;  
koşul

```
if (x > 0)  
    y = 1;  
else  
    y = -1;
```

$y = (x > 0) ? 1 : -1;$

## Örnek 10:

```
enbuyuk = (sayi1 > sayi2) ? sayi1 : sayi2;
```

```
System.out.println((sayi % 2 == 0) ? "sayi cift" : "sayi tek");
```

# Ödev 1:

13. satırdan önce System.currentTimeMillis() değerini konsolda gösteren kod satırını ekleyiniz.

```
1  //Toplama Testi
2  //
3  import java.util.Scanner;
4
5  public class ToplamaTesti {
6      public static void main(String [] args) {
7          int sayi1 = (int)(System.currentTimeMillis() % 10);
8          int sayi2 = (int)(System.currentTimeMillis() / 7 % 10);
9
10         // Scanner oluşturma (klavyeden/konsoldan giriş için)
11         Scanner giris = new Scanner(System.in);
12
13         System.out.print(sayi1 + " + " + sayi2 + " sonucu nedir? ");
14
15         int cevap = giris.nextInt();
16
17         System.out.println(sayi1 + " + " + sayi2 + " = " + cevap
18             + " sonucu " + (sayi1 + sayi2 == cevap));
19     }
20 }
```

## Ödev 2:

$ax^2 + bx + c = 0$  denklemin köklerini bulan programı Java dilinde yazınız. Kök hesaplama:

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \text{ ve } x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a} \text{ şeklindedir.}$$

# Ödev 3: Taş, kağıt, makas oyunu.

Taş-kağıt-makas oyununu oynayabilecek bir program yazınız. (*Makas kağıdı keser, taş makası ezer, kağıt taşı yener*). Program; taş (0), kağıt (1) ve makas (2) temsil etmek üzere 0, 1 ve 2 sayılardan birini rastgele üretir, daha sonra kullanıcıya seçmesi için mesaj verir: taş (0), kağıt (1) ve makas (2) hangisi?:

Kullanıcı bir değer yazıp ↵ bastıktan sonra program bilgisayarın seçtiğini ve kullanıcının seçtiğini konsola yazar; kimin kazandığını kimin kaybettiğini veya beraberliği açıklar.

//Örnek çalışma 1

taş (0), kağıt (1) ve makas (2) hangisi?: 2 ↵  
Bilgisayar taş, sen makas. Kaybettin!

//Örnek çalışma 2

taş (0), kağıt (1) ve makas (2) hangisi?: 2 ↵  
Bilgisayar kağıt, sen makas. Kazandın!

# Ödev 4:

Java ve Bilişim terimleri olarak

---

---

---

nedir?

İngilizce ve Türkçe karşılıklarını yazınız.

# 10. Ders: JAVA Programlama Dili (Mat. fonk., Unicode, ASCII, char, printf(), String)

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü

YMH111 Algoritma ve Programlama-I

Doç. Dr. Yaman Akbulut

# JAVA Programlama Dili

- <http://www.kriptarium.com/algoritma.html> (Yardımcı kaynak)
- JAVA Programlama Dili (Ders: 47-66) video. (**Mutlaka İzleyiniz!**)
  - Ders 47: Yorum/Açıklama Satırları (izle)
  - Ders 48: Java String (izle)
  - Ders 49: Java String Sınıf Neden Değiştirilemezdir? (izle)
  - Ders 50: Java String Karşılaştırma İşlemleri (izle)
  - Ders 51: Java String Karşılaştırma İşlemleri Örnek (izle)
  - Ders 52: Java String Ekleme İşlemleri (izle)
  - Ders 53: Java Substring (izle)
  - Ders 54: Java String Sınıfının Bazı Metotları (izle)
  - Ders 55: Java StringBuffer Sınıfı (izle)
  - Ders 56: Java StringBuilder Sınıfı (izle)
  - Ders 57: String ve StringBuffer Arasındaki Farklar (izle)
  - Ders 58: StringBuffer ve StringBuilder Arasındaki Farklar (izle)
  - Ders 59: Java toString() method (izle)
  - Ders 60: StringTokenizer (izle)
  - Ders 61: Tip Dönüşümleri I (izle)
  - Ders 62 - Tip Dönüşümleri II (izle)
  - Ders 63 - Tip Dönüşümleri III (izle)
  - Ders 64 - Tip Dönüşümleri IV (izle)
  - Ders 65 - Tip Dönüşümleri V (izle)
  - Ders 66 - Tip Dönüşümleri VI (izle)

# JAVA'da program yazma ve çalıştırma ortamı

1. JDK + notepad++

2. IDE NetBeans

# Özel Karakterler

|     |                 |
|-----|-----------------|
| { } | braces          |
| ( ) | parentheses     |
| [ ] | brackets        |
| //  | double slashes  |
| " " | quotation marks |
| ;   | semicolon       |

# Java Keywords

|          |            |           |              |
|----------|------------|-----------|--------------|
| abstract | double     | int       | super        |
| assert   | else       | interface | switch       |
| boolean  | enum       | long      | synchronized |
| break    | extends    | native    | this         |
| byte     | final      | new       | throw        |
| case     | finally    | package   | throws       |
| catch    | float      | private   | transient    |
| char     | for        | protected | try          |
| class    | goto       | public    | void         |
| const    | if         | return    | volatile     |
| continue | implements | short     | while        |
| default  | import     | static    |              |
| do       | instanceof | strictfp  |              |

# Matematiksel fonksiyonlar-1

**Math** sınıfında trigonometrik fonksiyon işlemleriyle ilgili aşağıdaki metotlar bulunmaktadır.

## Trigonometrik Metotlar

### metot

**sin**(radyan)

**cos**(radyan)

**tan**(radyan)

**toRadians**(derece)

**toDegree**(radyan)

**asin**(a)

**acos**(a)

**atan**(a)

### açıklama

bir radyan açının trigonometrik sinüsünü döndürür.

bir radyan açının trigonometrik kosinüsünü döndürür.

bir radyan açının trigonometrik tanjantını döndürür.

derece açının radyan değerini döndürür.

radyan açının derece değerini döndürür.

arcsin radyan

arccos radyan

arctan radyan

# Trigonometrik örnekler

Math.toDegrees(Math.PI / 2) sonucu 90.0

Math.toRadians(30) sonucu 0.5236 ( $\pi/6$  ile aynı)

Math.sin(0) sonucu 0.0

Math.sin(Math.toRadians(270)) sonucu -1.0

Math.sin(Math.PI / 6) sonucu 0.5

Math.sin(Math.PI / 2) sonucu 1.0

Math.cos(0) sonucu 1.0

Math.cos(Math.PI / 6) sonucu 0.866

Math.cos(Math.PI / 2) sonucu 0

Math.asin(0.5) sonucu 0.523598333 ( $\pi/6$  ile aynı)

Math.acos(0.5) sonucu 1.0472 ( $\pi/3$  ile aynı)

Math.atan(1.0) sonucu 0.785398 ( $\pi/4$  ile aynı)

**DENEYİN!**

**ölçün!**

**kontrol edin!**

# Matematiksel fonksiyonlar-2

**Math** sınıfında 5 üstel metot bulunmaktadır.

## Üstel Metotlar

**metot**

**açıklama**

**exp(x)**

$e^x$  değerini döndürür.

**log(x)**

doğal logaritma  $\ln(x)$  değerini döndürür. ( $\log_e x$ )

**log10(x)**

10 tabanında logaritma  $x$  değerini döndürür. ( $\log_{10} x$ )

**pow(a, b)**

$a^b$  değerini döndürür.

**sqrt(x)**

karekök  $x$  değerini döndürür.

# Üstel örnekler

DENEYİN!

ölçün!

kontrol edin!

Math.exp(1) sonucu 2.71828

Math.log(Math.E) sonucu 1.0

Math.log10(10) sonucu 1.0

Math.log10(100) sonucu 2.0

Math.pow(2, 3) sonucu 8.0

Math.pow(3, 2) sonucu 9.0

Math.pow(4.5, 2.5) sonucu 22.91765

Math.sqrt(4) sonucu 2.0

Math.sqrt(16) sonucu 4.0

Math.sqrt(10.5) sonucu 4.24

# Matematiksel fonksiyonlar-3

**Math** sınıfında 4 yuvarlama metodu bulunmaktadır.

## Yuvarlama Metotları

| metot            | açıklama                                                                                                                          |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| <b>ceil</b> (x)  | x üstteki en yakın tamsayıya yuvarlanır, dönen değer double olur.                                                                 |
| <b>floor</b> (x) | x alttaki en yakın tamsayıya yuvarlanır, dönen değer double olur.                                                                 |
| <b>rint</b> (x)  | x en yakın tamsayıya yuvarlanır, dönen değer double olur.<br>x alt ve üst taraftaki tamsayıya eşit uzaklıkta ise çift sayı döner. |
| <b>round</b> (x) | x, float ise (int)Math.floor(x + 0.5) sonucunu döndürür.<br>x, double ise (long)Math.floor(x + 0.5) sonucunu döndürür.            |

# Yuvarlama örnekleri

Math.ceil(2.1) sonucu 4.0

Math.ceil(2.0) sonucu 2.0

Math.ceil(-2.0) sonucu -2.0

Math.ceil(-2.1) sonucu -2.0

Math.floor(2.1) sonucu 2.0

Math.floor(2.0) sonucu 2.0

Math.floor(-2.0) sonucu -2.0

Math.floor(-2.1) sonucu -4.0

Math rint(2.1) sonucu 2.0

Math rint(-2.1) sonucu -2.0

Math rint(4.5) sonucu 4.0

Math rint(-2.0) sonucu -2.0

Math rint(2.5) sonucu 2.0

Math rint(-2.5) sonucu -2.0

Math.round(2.6f) sonucu 3

// int döner

Math.round(2.0) sonucu 2

// long döner

Math.round(-2.0f) sonucu -2

// int döner

Math.round(-2.6) sonucu -3

// long döner

Math.round(-2.4) sonucu -2

// long döner

**DENEYİN!**

**ölçün!**

**kontrol edin!**

# Matematiksel fonksiyonlar-4

## min, max ve abs metotları

### min, max, abs Metotları

| metot                  | açıklama                                                                  |
|------------------------|---------------------------------------------------------------------------|
| <code>min(x, y)</code> | iki sayıdan küçük olanı döndürür. (sayılar: int, long, float, double)     |
| <code>max(x, y)</code> | iki sayıdan büyük olanı döndürür. (sayılar: int, long, float, double)     |
| <code>abs(x)</code>    | bir sayının mutlak değerini döndürür. (sayılar: int, long, float, double) |

# min, max, abs örnekleri

Math.max(2, 3) sonucu 3

Math.max(2.5, 3) sonucu 3.0

Math.min(2.5, 4.6) sonucu 2.5

Math.abs(-2) sonucu 2

Math.abs(-2.1) sonucu 2.1

DENEYİN!

ölçün!

kontrol edin!

# Matematiksel fonksiyonlar-5

## random metodu

| metot         | açıklama                                                                                                                        |
|---------------|---------------------------------------------------------------------------------------------------------------------------------|
| Math.random() | <p>Bu yöntem 0.0'dan büyük veya eşit ve 1.0'den küçük rastgele bir double değer üretir.</p> $0 \leq \text{Math.random()} < 1.0$ |

# random örnekleri

`(int)(Math.random() * 10)`

0 ile 9 arasında rastgele bir tamsayı üretir.

`50 + (int)(Math.random() * 50)`

50 ile 99 arasında rastgele bir tamsayı üretir.

Genelleştirilmiş şekliyle:

`a + Math.random() * b`

a ile a + b arasında rastgele bir tamsayı üretir. (a+b hariç)

DENEYİN!

ölçün!

kontrol edin!

# char Veri Tipi

**char** sadece bir karakteri temsil eder.

```
char harf = 'A';  
char sayiChar = '4';
```

**char** için 'tek tırnak'  
**String** için "çift tırnak"  
kullanılır.

Unicode ve ASCII karakterler **char** veri tipi ile ifade edilebilir.

# Unicode

**Original Unicode**, Unicode Konsorsiyumu tarafından kurulmuştur.

16-bit karakter kodlaması olarak tasarlanmıştır.

**char** veri tipiyle bu karakterler temsil edilebilir.

Java **Unicode**'u destekler.

16-bit orijinal Unicode 65.536 karakteri barındırır ancak bu tüm dünyadaki karakterleri tanımlamak için yeterli değildir.

Bu nedenle Unicode 1.112.064 karakter sayısına genişletilmiştir. Genişletilmiş Unicode veya Ek Unicode şeklinde söylenebilir.

# Unicode

**Original Unicode** karakterler 16-bit'tir ve bu karakterler **char** veri tipinde saklanabilir.

16-bit, 2 Byte eder. Bu da **\u0000** hexadecimal (onaltılık) sayı sisteminde 4-bitlik 4 hane ile gösterilebilir.

**\u** hexadecimal (onaltılık) gösterim için kullanılır. Karakter aralığı **\u0000**'dan **\uFFFF**'e kadardır.

Yunan harfleri  $\alpha$ ,  $\beta$  ve  $\gamma$ 'nın Unicode'ları **\u03b1**, **\u03b2**, **\u03b3** şeklindedir. Decimal (onluk) karşılıkları ise sırasıyla **945**, **946** ve **947**. Klavyede ALT + **945** yaparsanız alfa karakterini ortaya çıkar. (945 numlock tarafında yazılmalı)

# ASCII (American Standard Code for Information Interchange)

ASCII kodlama sistemi 8-bit'tir.

Birçok bilgisayar büyük harf, küçük harf, rakamlar, noktalama işaretleri, kontrol işaretleri için ASCII kullanır.

Unicode, ASCII nin 7-bitlik `\u0000` ile `\u007F` lik bölümünü (128 adet ASCII) içermektedir.

Alfabemizde bulunan A, B ve C karakterlerinin ASCII kodları `\u0041`, `\u0042`, `\u0043` şeklindedir. Decimal (onluk) karşılıkları ise sırasıyla 65, 66 ve 67'dir. Klavyede ALT + 65 yaparsanız büyük A karakterini yazabilirsiniz. (65 numlock tarafında yazılmalı)

# ASCII (American Standard Code for Information Interchange)

## Sık kullanılan ASCII kodları

| Karakter  | Decimal değeri | Unicode değeri  |
|-----------|----------------|-----------------|
| '0' – '9' | 48 – 57        | \u0030 – \u0039 |
| 'A' – 'Z' | 65 – 90        | \u0041 – \u005A |
| 'a' – 'z' | 97 – 122       | \u0061 – \u007A |

```
char harf = 'A';  
char harf = '\u0041';
```

```
char ch = 'a';  
System.out.println(++ch);
```

# ASCII karakter seti (decimal)

|    | 0   | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 0  | nul | soh | stx | etx | eot | enq | ack | bel | bs  | ht  |
| 1  | nl  | vt  | ff  | cr  | so  | si  | dle | dcl | dc2 | dc3 |
| 2  | dc4 | nak | syn | etb | can | em  | sub | esc | fs  | gs  |
| 3  | rs  | us  | sp  | !   | ”   | #   | \$  | %   | &   | ,   |
| 4  | (   | )   | *   | +   | ,   | —   | .   | /   | 0   | 1   |
| 5  | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | :   | ;   |
| 6  | <   | =   | >   | ?   | @   | A   | B   | C   | D   | E   |
| 7  | F   | G   | H   | I   | J   | K   | L   | M   | N   | O   |
| 8  | P   | Q   | R   | S   | T   | U   | V   | W   | X   | Y   |
| 9  | Z   | [   | \   | ]   | ^   | —   | ,   | a   | b   | c   |
| 10 | d   | e   | f   | g   | h   | i   | j   | k   | l   | m   |
| 11 | n   | o   | p   | q   | r   | s   | t   | u   | v   | w   |
| 12 | x   | y   | z   | {   |     | }   | ~   | del |     |     |

# ASCII karakter seti (hexadecimal)

|   | 0   | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9  | A   | B   | C  | D  | E  | F   |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|----|-----|-----|----|----|----|-----|
| 0 | nul | soh | stx | etx | eot | enq | ack | bel | bs  | ht | nl  | vt  | ff | cr | so | si  |
| 1 | dle | dcl | dc2 | dc3 | dc4 | nak | syn | etb | can | em | sub | esc | fs | gs | rs | us  |
| 2 | sp  | !   | "   | #   | \$  | %   | &   | '   | (   | )  | *   | +   | ,  | -  | .  | /   |
| 3 | 0   | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9  | :   | ;   | <  | =  | >  | ?   |
| 4 | @   | A   | B   | C   | D   | E   | F   | G   | H   | I  | J   | K   | L  | M  | N  | O   |
| 5 | P   | Q   | R   | S   | T   | U   | V   | W   | X   | Y  | Z   | [   | \  | ]  | ^  | -   |
| 6 | '   | a   | b   | c   | d   | e   | f   | g   | h   | i  | j   | k   | l  | m  | n  | o   |
| 7 | p   | q   | r   | s   | t   | u   | v   | w   | x   | y  | z   | {   |    | }  | ~  | del |

# Hexadecimal Binary Decimal

| <i>Hexadecimal</i> | <i>Binary</i> | <i>Decimal</i> |
|--------------------|---------------|----------------|
| 0                  | 0000          | 0              |
| 1                  | 0001          | 1              |
| 2                  | 0010          | 2              |
| 3                  | 0011          | 3              |
| 4                  | 0100          | 4              |
| 5                  | 0101          | 5              |
| 6                  | 0110          | 6              |
| 7                  | 0111          | 7              |
| 8                  | 1000          | 8              |
| 9                  | 1001          | 9              |
| A                  | 1010          | 10             |
| B                  | 1011          | 11             |
| C                  | 1100          | 12             |
| D                  | 1101          | 13             |
| E                  | 1110          | 14             |
| F                  | 1111          | 15             |

# Escape karakteri (Kaçış karakteri \)

```
System.out.println("Burası "Yazilim Muhendisligi");  
// hata verir
```

```
System.out.println("Burası \"Yazilim Muhendisligi\"");
```

Çıktı sonucu: Burası "Yazilim Muhendisligi"

# Escape karakteri (Kaçış karakteri \)

## escape sequence

| kaçış kodları | adı             | Unicode kodu | decimal değeri |
|---------------|-----------------|--------------|----------------|
| \b            | backspace       | \u0008       | 8              |
| \t            | tab             | \u0009       | 9              |
| \n            | linefeed        | \u000A       | 10             |
| \f            | formfeed        | \u000C       | 12             |
| \r            | carriage return | \u000D       | 13             |
| \\            | backslash       | \u005C       | 92             |
| \"            | double quote    | \u0022       | 34             |

# Escape karakteri (Kaçış karakteri \)

```
System.out.println("\\t tab karakteridir.");
```

Çıktı sonucu:

```
"\t tab karakteridir."
```

```
System.out.println("Path su sekilde C:\\Program Files\\Java\\jdk-15.0.1\\bin");
```

Çıktı sonucu:

```
"Path su sekilde C:\Program Files\Java\jdk-15.0.1\bin"
```

# char ve nümerik tipler arasında çevrim-1

char herhangi bir nümerik tipe çevrilebilir ve bunun tersi de geçerlidir. Bir tamsayı char'a çevrilirken en düşük 16-bit kullanılır kalanı göz ardı edilir.

Örnek olarak:

```
char ch = (char)0xAB0041;    // En düşük 16-bit hexadecimal 0041
                               // ch değişkenine atanıyor
System.out.println(ch);      // ch, A karakteridir.
```

# char ve nümerik tipler arasında çevrim-2

Bir ondalık sayı bir karaktere (**char**) dönüştürüldüğünde, ondalık sayı önce bir int'e dönüştürülür ve daha sonra bir karaktere (**char**) dönüştürülür.

Örnek olarak:

```
char ch = (char)65.25;           // Onluk 65 değeri, ch değişkenine atanıyor.
```

```
System.out.println(ch);         // ch, A karakteridir.
```

# char ve nümerik tipler arasında çevrim-3

Bir karakter (char) nümerik bir tipe dönüştürüldüğünde, karakterin Unicode'u belirtilen nümerik tipe dönüştürülür.

Örnek olarak:

```
int i = (int)'A';           // A karakterinin Unicode değeri, i değişkenine atanıyor.
```

```
System.out.println(i); // i, 65 olur.
```

# char ve nümerik tipler arasında çevrim-4

Dönüşüm yapılan değer hedef değişkene uyuyorsa aşağıdaki gibi kapalı çevrimle kullanılır.

```
byte b = 'a'; // 'a' nın Unicode değeri 97  
int i = 'a';
```

Dönüşüm yapılan değer hedef değişkene uymuyorsa açık çevrimle yapılmalıdır.

```
byte b = '\uFFF4'; // bu hata verir.
```

Açık çevrim şu şekilde kullanılır:

```
byte b = (byte)\uFFF4;
```

# char ve nümerik tipler arasında çevrim-5

Onaltılık olarak 0 ile FFFF arasındaki herhangi bir pozitif tam sayı, kapalı çevrimle bir karaktere dönüştürülebilir. Bu aralıkta olmayan herhangi bir sayı açık çevrimle bir karaktere dönüştürülmelidir.

```
int i = '2' + '3';           // (int)'2', 50 ve (int)'3', 51 eder.  
System.out.println("i: " + i); // i: 101
```

```
int j = 2 + 'a';             // (int)'a', 97 eder.  
System.out.println("j: " + j); // j: 99
```

```
System.out.println(j + ", " + (char)j // 99, c nin Unicode karakteridir.  
+ " nin Unicode karakteridir.");
```

```
System.out.println("Chapter " + '2'); // Chapter 2
```

# char karşılaştırma-1

İki karakter sayılar gibi Unicode değerleri üzerinden karşılaştırılabilir.

'a' < 'b'    **true**, 'a' nın (97) Unicode değeri, 'b' nin Unicode (98) değerinden küçüktür.

'a' < 'A'    **false**, 'a' nın (97) Unicode değeri, 'A' nın Unicode (65) değerinden büyüktür.

'1' < '8'    **true**, '1' in (49) Unicode değeri, '8' in Unicode (56) değerinden küçüktür.

# char karşılaştırma-2

|   | 0   | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9  | A   | B   | C  | D  | E  | F   |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|----|-----|-----|----|----|----|-----|
| 0 | nul | soh | stx | etx | eot | enq | ack | bel | bs  | ht | nl  | vt  | ff | cr | so | si  |
| 1 | dle | dcl | dc2 | dc3 | dc4 | nak | syn | etb | can | em | sub | esc | fs | gs | rs | us  |
| 2 | sp  | !   | "   | #   | \$  | %   | &   | '   | (   | )  | *   | +   | ,  | -  | .  | /   |
| 3 | 0   | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9  | :   | ;   | <  | =  | >  | ?   |
| 4 | @   | A   | B   | C   | D   | E   | F   | G   | H   | I  | J   | K   | L  | M  | N  | O   |
| 5 | P   | Q   | R   | S   | T   | U   | V   | W   | X   | Y  | Z   | [   | \  | ]  | ^  | _   |
| 6 | `   | a   | b   | c   | d   | e   | f   | g   | h   | i  | j   | k   | l  | m  | n  | o   |
| 7 | p   | q   | r   | s   | t   | u   | v   | w   | x   | y  | z   | {   |    | }  | ~  | del |

Program yazarken bir karakterin harf mi sayı mı, büyük harf mi yoksa küçük harf mi diye test etmemiz gerekebilir. Bunun için karakterlerin Unicode'larının bulunduğu ASCII karakter tablosundan faydalanabiliriz. Tablodan 'a' dan 'z' ye, 'A' dan 'Z' ye ve '0' dan '9' sıralı bir şekilde ardışık olarak kodlar devam etmektedir.

```
if (ch >= 'A' && ch <= 'Z')
```

```
    System.out.println(ch + " karakteri büyük harftir");
```

```
else if (ch >= 'a' && ch <= 'z')
```

```
    System.out.println(ch + " karakteri küçük harftir ");
```

```
else if (ch >= '0' && ch <= '9')
```

```
    System.out.println(ch + " sayısal karakterdir.");
```

# char karşılaştırma-3

**Character** sınıfında aşağıdaki metotlar vardır.

## metot

**isDigit**(ch)

**isLetter**(ch)

**isLetterOfDigit**(ch)

**isLowerCase**(ch)

**isUpperCase**(ch)

**toLowerCase**(ch)

**toUpperCase**(ch)

## açıklama

Belirtilen karakter sayı ise **true** döndürür.

Belirtilen karakter harf ise **true** döndürür.

Belirtilen karakter sayı veya harf ise **true** döndürür.

Belirtilen karakter küçük harf ise **true** döndürür.

Belirtilen karakter BÜYÜK harf ise **true** döndürür.

Belirtilen karakteri küçük harfe çevirir.

Belirtilen karakteri BÜYÜK harfe çevirir.

# char karşılaştırma örnekleri

```
System.out.println("isDigit('a') " + Character.isDigit('a')); // isDigit('a') false
```

```
System.out.println("isLetter('a') " + Character.isLetter('a')); // isLetter('a') true
```

```
System.out.println("isLowerCase('a') " + Character.isLowerCase('a')); // isLowerCase('a') true
```

```
System.out.println("isUpperCase('a') " + Character.isUpperCase('a')); // isUpperCase('a') false
```

```
System.out.println("toLowerCase('T') " + Character.toLowerCase('T')); // toLowerCase('T') t
```

```
System.out.println("toUpperCase('q') " + Character.toUpperCase('q')); //toUpperCase('q') Q
```

# String Tipi

char

sadece bir karakteri temsil eder.

```
char ch = 'A';
```

char için 'tek tırnak'  
String için "çift tırnak"  
kullanılır.

String

birden fazla karakter dizisini ifade eder.

```
String mesaj = "Merhaba Dünya";
```

# String tipi için temel metotlar

| metot         | açıklama                                                                                                   | <div>whitespace karakterler:<br/>' ', \t, \f, \r, \n</div> |
|---------------|------------------------------------------------------------------------------------------------------------|------------------------------------------------------------|
| length()      | dizideki karakter sayısını döndürür.                                                                       |                                                            |
| charAt(index) | dizideki belirtilen indeksteki karakteri döndürür.                                                         |                                                            |
| concat(s1)    | bu dizi ile s1 dizisini birleştirerek yeni bir dizi döndürür.                                              |                                                            |
| toUpperCase() | dizideki bütün karakterleri büyük harf yapıp yeni bir dizi döndürür.                                       |                                                            |
| toLowerCase() | dizideki bütün karakterleri küçük harf yapıp yeni bir dizi döndürür.                                       |                                                            |
| trim()        | dizinin her iki yanındaki whitespace karakterleri (' ', \t, \f, \r, \n) yok ederek yeni bir dizi döndürür. |                                                            |

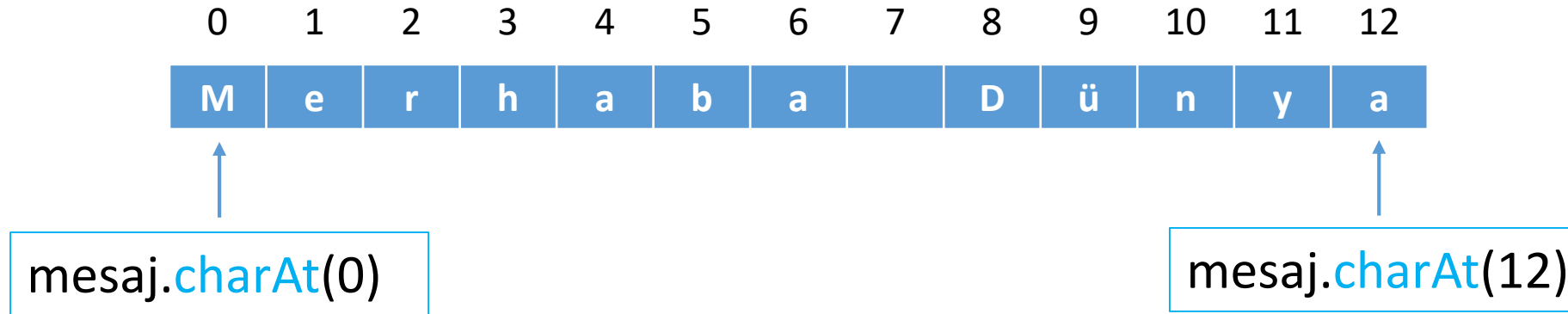
# length() dizinin uzunlugu

```
String mesaj = "Merhaba Dünya";  
System.out.println(mesaj + " mesajinin uzunlugu " + mesaj.length() + " dir. ");  
// Merhaba Dünya mesajinin uzunlugu 13 dir.
```

```
String mesaj2 = "";  
System.out.println(mesaj2 + " mesajinin uzunlugu " + mesaj2.length() + " dir. ");  
// ?? deneyiniz
```

DE  
NE  
Yİ  
Nİ  
Z!

# charAt(index) indekse göre karakter getirme



# concat(s1) concatenating, dizileri birleştirme

`String s3 = s1.concat(s2);`      =      `String s3 = s1 + s2;`

`String benimMesaj = mesaj + " ve " + " arkadaşlar";`

```
String mesaj = "Merhaba " + "Dünya " + "ve " + "arkadaşlar" ;
```

```
String s1 = "Ek-" + 'B' ;  
// s1, Ek-B olur
```

```
String s = "Bölüm" + 2 ;  
// s, Bölüm2 olur
```

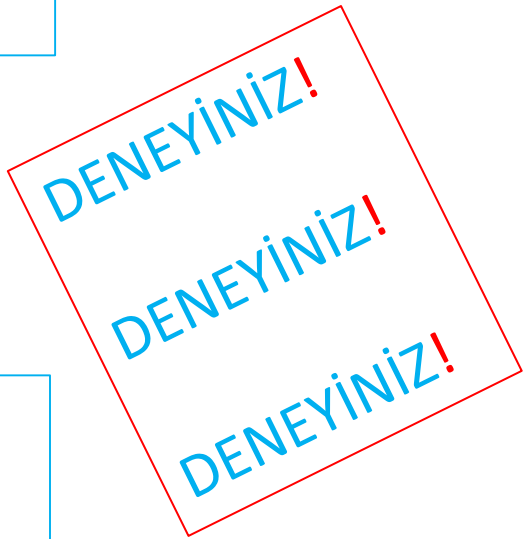
Daha önceki mesaj içeriği "Merhaba Dünya" idi.  
Şimdi buna += operatörü ile " ve arkadaşlar hoşgeldiniz" dizisini ekleyebiliyoruz.

```
mesaj += " ve arkadaşlar hoşgeldiniz";
```

```
//yeni mesaj, "Merhaba Dünya ve arkadaşlar hoşgeldiniz" olur.
```

```
int i = 2, j = 3 ise  
System.out.println(" i + j " + i + j);  
// sonucu nedir?
```

```
int i = 2, j = 3 ise  
System.out.println(" i + j " + (i + j));  
// sonucu nedir?
```



# toLowerCase(), toUpperCase(), trim()

```
"Merhaba".toLowerCase()  
// merhaba haline gelir
```

```
"Merhaba".toUpperCase()  
// MERHABA haline gelir
```

**trim()** bir dizinin her iki yanındaki whitespace karakterlerini yok eder.

whitespace karakterler: ' ', \t, \f, \r, \n

```
"\t      İyi Akşamlar      \n".trim()  
// İyi Akşamlar haline gelir
```

# next() Konsoldan string (dizi) girişı

```
Scanner giris = new Scanner(System.in);  
String s = giris.next();
```

## Örnek 1:

```
1 //String Tipi
2 import java.util.Scanner;
3
4 public class StringTipi {
5     public static void main(String [] args) {
6         Scanner giris = new Scanner(System.in);
7         System.out.print("Boslukla ayrilmis 3 kelime yaziniz: ");
8         String s1 = giris.next();
9         String s2 = giris.next();
10        String s3 = giris.next();
11        System.out.println("s1: " + s1);
12        System.out.println("s2: " + s2);
13        System.out.println("s3: " + s3);
14    }
15 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>javac StringTipi.java
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java StringTipi
Boslukla ayrilmis 3 kelime yaziniz: Yazilim Muhendisligi Bolumu
s1: Yazilim
s2: Muhendisligi
s3: Bolumu
```

# nextLine() Konsoldan satır girişi

```
Scanner giris = new Scanner(System.in);  
String s = giris.nextLine();
```

## Örnek 2: nextLine() tüm satır

```
1 //String Tüm Satır
2 import java.util.Scanner;
3
4 public class StringTumSatir {
5     public static void main(String [] args) {
6         Scanner giris = new Scanner(System.in);
7         System.out.print("Bir satir yaziniz: ");
8         String s = giris.nextLine();
9         System.out.println("Girilen satir: " + s);
10    }
11 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>javac StringTumSatir.java
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java StringTumSatir
```

```
Bir satir yaziniz: Yazilim Muhendisligi Bolumu Algoritma ve Programlama Dersi
```

```
Girilen satir: Yazilim Muhendisligi Bolumu Algoritma ve Programlama Dersi
```

# nextLine() Konsoldan karakter(char) okuma

```
Scanner giris = new Scanner(System.in);  
String s = giris.nextLine();  
char ch = s.charAt(0);
```

# Örnek 3: nextLine() karakter

```
1 //Karakter Okuma
2 import java.util.Scanner;
3
4 public class KarakterOkuma {
5     public static void main(String [] args) {
6         Scanner giris = new Scanner(System.in);
7         System.out.print("Bir karakter yaziniz: ");
8         String s = giris.nextLine();
9         char ch = s.charAt(0);
10        System.out.println("Girilen karakter: " + ch);
11    }
12 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java KarakterOkuma
Bir karakter yaziniz: Yazılım
Girilen karakter: Y
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java KarakterOkuma
Bir karakter yaziniz: Y
Girilen karakter: Y
```

# Karakter Dizilerinin (String) Karşılaştırılması

| metot (yöntem)                       | açıklama                                                                                                                                        |
|--------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>equals(s1)</code>              | dizi, s1 dizisine eşit ise <b>true</b> döndürür.                                                                                                |
| <code>equalsIgnoreCase(s1)</code>    | dizi, s1 dizisine eşit ise <b>true</b> döndürür, büyük küçük harf dikkate alınmaz.                                                              |
| <code>compareTo(s1)</code>           | dizi, s1 dizisinden büyükse sıfırdan büyük bir sayı, eşitse sıfır ve küçükse sıfırdan küçük bir sayı döndürür.                                  |
| <code>compareToIgnoreCase(s1)</code> | dizi, s1 dizisinden büyükse sıfırdan büyük bir sayı, eşitse sıfır ve küçükse sıfırdan küçük bir sayı döndürür, büyük küçük harf dikkate alınmaz |
| <code>startsWith(prefix)</code>      | dizi, belirlenen bir önek ile başlamışsa <b>true</b> döndürür.                                                                                  |
| <code>endsWith(suffix)</code>        | dizi, belirlenen bir son ek ile bitmişse <b>true</b> döndürür.                                                                                  |
| <code>contains(s1)</code>            | dizinin içinde s1 alt stringi mevcut ise <b>true</b> döndürür.                                                                                  |

```
String s1 = "Merhaba Dunya";  
String s2 = "Merhaba Dunya";  
String s3 = "Merhaba Mars";  
System.out.println(s1.equals(s2)); // true  
System.out.println(s1.equals(s3)); // false
```

```
s1.compareTo(s2)  
// bunun sonucunda 0 döner.
```

String karşılaştırma için  
>, >=, <, <=  
kullanmayınız.

compareTo sonucunda  
sıfır,  
pozitif tamsayı ya da  
negatif tam sayı döner.

```
String s1 = "Merhaba Dunya";  
String s2 = "Merhaba Dunya";  
String s3 = "Merhaba Mars";
```

```
s1.startsWith("Me"); // true  
s1.startsWith("me"); // false
```

```
s1.endsWith("ya"); // true  
s1.endsWith("y"); // false
```

```
s1.contains("Du"); // true  
s1.contains("du"); // false
```

## Örnek 4:

```
1 //İki Sehir Siralama
2 import java.util.Scanner;
3
4 public class IkiSehirSiralama {
5     public static void main(String [] args) {
6         Scanner giris = new Scanner(System.in);
7
8         System.out.print("İlk sehri yaziniz: ");
9         String sehir1 = giris.nextLine();
10        System.out.print("İkinci sehri yaziniz: ");
11        String sehir2 = giris.nextLine();
12
13        if (sehir1.compareTo(sehir2) < 0)
14            System.out.println("Sehirlerin alfabetik sirasi: " +
15                               sehir1 + ", " + sehir2);
16        else
17            System.out.println("Sehirlerin alfabetik sirasi: " +
18                               sehir2 + ", " + sehir1);
19    }
20 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java IkiSehirSiralama
İlk sehri yaziniz: Istanbul
İkinci sehri yaziniz: Ankara
Sehirlerin alfabetik sirasi: Ankara, Istanbul
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java IkiSehirSiralama
İlk sehri yaziniz: Ankara
İkinci sehri yaziniz: Istanbul
Sehirlerin alfabetik sirasi: Ankara, Istanbul
```

# substring(index1, [index2]) Alt string elde etme

metot (yöntem)

açıklama

substring(baslamaIndeksi)

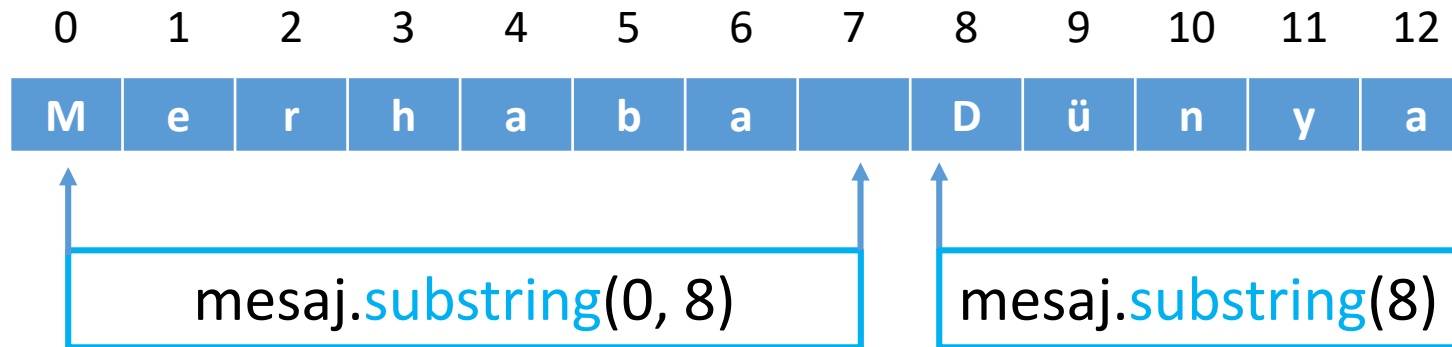
dizideki karakterleri **baslamaIndeksi**nden dizinin sonuna kadar elde eder.

substring(baslamaIndeksi, bitisIndeksi)

dizideki karakterleri **baslamaIndeksi**nden **bitisIndeksi-1**'e kadar elde eder.

# substring(index) alt stringleri elde etme

mesaj



başlamaİndeksi

bitisİndeksi'ne eşit ise  
boş döner.

başlamaİndeksi

bitisİndeksi'nden büyük  
ise hata verir.

```
String altMesaj1 = mesaj.substring(0, 8); // "Merhaba "  
String altMesaj2 = mesaj.substring(8);   // "Dünya"
```

# Karakter ya da alt string Bulma

metot (yöntem)

açıklama

`indexOf(ch)`

dizi içindeki **ilk** ch'nin indeksini döndürür, eşleşme olmazsa -1 döndürür.

`indexOf(ch, indeksSonrasi)`

dizi içindeki `indeksSonrasi`'ndan itibaren **ilk** ch'nin indeksini döndürür, eşleşme olmazsa -1 döndürür.

`indexOf(s)`

dizi içindeki **ilk** s'nin indeksini döndürür, eşleşme olmazsa -1 döndürür.

`indexOf(s, indeksSonrasi)`

dizi içindeki `indeksSonrasi`'ndan itibaren **ilk** s'nin indeksini döndürür, eşleşme olmazsa -1 döndürür.

# Karakter ya da alt string Bulma

metot (yöntem)

açıklama

`lastIndexOf(ch)`

dizi içindeki **son** ch'nin indeksini döndürür, eşleşme olmazsa -1 döndürür.

`lastIndexOf(ch, indeksSonrasi)`

dizi içindeki `indeksSonrasi`'ndan itibaren **son** ch'nin indeksini döndürür, eşleşme olmazsa -1 döndürür.

`lastIndexOf(s)`

dizi içindeki **son** s'nin indeksini döndürür, eşleşme olmazsa -1 döndürür.

`lastIndexOf(s, indeksSonrasi)`

dizi içindeki `indeksSonrasi`'ndan itibaren **son** s'nin indeksini döndürür, eşleşme olmazsa -1 döndürür.

# indexOf()

# ilk görünen indeksi bulma

```
"Merhaba Dünya".indexOf('M');           // 0 döner.  
"Merhaba Dünya".indexOf('a');           // 4 döner.  
"Merhaba Dünya".indexOf('a', 5);        // 6 döner.  
  
"Merhaba Dünya".indexOf("haba");        // 3 döner.  
"Merhaba Dünya".indexOf("Dünya", 5);    // 8 döner.  
"Merhaba Dünya".indexOf("dunya", 5);    // -1 döner.
```

# lastIndexOf()      son görünen indeksi bulma

```
"Merhaba Dünya".lastIndexOf('M');      // 0 döner.  
"Merhaba Dünya".lastIndexOf('a');      // 12 döner.  
"Merhaba Dünya".lastIndexOf('a', 5);      // 12 döner.  
  
"Merhaba Dünya".lastIndexOf("haba");      // 3 döner.  
"Merhaba Dünya".lastIndexOf("Dunya", 5);      // 8 döner.  
"Merhaba Dünya".lastIndexOf("dunya", 5);      // -1 döner.
```

# printf() konsol çıktısı biçimlendirme-3

```
double miktar = 12618.98;  
double ucretOrani = 0.0013;  
double ucret = miktar * ucretOrani;  
System.out.println("Ucret: " + ucret + " TL"); // 16.404674 TL
```

```
double miktar = 12618.98;  
double ucretOrani = 0.0013;  
double ucret = miktar * ucretOrani;  
System.out.println("Ucret: " + (int)(ucret * 100) / 100.0 + " TL"); // 16.4 TL
```

```
double miktar = 12618.98;  
double ucretOrani = 0.0013;  
double ucret = miktar * ucretOrani;  
System.out.printf("Ucret: %4.2f TL", ucret); // 16.40 TL
```

# printf()

# konsol çıktısı biçimlendirme-1

`System.out.printf()` ile konsol (siyah ekran) üzerinde program çıktılarının biçimlerini düzenleyebilirsiniz.

```
System.out.printf(" biçimli metin ", nesne1, nesne2, ..., nesneN);
```

```
// nesne: sayı, karakter, true/false değer, string olabilir.
```

```
// " biçimli metin ": nesneleri ve diğer istenen metinleri (yazıları) barındıran bölüm.
```

# printf() Sık kullanılan biçim tanımlayıcıları

| biçim tanımlayıcısı | çıktı                      | örnek               |
|---------------------|----------------------------|---------------------|
| %b                  | boolean değer              | true veya false     |
| %c                  | karakter                   | 'a'                 |
| %d                  | onluk tam sayı             | 200                 |
| %f                  | ondalık sayı               | 45.460000           |
| %e                  | sayının bilimsel gösterimi | 4.556000e+01        |
| %s                  | karakter dizisi, string    | "Java eğlencelidir" |

# printf() biçim tanımlayıcısı

```
System.out.printf("Ucret: %4.2f TL", ucret); // 16.40 TL
```

biçim tanımlayıcısı



```
System.out.printf("Ucret: %5.3f TL", ucret); // ??
```

# printf() biçim tanımlayıcıları

```
int adet = 5;  
double miktar = 45.56;  
System.out.printf("Adet %d ve miktar %f", adet, miktar);  
  
// Adet 5 ve miktar 45.560000
```



printf() içinde %'nin kendisini kullanmak için %% yazınız.

# printf() Alan genişliği ve hassasiyet örnekleri

| örnek               | çıktı                                                                                                                                                                      |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>%5c</code>    | karakteri çıktı verir, karakterden önce 4 boşluk ekler.                                                                                                                    |
| <code>%6b</code>    | boolean değeri çıktı verir, true ise öncesine 2 karakter, false ise öncesine 1 karakter ekler.                                                                             |
| <code>%5d</code>    | tam sayıyı çıktı verir, 5'ten küçük ise öncesine boşluk ekler, 5'ten büyük ise alan genişliği otomatik olarak artar.                                                       |
| <code>%10.2f</code> | noktadan sonra 2 hane olacak, toplam genişlik 10 olduğundan tam kısma 7 hane kalır. 7'den küçük ise boşluk ile doldurulur, 7'den büyük ise otomatik olarak genişlik artar. |
| <code>%10.2e</code> | sayının bilimsel gösterimi (DENEYİNİZ.)                                                                                                                                    |
| <code>%12s</code>   | String en az 12 karakter uzunluğunda olmalı, 12'den az ise boşluk ile doldurulur. 12'den fazla ise otomatik olarak alan genişliği artar.                                   |

## Örnek 5: printf() kullanımı

```
1 //Bicim Demosu
2 public class BicimDemosu {
3     public static void main(String[] args) {
4         // tablonun basligini gosterme
5         System.out.printf("%-10s%-10s%-10s%-10s%-10s\n", "Derece",
6             "Radyan", "Sinus", "Cosinus", "Tanjant");
7
8         // 30 derece icin deger gosterimi
9         int derece = 30;
10        double radyan = Math.toRadians(derece);
11        System.out.printf("%-10d%-10.4f%-10.4f%-10.4f%-10.4f\n",
12            derece, radyan, Math.sin(radyan), Math.cos(radyan),
13            Math.tan(radyan));
14
15        // 60 derece icin deger gosterimi
16        derece = 60;
17        radyan = Math.toRadians(derece);
18        System.out.printf("%-10d%-10.4f%-10.4f%-10.4f%-10.4f\n",
19            derece, radyan, Math.sin(radyan), Math.cos(radyan),
20            Math.tan(radyan));
21    }
22 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java BicimDemosu
Derece      Radyan      Sinus      Cosinus     Tanjant
30           0,5236     0,5000     0,8660     0,5774
60           1,0472     0,8660     0,5000     1,7321
```

# Ödev 1:

Aşağıdaki programın çıktısını gösteriniz.

```
public class Test {  
    public static void main(String[] args) {  
        char x = 'a';  
        char y = 'c';  
        System.out.println(++x);  
        System.out.println(y++);  
        System.out.println(x - y);  
    }  
}
```

# Ödev 2:

`mesaj` dizisindeki harfleri (karakterleri) kullanarak `charAt()` yöntemi ile `"baMya"` kelimesini oluşturunuz. (bir indeks en fazla bir kere kullanılabilir)

|   |   |   |   |   |   |   |   |   |   |    |    |    |
|---|---|---|---|---|---|---|---|---|---|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| M | e | r | h | a | b | a |   | D | ü | n  | y  | a  |

# Ödev 3:

`adSoyad` dizisi içindeki bir kişiye ait adı ve soyadı ayırıp; `ad` ve `soyad` dizilerine kaydeden ve konsolda gösteren programı yazınız.

|   |   |   |   |   |   |   |   |   |   |    |    |    |
|---|---|---|---|---|---|---|---|---|---|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| Y | a | m | a | n |   | A | k | b | u | l  | u  | t  |

# Ödev 4:

Java ve Bilişim terimleri olarak

ceil, floor, abs, radian, asin, max, min, string, String,

---

---

nedir?

İngilizce ve Türkçe karşılıklarını yazınız.

# 11. Ders: JAVA Programlama Dili

## Döngüler (while, do-while, for)

## break, continue, iç içe döngüler

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü

YMH111 Algoritma ve Programlama-I

Doç. Dr. Yaman Akbulut

# JAVA Programlama Dili

- <http://www.kriptarium.com/algorithm.html> (Yardımcı kaynak)
- JAVA Programlama Dili (Ders: 33-41) video. (Mutlaka İzleyiniz!)
  - Ders 33: Döngüler (izle)
  - Ders 34: For Döngü İfadesinin Kullanımı (izle)
  - Ders 35: Döngü Uygulamaları 1 (Faktoriyel Hesabı) (izle)
  - Ders 36: Döngü Uygulamaları 2 (Fibonacci Sayıları) (izle)
  - Ders 37: ForEach Döngü Yapısının Kullanımı (izle)
  - Ders 38: Döngülerde Break Anahtar Sözcüğünün Kullanımı (izle)
  - Ders 39: Döngülerde Break ve Continue Kullanımı (izle)
  - Ders 40: For ile Sonsuz Döngü Örneği (izle)
  - Ders 41: While ve DoWhile İfadeleri ile Döngü Oluşturmak (izle)

# Java Keywords

abstract

assert

boolean

break

byte

case

catch

char

class

const

continue

default

do

double

else

enum

extends

final

finally

float

for

goto

if

implements

import

instanceof

int

interface

long

native

new

package

private

protected

public

return

short

static

strictfp

super

switch

synchronized

this

throw

throws

transient

try

void

volatile

while

# Java'da Döngüler

1. `while` döngüsü
2. `do-while` döngüsü
3. `for` döngüsü

## Döngü nedir?

Programlarda tekrar eden veya tekrar etmesi istenen durumları gerçekleştiren yapılardır.

# Döngü 1: **while** döngüsü

başlangıç değişkeni tanımlama

```
while (döngüyü devam ettirme koşulu) {  
    // Döngünün Gövdesi  
    // Koşul true ise burası yapılır, döngü devam eder  
    // yapılacaklar 1  
    // yapılacaklar 2 ...  
    // değişken artırma ya da azaltma  
}  
// Koşul false ise döngüden çıkılır, buradan devam eder...
```



**while** döngüsü akış diyagramı

# Döngü 1: **while** döngüsü

[illegible]

# Döngü 1: `while` döngüsü

```
// 100 defa "Merhaba Dunya" yazdırma
int sayac = 0;

while (sayac < 100) {
    //Döngünün Gövdesi
    System.out.println("Merhaba Dunya");
    sayac++;
}
```

```
// comment (yorum)
başlangıç değişkeni tanımlama

while (döngüyü devam ettirme koşulu) {
    // Döngünün Gövdesi
    // yapılacaklar 1
    // yapılacaklar 2 ...
    // değişken artırma/azaltma
}

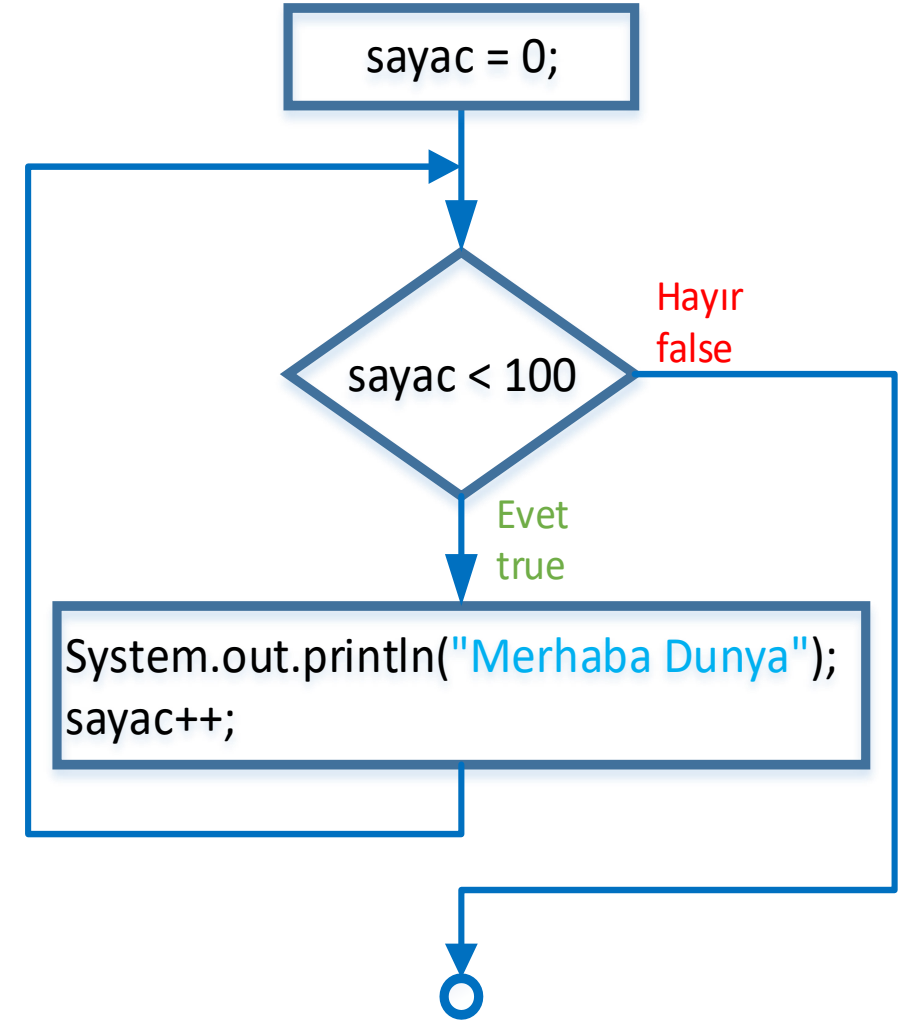
// döngüden çıkınca buradan devam edecek...
```

# Döngü 1: **while** döngüsü

```
// 100 defa "Merhaba Dunya"  
// yazdırma  
int sayac = 0;  
  
while (sayac < 100) {  
    //Döngünün Gövdesi  
    System.out.println("Mer"  
        + "haba Dunya");  
    sayac++;  
}
```



**while** döngüsü akış diyagramı



Soldaki kodların akış diyagramı

# Döngü 1: `while` döngüsü

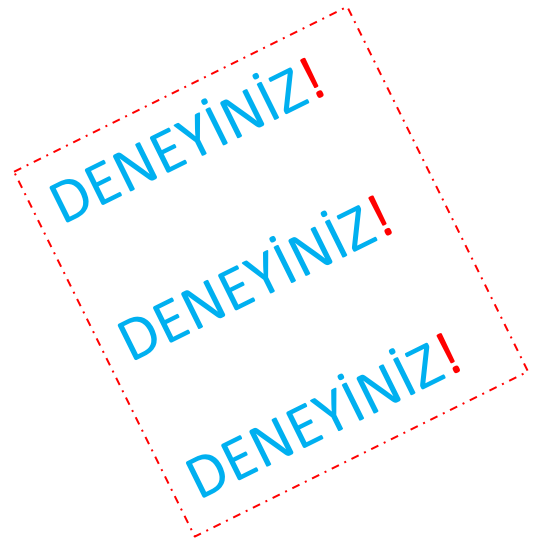
```
1 int top = 0, i = 1;
2
3 while (i < 10) {
4     top = top + i;
5     i++;
6 }
7 System.out.println("Top: " + top);
// 1 + 2 + ... + 8 + 9 = ?
// Top: 45'tir.
```

```
1 başlangıç değişkenleri tanımlama
2
3 while (döngü kriteri) {
4     // Döngünün Gövdesi
5     // değişken artırma/azaltma
6 }
7 // döngü bitince buradan devam edecek...
```

# Örnek 1:

Takılan programları  
**CTRL+C** ile iptal  
edebilirsiniz.

```
int top = 0, i = 1;  
while (i < 10) {  
    top = top + i;  
    System.out.println("Toplam: " + top);  
}
```



## Örnek 2:

// 1

```
int sayac = 0;
while (sayac < 100) {
    System.out.println("Merhaba Dunya");
    sayac++;
}
```

// 2

```
int sayac = 0;
while (sayac <= 100) {
    System.out.println("Merhaba Dunya");
    sayac++;
}
```

// 3

```
int sayac = 1;
while (sayac < 100) {
    System.out.println("Merhaba Dunya");
    sayac++;
}
```

// 4

```
int sayac = 1;
while (sayac <= 100) {
    System.out.println("Merhaba Dunya");
    sayac++;
}
```

# Örnek 3:

// 5

```
int sayac = 0;
while (sayac < 101) {
    System.out.println("Merhaba Dunya");
    sayac++;
}
```

// 6

```
int sayac = 0;
while (sayac <= 101) {
    System.out.println("Merhaba Dunya");
    sayac++;
}
```

// 7

```
int sayac = 1;
while (sayac < 101) {
    System.out.println("Merhaba Dunya");
    sayac++;
}
```

// 8

```
int sayac = 1;
while (sayac <= 101) {
    System.out.println("Merhaba Dunya");
    sayac++;
}
```

## Örnek 4:

```
1 //Tekrarli Toplama Testi
2 import java.util.Scanner;
3
4 public class TekrarliToplamaTesti {
5     public static void main(String [] args) {
6         int sayi1 = (int) (Math.random() * 10);
7         int sayi2 = (int) (Math.random() * 10);
8
9         // klavyeden giriş
10        Scanner giris = new Scanner(System.in);
11
12        System.out.print(sayi1 + " + " + sayi2 + " sonucu nedir? ");
13        int cevap = giris.nextInt();
14
15        while (sayi1 + sayi2 != cevap) {
16            System.out.print("Yanlis cevap! Tekrar deneyin. "
17                + sayi1 + " + " + sayi2 + " sonucu nedir? ");
18            cevap = giris.nextInt();
19        }
20        System.out.print("Tebrikler, bildin!");
21    }
22 }
```

```
C:\Program Files\Java\jdk-16.0.1\bin\yeni>java TekrarliToplamaTesti
4 + 8 sonucu nedir? 15
Yanlis cevap! Tekrar deneyin. 4 + 8 sonucu nedir? 14
Yanlis cevap! Tekrar deneyin. 4 + 8 sonucu nedir? 13
Yanlis cevap! Tekrar deneyin. 4 + 8 sonucu nedir? 12
Tebrikler, bildin!
```

C:\Program Files\Java\jdk-16.0.1\bin\yeni>java SayiTahmini  
0 ile 100 arasında sihirli bir sayi tahmin ediniz.

## Örnek 5:

Tahmin ediniz: 50  
Tahmininiz çok düşük

Tahmin ediniz: 80  
Tahmininiz çok yüksek

Tahmin ediniz: 70  
Tahmininiz çok yüksek

Tahmin ediniz: 60  
Evet, sayi: 60

```
1 //SayiTahmini
2 import java.util.Scanner;
3
4 public class SayiTahmini {
5     public static void main(String [] args) {
6         // tahmin edilecek sayiyi üretme
7         int sayi = (int) (Math.random() * 101);
8
9         // klavyeden giriş
10        Scanner giris = new Scanner(System.in);
11
12        // kullanıcının tahmin etmesi isteniyor
13        System.out.println("0 ile 100 arasında sihirli "
14        + "bir sayi tahmin ediniz.");
15
16        int tahmin = -1;
17        while (tahmin != sayi) {
18            System.out.print("\nTahmin ediniz: ");
19            tahmin = giris.nextInt();
20
21            if (tahmin == sayi)
22                System.out.println("Evet, sayi: " + sayi);
23            else if (tahmin > sayi)
24                System.out.println("Tahmininiz çok yüksek");
25            else
26                System.out.println("Tahmininiz çok düşük");
27        } // while end
28    } // main end
29 } // class end
```

## Örnek 6.a:

```
1 //Gozcu Deger
2 import java.util.Scanner;
3
4 public class GozcuDeger {
5     public static void main(String [] args) {
6         Scanner giris = new Scanner(System.in);
7
8         System.out.print("Bir tam sayi giriniz "
9             + "(0 girilirse giris sonlanir!): ");
10        int veri = giris.nextInt();
11
12        int toplam = 0;
13        while (veri != 0) {
14            toplam += veri;
15
16            System.out.print("Bir tam sayi giriniz "
17                + "(0 girilirse giris sonlanir!): ");
18            veri = giris.nextInt();
19        }
20
21        System.out.print("Toplam: " + toplam);
22    }
23 }
```

```
i>javac GozcuDeger.java
```

```
i>java GozcuDeger
```

```
sonlanir!): 5
```

```
sonlanir!): 6
```

```
sonlanir!): 7
```

```
sonlanir!): 8
```

```
Bir tam sayi giriniz (0 girilirse giris sonlanir!): 10
```

```
Bir tam sayi giriniz (0 girilirse giris sonlanir!): 20
```

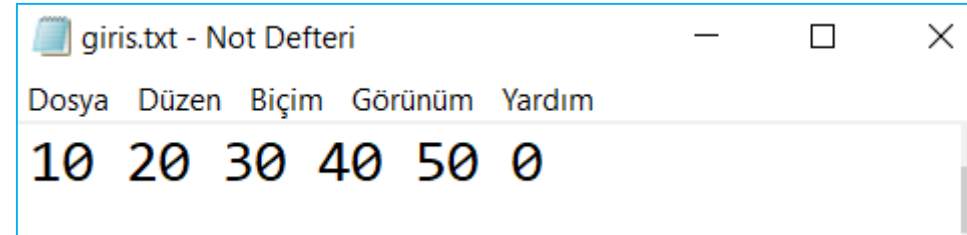
```
Bir tam sayi giriniz (0 girilirse giris sonlanir!): 30
```

```
Bir tam sayi giriniz (0 girilirse giris sonlanir!): 0
```

```
Toplam: 86
```

# Örnek 6.b:

```
java GozcuDeger < giris.txt
```



```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java GozcuDeger < giris.txt
Bir tam sayi giriniz (0 girilirse giris sonlanir!): Bir tam sayi giriniz
(0 girilirse giris sonlanir!): Bir tam sayi giriniz (0 girilirse giris
sonlanir!): Bir tam sayi giriniz (0 girilirse giris sonlanir!): Bir tam
sayi giriniz (0 girilirse giris sonlanir!): Bir tam sayi giriniz (0 giri
lirse giris sonlanir!): Toplam: 150
C:\Program Files\Java\jdk-15.0.1\bin\yeni>
```

```
java GozcuDeger > cikis.txt
```

```
java GozcuDeger < giris.txt > cikis.txt
```

# Döngü 2: **do-while** döngüsü

başlangıç değişkeni tanımlama

```
do{  
    // Döngünün Gövdesi  
    // Koşul true ise burası yapılır, döngü devam eder  
    // yapılacaklar 1  
    // yapılacaklar 2 ...  
    // değişken artırma ya da azaltma  
} while (döngüyü devam ettirme koşulu) ;  
  
// Koşul false ise döngüden çıkılır, buradan devam eder...
```



**do-while** döngüsü akış diyagramı

# Döngü 2: **do-while** döngüsü

**do-while** döngüsü  
**while** döngüsünün  
bir türevidir.

**while** döngüsünden  
farkı ilk çalıştığında  
koşula bakmaksızın  
bir kez çalışıp daha  
sonra döngünün  
devam şartını  
(koşulunu) kontrol  
etmesidir.



**while** döngüsü



**do-while** döngüsü

# Örnek 7:

Örnek  
6.a'daki  
program  
kodunu  
do-while  
kullanarak  
yeniden  
yazıp  
çalıştırınız.

```
1 //Gozcu Deger
2 import java.util.Scanner;
3
4 public class GozcuDeger {
5     public static void main(String [] args) {
6         Scanner giris = new Scanner(System.in);
7
8         System.out.print("Bir tam sayi giriniz "
9             + "(0 girilirse giris sonlanir!): ");
10        int veri = giris.nextInt();
11
12        int toplam = 0;
13        while (veri != 0) {
14            toplam += veri;
15
16            System.out.print("Bir tam sayi giriniz "
17                + "(0 girilirse giris sonlanir!): ");
18            veri = giris.nextInt();
19        }
20
21        System.out.print("Toplam: " + toplam);
22    }
23 }
```

```
i>javac GozcuDeger.java
```

```
i>java GozcuDeger
```

```
sonlanir!): 5
```

```
sonlanir!): 6
```

```
sonlanir!): 7
```

```
sonlanir!): 8
```

```
Bir tam sayi giriniz (0 girilirse giris sonlanir!): 10
```

```
Bir tam sayi giriniz (0 girilirse giris sonlanir!): 20
```

```
Bir tam sayi giriniz (0 girilirse giris sonlanir!): 30
```

```
Bir tam sayi giriniz (0 girilirse giris sonlanir!): 0
```

```
Toplam: 86
```

# Döngü 3: **for** döngüsü

**while** döngüsü ile yapılabilen her şey **for** döngüsü ile de yapılabilir.

**for** döngüsünün kısa bir yazım şekli vardır.

// for döngüsü, döngü yapısı (sözdizimi)

```
for (başlangıç değeri ; döngüyü devam ettirme koşulu ; artırma/azaltma ) {  
    // Döngünün Gövdesi  
    // yapılacaklar 1  
    // yapılacaklar 2 ...  
}  
// döngüden çıkınca buradan devam edecek...
```

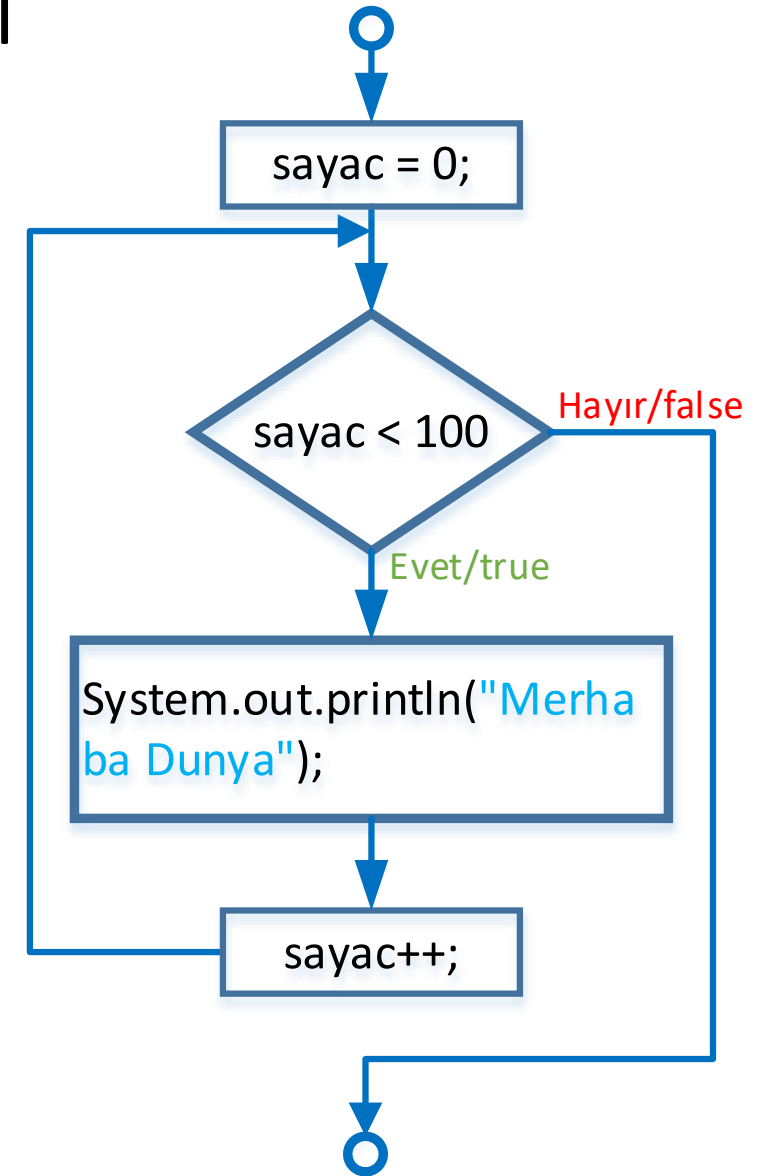


**for** döngüsü akış diyagramı

# for döngüsü ve akış diyagramı

```
// 100 defa "Merhaba Dünya" yazdırma
```

```
for (int sayac = 0; sayac < 100 ; sayac++) {  
    //Döngünün Gövdesi  
    System.out.println("Merhaba Dünya");  
}
```



# for döngüsü vs while döngüsü

```
// 100 defa "Merhaba Dunya" yazdırma

for (int sayac = 0; sayac < 100 ; sayac++) {
    //Döngünün Gövdesi
    System.out.println("Merhaba Dunya");
}
```

```
// 100 defa "Merhaba Dunya" yazdırma
int sayac = 0;

while (sayac < 100) {
    //Döngünün Gövdesi
    System.out.println("Merhaba Dunya");
    sayac++;
}
```

# Hangi döngüyü kullanmalıyım?

`while` ve `for` önce şartı kontrol eder, sonra çalışır.

## 1. `while` döngüsü

ne zaman sonlanacağı belirlenmemiş (belli olmayan) durumlar için kullanılması daha uygundur.

## 2. `do-while` döngüsü

başlangıçta bir kez çalışan ve ne zaman sonlanacağı belirlenmemiş (belli olmayan) durumlar için kullanılması daha uygundur.

## 3. `for` döngüsü

başlangıcı ve sonu belirli (belli) durumlar için kullanılması daha uygundur.

# Sonsuz döngü

```
for ( ; ; ) {  
    //Bir şeyler yap...  
}
```

≡

```
for ( ; true ; ) {  
    //Bir şeyler yap...  
}
```

≡

```
while ( true ) {  
    //Bir şeyler yap...  
}  
// tercih edilen
```

# İç içe döngüler

```
1 //Carpim Tablosu
2 import java.util.Scanner;
3
4 public class CarpimTablosu {
5     public static void main(String [] args) {
6         // tablonun başlığı
7         System.out.println("                Carpim Tablosu");
8
9         // sayı başlığı
10        System.out.print("        ");
11        for (int j = 1; j <= 9; j++)
12            System.out.print("    " + j);
13
14        System.out.println("\n-----");
15
16        // tablonun gövdesi
17        for (int i = 1; i <= 9; i++) {
18            System.out.print(i + " | ");
19            for (int j = 1; j <= 9; j++) {
20                // carpimlar ve duzenli yerlestirme
21                System.out.printf("%4d", i * j);
22            }
23            System.out.println();
24        }
25    }
26 }
```

| Carpim Tablosu |   |    |    |    |    |    |    |    |    |
|----------------|---|----|----|----|----|----|----|----|----|
|                | 1 | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  |
| 1              | 1 | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  |
| 2              | 2 | 4  | 6  | 8  | 10 | 12 | 14 | 16 | 18 |
| 3              | 3 | 6  | 9  | 12 | 15 | 18 | 21 | 24 | 27 |
| 4              | 4 | 8  | 12 | 16 | 20 | 24 | 28 | 32 | 36 |
| 5              | 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 |
| 6              | 6 | 12 | 18 | 24 | 30 | 36 | 42 | 48 | 54 |
| 7              | 7 | 14 | 21 | 28 | 35 | 42 | 49 | 56 | 63 |
| 8              | 8 | 16 | 24 | 32 | 40 | 48 | 56 | 64 | 72 |
| 9              | 9 | 18 | 27 | 36 | 45 | 54 | 63 | 72 | 81 |

# İç içe döngüler

```
1 //Carpim Tablosu Indis
2 import java.util.Scanner;
3
4 public class CarpimTablosuIndis {
5     public static void main(String [] args) {
6         // tablonun başlığı
7         System.out.println("                Carpim Tablosu Indisleri");
8
9         // sayı başlığı
10        System.out.print("                ");
11        for (int j = 1; j <= 9; j++)
12            System.out.print("        " + j);
13
14        System.out.println("\n-----");
15
16        // tablonun gövdesi
17        for (int i = 1; i <= 9; i++) {
18            System.out.print(i + " | ");
19            for (int j = 1; j <= 9; j++) {
20                // indisler ve duzenli yerlestirme
21                System.out.printf("    %d%d", i, j);
22            }
23            System.out.println();
24        }
25    }
26 }
```

| Carpim Tablosu Indisleri |    |    |    |    |    |    |    |    |    |
|--------------------------|----|----|----|----|----|----|----|----|----|
|                          | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  |
| 1                        | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
| 2                        | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 |
| 3                        | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 |
| 4                        | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 |
| 5                        | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 |
| 6                        | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 |
| 7                        | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 |
| 8                        | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 |
| 9                        | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 |

# İç içe döngüler

Bu döngüde println() kaç defa çalışır?

```
for (int i = 0; i < 10; i++)  
    for (int j = 0; j < i; j++)  
        System.out.println(i * j)
```

Bu döngü kaç defa çalışır?

```
for (int i = 0; i < 10000; i++)  
    for (int j = 0; j < 10000; j++)  
        for (int k = 0; k < 10000; k++)
```

# Örnek 8:

```
1 //IciceDonguOrnek
2 import java.util.Scanner;
3
4 public class IciceDonguOrnek {
5     public static void main(String [] args) {
6         for (int i = 1; i < 5; i++) {
7             int j = 0;
8             while (j < i) {
9                 System.out.print(j + " ");
10                j++;
11            }
12            System.out.println();
13        }
14    }
15 }
```

```
0
0 1
0 1 2
0 1 2 3
```

```
1 //IciceDonguOrnek2
2 import java.util.Scanner;
3
4 public class IciceDonguOrnek2 {
5     public static void main(String [] args) {
6         for (int i = 4; i > 0; i--) {
7             int j = 0;
8             while (j < i) {
9                 System.out.print(j + " ");
10                j++;
11            }
12            System.out.println();
13        }
14    }
15 }
```

```
0 1 2 3
0 1 2
0 1
0
```

# Örnek 9:

```
1 //IciceDonguOrnek3
2 import java.util.Scanner;
3
4 public class IciceDonguOrnek3 {
5     public static void main(String [] args) {
6         int i = 0;
7         while (i < 5) {
8             for (int j = i; j > 1; j--){
9                 System.out.print(j + "****");
10            }
11            System.out.println("-" + i + "-");
12            i++;
13        }
14    }
15 }
```

```
-0-
-1-
2***-2-
3***2***-3-
4***3***2***-4-
```

# Örnek 10:

```
1 //IciceDonguOrnek4
2 import java.util.Scanner;
3
4 public class IciceDonguOrnek4 {
5     public static void main(String [] args) {
6         int i = 1;
7         while (i < 8) {
8             for (int j = i; j > 0; j--){
9                 System.out.print(j + "***");
10            }
11            System.out.println("-" + i + "-");
12            i++;
13        }
14    }
15 }
```

```
1***-1-
2***1***-2-
3***2***1***-3-
4***3***2***1***-4-
5***4***3***2***1***-5-
6***5***4***3***2***1***-6-
7***6***5***4***3***2***1***-7-
```

```
1 //IciceDonguOrnek4
2 import java.util.Scanner;
3
4 public class IciceDonguOrnek4 {
5     public static void main(String [] args) {
6         int i = 1;
7         while (i < 8) {
8             for (int j = i; j > 0; j--){
9                 System.out.print(j + " ");
10            }
11            System.out.println();
12            i++;
13        }
14    }
15 }
```

```
1
2 1
3 2 1
4 3 2 1
5 4 3 2 1
6 5 4 3 2 1
7 6 5 4 3 2 1
```

# Örnek 11:

```
1 //En Buyuk Ortak Bolen
2 import java.util.Scanner;
3
4 public class EnBuyukOrtakBolen {
5     public static void main(String [] args) {
6         Scanner giris = new Scanner(System.in);
7
8         System.out.print("Ilk tam sayiyi giriniz: ");
9         int sayi1 = giris.nextInt();
10        System.out.print("Ikinci tam sayiyi giriniz: ");
11        int sayi2 = giris.nextInt();
12
13        int ebob = 0; // baslangic ebob
14        int k = 2;    // potansiyel ebob
15        while (k <= sayi1 && k <= sayi2) {
16            if (sayi1 % k == 0 && sayi2 % k == 0)
17                ebob = k; // ebob guncelle
18            k++;
19        }
20
21        System.out.println(sayi1 + " ve " + sayi2 + " icin "
22        + "En buyuk ortak bolen: " + ebob);
23    }
24 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java EnBuyukOrtakBolen
```

```
Ilk tam sayiyi giriniz: 2550
```

```
Ikinci tam sayiyi giriniz: 150
```

```
2550 ve 150 icin En buyuk ortak bolen: 150
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java EnBuyukOrtakBolen
```

```
Ilk tam sayiyi giriniz: 2525
```

```
Ikinci tam sayiyi giriniz: 125
```

```
2525 ve 125 icin En buyuk ortak bolen: 25
```

# break

```
1  //Break Testi
2  import java.util.Scanner;
3
4  public class BreakTesti {
5      public static void main(String [] args) {
6          int top = 0;
7          int sayi = 0;
8
9          while (sayi < 20) {
10             sayi++;
11             top += sayi;
12             if (top >= 100)
13                 break;
14         }
15         System.out.println("Sayi: " + sayi);
16         System.out.println("Top: " + top);
17     }
18 }
```

C:\Program Files\Java\jdk-15.0.1\bin\yeni>java BreakTesti  
Sayi: 14  
Top: 105

# continue

```
1 //Continue Testi
2 import java.util.Scanner;
3
4 public class ContinueTesti {
5     public static void main(String [] args) {
6         int top = 0;
7         int sayi = 0;
8
9         while (sayi < 20) {
10             sayi++;
11             if (sayi == 10 || sayi == 11)
12                 continue;
13             top += sayi;
14         }
15         System.out.println("Sayi: " + sayi);
16         System.out.println("Top: " + top);
17     }
18 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java ContinueTesti
Sayi: 20
Top: 189
```

# Ödev 1.a:

Konsolda alt alta aşağıdaki on bin adet mesajı çıktı veren Java programını **while** döngüsü ile yazınız. (Not: println, print vb metottan 1 tane kullanınız.)

Yazilim Muhendisligi-1

Yazilim Muhendisligi-2

Yazilim Muhendisligi-3

Yazilim Muhendisligi-4

...

Yazilim Muhendisligi-9998

Yazilim Muhendisligi-9999

Yazilim Muhendisligi-10000

# Ödev 1.b:

Konsolda alt alta aşağıdaki on bin adet mesajı çıktı veren Java programını **for** döngüsü ile yazınız. (Not: println, print vb metottan 1 tane kullanınız.)

Yazilim Muhendisligi-1

Yazilim Muhendisligi-2

Yazilim Muhendisligi-3

Yazilim Muhendisligi-4

...

Yazilim Muhendisligi-9998

Yazilim Muhendisligi-9999

Yazilim Muhendisligi-10000

## Ödev 2.a:

Konsolda alt alta aşağıdaki iki farklı mesajı toplamda on bin adet çıktı veren Java programını yazınız.

Yazilim Muhendisligi-1

Cok calismaliyiz cokk-2

Yazilim Muhendisligi-3

Cok calismaliyiz cokk-4

...

Cok calismaliyiz cokk-9998

Yazilim Muhendisligi-9999

Cok calismaliyiz cokk-10000

## Ödev 2.b:

Konsolda alt alta aşağıdaki iki farklı mesajı on bin adet çıktı veren Java programını yazınız. (Not: `println`, `print vb` metottan 1 tane kullanınız.)

Yazilim Muhendisligi-1

Cok calismaliyiz cokk-2

Yazilim Muhendisligi-3

Cok calismaliyiz cokk-4

...

Cok calismaliyiz cokk-9998

Yazilim Muhendisligi-9999

Cok calismaliyiz cokk-10000

# Ödev 3.a:

Bir `isimler.txt` dosyasında satır satır isimler yazmaktadır. Adi ve soyadi arasında 1 boşluk vardır. Bu `isimler.txt` dosyasını giriş olarak alan ve her satırdaki isimleri "SAYIN ADI1 SOYADI1" şekline getirip `SN_ISIMLER.txt` dosyasına kaydeden Java programını yazınız.

(Not: Giriş dosyasında 30 kişinin adı ve soyadı bulunmaktadır.)

`isimler.txt`

```
Adi1 Soyadi1  
Adi2 Soyadi2  
Yaman Akbulut  
Adi3 Soyadi3  
Adi4 Soyadi4  
Adi5 Soyadi5  
...
```

`SN_ISIMLER.txt`

```
SAYIN ADI1 SOYADI1  
SAYIN ADI2 SOYADI2  
SAYIN YAMAN AKBULUT  
SAYIN ADI3 SOYADI3  
SAYIN ADI4 SOYADI4  
SAYIN ADI5 SOYADI5  
...
```

## Ödev 3.b:

Bir **isimler.txt** dosyasında satır satır isimler yazmaktadır. Adi ve soyadi arasında **en az** 1 boşluk vardır, satır başında ve sonunda da boşluklar olabilir. Bu **isimler.txt** dosyasını giriş olarak alan ve her satırdaki isimleri "SAYIN ADI1 SOYADI1" şekline (aralarda sadece 1 boşluk var) getirip **SN\_ISIMLER.txt** dosyasına kaydeden Java programını yazınız. (Not: Giriş dosyasında 30 kişinin adı ve soyadı bulunmaktadır.)

**isimler.txt**

```
Adi1 Soyadi1
Adi2   Soyadi2
    Yaman Akbulut
Adi3 Soyadi3
    Adi4 Soyadi4
Adi5   Soyadi5
...
```

**SN\_ISIMLER.txt**

```
SAYIN ADI1 SOYADI1
SAYIN ADI2 SOYADI2
SAYIN YAMAN AKBULUT
SAYIN ADI3 SOYADI3
SAYIN ADI4 SOYADI4
SAYIN ADI5 SOYADI5
...
```

# Ödev 4:

Örnek  
4'teki  
program  
kodunu  
do-while  
kullanarak  
yeniden  
yazıp  
çalıştırınız.

```
1 //Tekrarli Toplama Testi
2 import java.util.Scanner;
3
4 public class TekrarliToplamaTesti {
5     public static void main(String [] args) {
6         int sayi1 = (int) (Math.random() * 10);
7         int sayi2 = (int) (Math.random() * 10);
8
9         // klavyeden giriş
10        Scanner giris = new Scanner(System.in);
11
12        System.out.print(sayi1 + " + " + sayi2 + " sonucu nedir? ");
13        int cevap = giris.nextInt();
14
15        while (sayi1 + sayi2 != cevap) {
16            System.out.print("Yanlis cevap! Tekrar deneyin. "
17                + sayi1 + " + " + sayi2 + " sonucu nedir? ");
18            cevap = giris.nextInt();
19        }
20        System.out.print("Tebrikler, bildin!");
21    }
22 }
```

C:\Program Files\Java\jdk-15.0.1\bin\yeni>java TekrarliToplamaTesti  
8 + 7 sonucu nedir? 14  
Yanlis cevap! Tekrar deneyin. 8 + 7 sonucu nedir? 16  
Yanlis cevap! Tekrar deneyin. 8 + 7 sonucu nedir? 13  
Yanlis cevap! Tekrar deneyin. 8 + 7 sonucu nedir? 15  
Tebrikler, bildin!

# Ödev 5:

Örnek  
5'teki  
program  
kodunu  
do-while  
kullanarak  
yeniden  
yazıp  
çalıştırınız.

```
Tahmin ediniz: 50
Tahmininiz cok dusuk

Tahmin ediniz: 75
Tahmininiz cok yuksek

Tahmin ediniz: 60
Tahmininiz cok dusuk

Tahmin ediniz: 70
Tahmininiz cok yuksek

Tahmin ediniz: 68
Tahmininiz cok dusuk

Tahmin ediniz: 69
Evet, sayi: 69
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java SayiTahmini
0 ile 100 arasinda sihirli bir sayi tahmin ediniz.
```

```
1 //SayiTahmini
2 import java.util.Scanner;
3
4 public class SayiTahmini {
5     public static void main(String [] args) {
6         // tahmin edilecek sayiyi üretme
7         int sayi = (int)(Math.random() * 101);
8
9         // klavyeden giriş
10        Scanner giris = new Scanner(System.in);
11
12        // kullanıcının tahmin etmesi isteniyor
13        System.out.println("0 ile 100 arasinda sihirli "
14        + "bir sayi tahmin ediniz.");
15
16        int tahmin = -1;
17        while (tahmin != sayi) {
18            System.out.print("\nTahmin ediniz: ");
19            tahmin = giris.nextInt();
20
21            if (tahmin == sayi)
22                System.out.println("Evet, sayi: " + sayi);
23            else if (tahmin > sayi)
24                System.out.println("Tahmininiz cok yuksek");
25            else
26                System.out.println("Tahmininiz cok dusuk");
27        } // while end
28    } // main end
29 } // class end
```

# Ödev 6:

Girişlerin 2 3 4 5 0  
olduğunu varsayınız.

Buna göre yandaki  
kodun çıktısı nedir?

```
import java.util.Scanner;

public class Test {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        int number, max;
        number = input.nextInt();
        max = number;

        while (number != 0) {
            number = input.nextInt();
            if (number > max)
                max = number;
        }

        System.out.println("max is " + max);
        System.out.println("number " + number);
    }
}
```

# Ödev 7:

Aşağıdaki çıktıyı veren Java programını yazınız.

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java TabloIndisToplami
      Tablo Indisleri ve Indis Toplamlari
      |      1      2      3      4      5      6      7      8      9
----|-----
1 | 11- 2 21- 3 31- 4 41- 5 51- 6 61- 7 71- 8 81- 9 91-10
2 | 12- 3 22- 4 32- 5 42- 6 52- 7 62- 8 72- 9 82-10 92-11
3 | 13- 4 23- 5 33- 6 43- 7 53- 8 63- 9 73-10 83-11 93-12
4 | 14- 5 24- 6 34- 7 44- 8 54- 9 64-10 74-11 84-12 94-13
5 | 15- 6 25- 7 35- 8 45- 9 55-10 65-11 75-12 85-13 95-14
6 | 16- 7 26- 8 36- 9 46-10 56-11 66-12 76-13 86-14 96-15
7 | 17- 8 27- 9 37-10 47-11 57-12 67-13 77-14 87-15 97-16
8 | 18- 9 28-10 38-11 48-12 58-13 68-14 78-15 88-16 98-17
9 | 19-10 29-11 39-12 49-13 59-14 69-15 79-16 89-17 99-18
```

# 12. Ders: JAVA Programlama Dili

## Diziler (tek boyutlu, çok boyutlu)

Fırat Üniversitesi Teknoloji Fakültesi Yazılım Mühendisliği Bölümü

YMH111 Algoritma ve Programlama-I

Doç. Dr. Yaman Akbulut

# JAVA Programlama Dili

- <http://www.kriptarium.com/algorithm.html> (Yardımcı kaynak)
- JAVA Programlama Dili (Ders: 37, 42-47) video. (**Mutlaka İzleyiniz!**)
  - Ders 42: Diziler (izle)
  - Ders 43: Tek Boyutlu Diziler (izle)
  - Ders 44: Bir Dizideki En Küçük Elemanı Bulma (izle)
  - Ders 45: Çok Boyutlu Diziler (izle)
  - Ders 46: Çok Boyutlu Dizi Uygulaması İki Matrisin Toplamı (izle)
  - Ders 47: Yorum/Açıklama Satırları (izle)
  - Ders 37: ForEach Döngü Yapısının Kullanımı (izle)

# Java Keywords

abstract

assert

boolean

break

byte

case

catch

char

class

const

continue

default

do

double

else

enum

extends

final

finally

float

for

goto

if

implements

import

instanceof

int

interface

long

native

new

package

private

protected

public

return

short

static

strictfp

super

switch

synchronized

this

throw

throws

transient

try

void

volatile

while

# Diziler

sayi0, sayi1, sayi2, sayi3, sayi4, sayi5, sayi6, sayi7, sayi8, sayi9

```
double[] sayiListe = new double[10];
```

sayiListe[0], sayiListe[1], sayiListe[2], ... , sayiListe[9]

a[0][0] a[0][1] a[0][2]

a[1][0] a[1][1] a[1][2]

a[2][0] a[2][1] a[2][2]

# Tek boyutlu dizi tanımlama

```
elemanTipi[] dizi = new elemanTipi[diziBoyutu];
```

III

```
elemanTipi dizi[] = new elemanTipi[diziBoyutu];
```

Dizi indeksi  
tam sayıdır ve  
0'dan başlar.

Diziler doğal  
yapıları gereği  
for döngüsüne  
uygundur.

### sayiListe

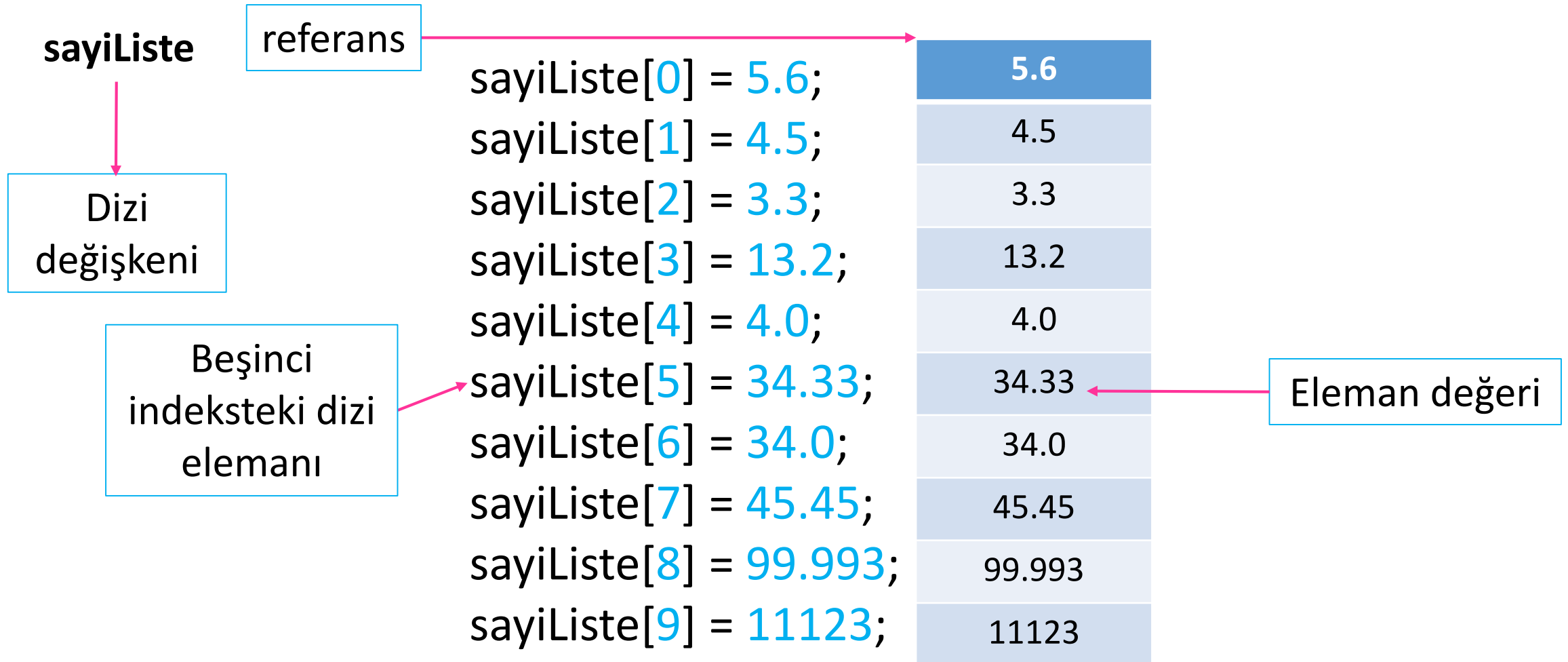
```
sayiListe[0] = 5.6;  
sayiListe[1] = 4.5;  
sayiListe[2] = 3.3;  
sayiListe[3] = 13.2;  
sayiListe[4] = 4.0;  
sayiListe[5] = 34.33;  
sayiListe[6] = 34.0;  
sayiListe[7] = 45.45;  
sayiListe[8] = 99.993;  
sayiListe[9] = 11123;
```

Diziler tek tip  
elemanlardan  
oluşur.

Dizilerin  
boyutları  
belirlendikten  
sonra  
değiştirilemez.

# Tek boyutlu dizi oluşturma

```
double[] sayiListe = new double[10];
```



# Dizi boyutu ve varsayılan değerler

Dizi oluşturulurken boyutun belirlenmesi gerek:

```
double[] sayiListe = new double[10];
```

Mevcut dizinin boyutunu elde etmek için:

```
sayiListe.length;
```

Yeni bir dizi oluşturulduğunda varsayılan değerler olarak:

1. Numerik tipler için 0
2. Karakterler için \u0000
3. Boolean tipler için false

Mevcut dizinin tüm elemanlarına indeksle ulaşmak için:

indeks  
0'dan sayiListe.length-1'e  
değişir.

# Dizi elemanı

Dizi elemanına erişmek için indeks kullanılır:

```
dizi[indeks];
```

```
int a = sayiListe[3];
```

Dizi elemanına değer atama için indeks kullanılır:

```
dizi[indeks] = değer;
```

```
sayiListe[3] = 10;
```

|        |
|--------|
| 5.6    |
| 4.5    |
| 3.3    |
| 13.2   |
| 4.0    |
| 34.33  |
| 34.0   |
| 45.45  |
| 99.993 |
| 11123  |

# Diziyi tanımlama, oluşturma ve başlangıç değerler

```
double[] benimDizi = {1.9, 2.9, 3.4, 3.5};
```

III

```
double[] benimDizi = new double[4];
```

```
benimDizi[0] = 1.9;
```

```
benimDizi[1] = 2.9;
```

```
benimDizi[2] = 3.4;
```

```
benimDizi[3] = 3.5;
```

# Dizilerle işlemler (diziye eleman değeri girişi)

Diziler boyutu belli olduğundan doğal yapıları **for** döngüsüne uygundurlar.

```
int[]  
double[]  
char[]  
String[]
```

# Örnek 1:

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java DiziDegerGirisi
Lutfen 5 sayi giriniz: 1,2 3,5 10 20 100
1.2 3.5 10.0 20.0 100.0
```

```
1 //Dizi deger girisi
2 import java.util.Scanner;
3
4 public class DiziDegerGirisi {
5     public static void main(String [] args) {
6         double[] benimDizi = new double[5];
7         Scanner giris = new Scanner(System.in);
8
9         System.out.print("Lutfen " + benimDizi.length
10 + " sayi giriniz: ");
11         for (int i = 0; i < benimDizi.length; i++){
12             benimDizi[i] = giris.nextDouble();
13         }
14
15         for (int i = 0; i < benimDizi.length; i++)
16             System.out.print(benimDizi[i] + " ");
17     }
18 }
```

```
Lutfen 5 sayi giriniz: 1,2
3,5
10
20
100
1.2 3.5 10.0 20.0 100.0
```

## Örnek 2:

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java DiziDegerGirisi2
```

```
benimDizi[0]: 1,2
```

```
benimDizi[1]: 3,5
```

```
benimDizi[2]: 10
```

```
benimDizi[3]: 20
```

```
benimDizi[4]: 100
```

```
benimDizi[0] = 1,2
```

```
benimDizi[1] = 3,5
```

```
benimDizi[2] = 10,0
```

```
benimDizi[3] = 20,0
```

```
benimDizi[4] = 100,0
```

```
1 //Dizi deger girisi 2
2 import java.util.Scanner;
3
4 public class DiziDegerGirisi2 {
5     public static void main(String [] args) {
6         double[] benimDizi = new double[5];
7         Scanner giris = new Scanner(System.in);
8
9         for (int i = 0; i < benimDizi.length; i++) {
10             System.out.printf("benimDizi[%d]: ", i);
11             benimDizi[i] = giris.nextDouble();
12         }
13
14         for (int i = 0; i < benimDizi.length; i++)
15             System.out.printf("benimDizi[%d] = %5.1f"
16                               + "\n", i, benimDizi[i]);
17     }
18 }
```

## Örnek 3:

```
1 //Dizi deger girisi 3
2 import java.util.Scanner;
3
4 public class DiziDegerGirisi3 {
5     public static void main(String [] args) {
6         double[] benimDizi = new double[5];
7         Scanner giris = new Scanner(System.in);
8
9         for (int i = 0; i < benimDizi.length; i++){
10             System.out.printf("benimDizi[%d]: ",i);
11             benimDizi[i] = giris.nextDouble();
12         }
13
14         System.out.print("benimDizi = {");
15         for (int i = 0; i < benimDizi.length; i++)
16             System.out.printf(" %5.1f",benimDizi[i]);
17
18         System.out.print("}");
19     }
20 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java DiziDegerGirisi3
benimDizi[0]: 1,2
benimDizi[1]: 3,5
benimDizi[2]: 10
benimDizi[3]: 20
benimDizi[4]: 100
benimDizi = { 1,2 3,5 10,0 20,0 100,0}
```

# Dizilerle işlemler (char)

```
char[] sehir = {'I', 's', 't', 'a', 'n', 'b', 'u', 'I'};  
System.out.println(sehir);
```

## Örnek 4:

```
1 //Dizi deger girisi 4
2 import java.util.Scanner;
3
4 public class DiziDegerGirisi4 {
5     public static void main(String [] args) {
6         char[] sehir = {'I', 's', 't', 'a', 'n', 'b', 'u', 'l'};
7         System.out.println(sehir);
8     }
9 }
```

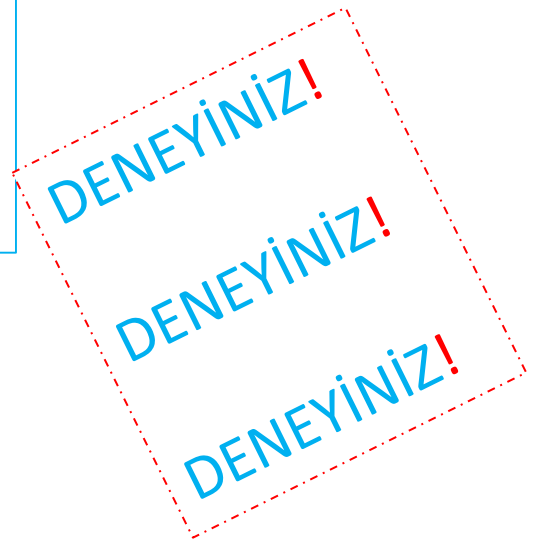
```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java DiziDegerGirisi4
Istanbul
```

# Dizilerle işlemler (dizi elemanlarını toplama)

```
double toplam = 0;
```

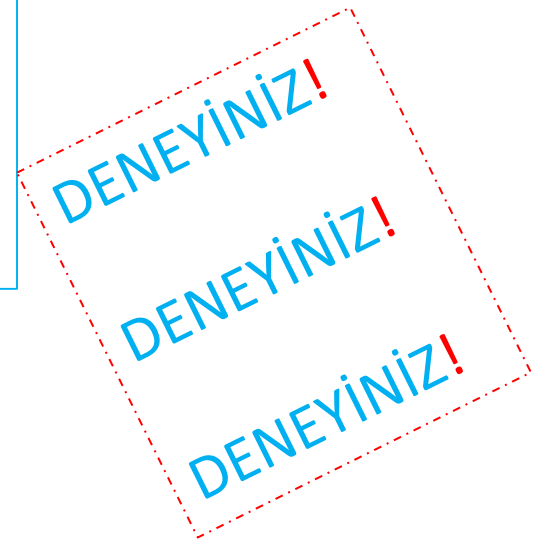
```
for (int i = 0; i < benimDizi.length; i++)  
    toplam += benimDizi[i];
```

```
System.out.println("sonuc: " + toplam);
```



# Dizilerle işlemler (dizide en büyük bulma)

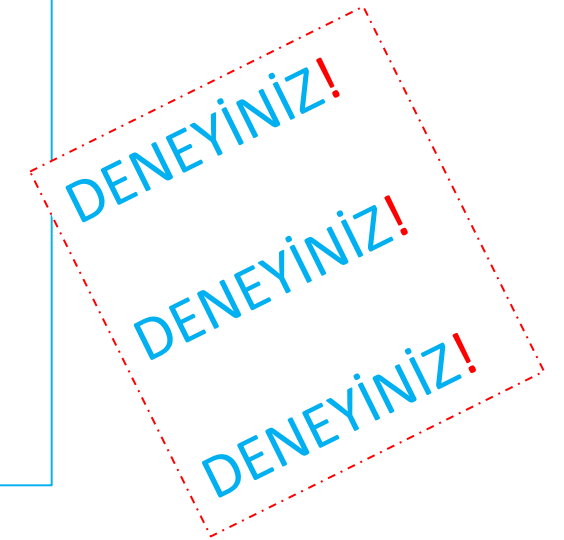
```
double enBuyuk = benimDizi[0];  
  
for (int i = 1; i < benimDizi.length; i++) {  
    if (benimDizi[i] > enBuyuk)  
        enBuyuk = benimDizi[i];  
}
```



# Dizilerle işlemler

(dizide en büyük eleman, en küçük indeks bulma)

```
double enBuyuk = benimDizi[0];  
int enBuyukIndeks = 0;  
  
for (int i = 1; i < benimDizi.length; i++){  
    if (benimDizi[i] > enBuyuk){  
        enBuyuk = benimDizi[i];  
        enBuyukIndeks = i;  
    }  
}
```



# Dizilerle işlemler (String)

```
String[] aylar = {"Ocak", "Subat", "Mart",  
"Nisan", "Mayis", "Haziran", "Temmuz",  
"Agustos", "Eylul", "Ekim", "Kasim", "Aralik"};  
  
System.out.print("Bir ay no giriniz (1-12): ");  
int ayNo = giris.nextInt();  
System.out.println("Ay: " + aylar[ayNo - 1]);
```

```
System.out.print("Bir ay no giriniz (1-12): ");  
int ayNo = giris.nextInt();  
  
if (ayNo == 1)  
    System.out.println("Ay: Ocak");  
else if (ayNo == 2)  
    System.out.println("Ay: Subat");  
else if (ayNo == 3)  
    System.out.println("Ay: Mart");  
...  
...  
else if (ayNo == 11)  
    System.out.println("Ay: Kasim");  
else  
    System.out.println("Ay: Aralik");
```

## Örnek 5:

```
1 //Dizi deger girisi 5
2 import java.util.Scanner;
3
4 public class DiziDegerGirisi5 {
5     public static void main(String [] args) {
6         Scanner giris = new Scanner(System.in);
7         String[] aylar = {"Ocak", "Subat", "Mart",
8             "Nisan", "Mayis", "Haziran", "Temmuz", "Agustos",
9             "Eylul", "Ekim", "Kasim", "Aralik"};
10
11         System.out.print("Bir ay no giriniz (1-12): ");
12         int ayNo = giris.nextInt();
13         System.out.println("Ay: " + aylar[ayNo - 1]);
14     }
15 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\yeni>java DiziDegerGirisi5
Bir ay no giriniz (1-12): 5
Ay: Mayis

C:\Program Files\Java\jdk-15.0.1\bin\yeni>java DiziDegerGirisi5
Bir ay no giriniz (1-12): 11
Ay: Kasim
```

# Foreach döngüsü (for)

Java, indis kullanmadan dizinin tüm elemanlarına erişebilen **foreach** döngüsü olarak bilinen döngü çeşidini destekler ve bunu **for** döngüsü kullanarak gerçekleştirir.

```
for (double e: benimDizi){  
    System.out.println(e);  
}
```

# Örnek 6:

```
C:\Program Files\Java\jdk-16.0.1\bin\yeni>java Listforeach
```

Ocak

Subat

Mart

Nisan

Mayis

Haziran

Temmuz

Agustos

Eylul

Ekim

Kasim

Aralik

```
1  // Liste foreach
2  import java.util.Scanner;
3
4  public class Listforeach {
5      public static void main(String [] args) {
6          String[] aylar = {"Ocak", "Subat", "Mart",
7                          "Nisan", "Mayis", "Haziran", "Temmuz", "Agustos",
8                          "Eylul", "Ekim", "Kasim", "Aralik"};
9
10         for (String e: aylar){
11             System.out.println(e);
12         }
13     }
14 }
```

```
C:\Program Files\Java\jdk-16.0.1\bin\yeni>java Listforeach
```

Ocak Subat Mart Nisan Mayis Haziran Temmuz Agustos Eylul Ekim Kasim Aralik

## Örnek 7:

```
1 public class KartDestesi {
2     public static void main(String[] args) {
3         int[] deste = new int[52];
4         String[] gruplar = {"Maca", "Kupa", "Karo", "Sinek"};
5         String[] siralar = {"As", "2", "3", "4", "5", "6",
6             "7", "8", "9", "10", "Joker", "Kiz", "Papaz"};
7
8         // kartlarin baslangici
9         for (int i = 0; i < deste.length; i++)
10             deste[i] = i;
11
12        // kartlari karistir
13        for (int i = 0; i < deste.length; i++) {
14            // restgele bir indeks olustur
15            int index = (int) (Math.random() * deste.length);
16            int temp = deste[i];
17            deste[i] = deste[index];
18            deste[index] = temp;
19        }
20
21        // ilk dort karti goster
22        for (int i = 0; i < 4; i++) {
23            String grup = gruplar[deste[i] / 13];
24            String sira = siralar[deste[i] % 13];
25            System.out.println("Kart numarasi " + deste[i] + ": "
26                + sira + " - " + grup);
27        }
28    }
29 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin
\yeni>java KartDestesi
Kart numarasi 3: 4 - Maca
Kart numarasi 44: 6 - Sinek
Kart numarasi 33: 8 - Karo
Kart numarasi 28: 3 - Karo
```

## Örnek 8:

```
1 public class SayilarinAnalizi {
2     public static void main(String[] args) {
3         java.util.Scanner giris = new java.util.Scanner(System.in);
4
5         System.out.print("boyutu giriniz: ");
6         int n = giris.nextInt();
7         double[] sayilar = new double[n];
8         double toplam = 0;
9
10        System.out.print("sayilari giriniz: ");
11        for (int i = 0; i < n; i++) {
12            sayilar[i] = giris.nextDouble();
13            toplam += sayilar[i];
14        }
15
16        double ortalama = toplam / n;
17
18        int sayac = 0; // Ortalamanin ustundeki eleman sayisi
19        for (int i = 0; i < n; i++)
20            if (sayilar[i] > ortalama)
21                sayac++;
22
23        System.out.println("Ortalama: " + ortalama);
24        System.out.println("Ortalamanin ustundeki eleman sayisi: "
25            + sayac);
26    }
27 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin\ye
ni>java SayilarinAnalizi
boyutu giriniz: 10
sayilari giriniz: 1 1 1 1 1 4 4 4 4 4
Ortalama: 2.5
Ortalamanin ustundeki eleman sayisi: 5
```

# Çok boyutlu dizi (matris) tanımlama, oluşturma

```
elemanTipi[][] dizi = new elemanTipi[diziBoyutu][diziBoyutu];
```

III

```
elemanTipi dizi[][] = new elemanTipi[diziBoyutu][diziBoyutu];
```

# Matris tanımlama, oluşturma, değer atama

```
int[][] dizi = {  
    {1, 2, 3},  
    {4, 5, 6},  
    {7, 8, 9},  
    {10, 11, 12}  
};
```

=

```
int[][] dizi = new int[4][3];  
dizi[0][0] = 1;   dizi[0][1] = 2;   dizi[0][2] = 3;  
dizi[1][0] = 4;   dizi[1][1] = 5;   dizi[1][2] = 6;  
dizi[2][0] = 7;   dizi[2][1] = 8;   dizi[2][2] = 9;  
dizi[3][0] = 10;  dizi[3][1] = 11;  dizi[3][2] = 12;
```

# Çok boyutlu dizi `length`

```
int[][] x = {  
    {1, 2, 3},  
    {4, 5, 6},  
    {7, 8, 9},  
    {10, 11, 12}  
};
```

|                           |                      |
|---------------------------|----------------------|
| <code>x.length</code>     | // satır uzunluğu    |
| <code>x[0].length;</code> | // 0. sütun uzunluğu |
| <code>x[1].length;</code> | // 1. sütun uzunluğu |
| <code>x[2].length;</code> | // 2. sütun uzunluğu |
| <code>x[3].length;</code> | // 3. sütun uzunluğu |

# Örnek 9:

Görüntü işleme  
Kriptoloji  
Yapay zeka  
Veri madenciliği  
...

```
1 //Matris Toplami
2 public class MatrisToplami {
3     public static void main(String[] args) {
4         // matris olusturma
5         int[][] a = {{1, 2, 3},{4, 5, 6},{7, 8, 9},{10, 11, 12}};
6         int[][] b = {{1, 2, 3},{4, 5, 6},{7, 8, 9},{10, 11, 12}};
7
8         // toplam sonuc matrisi tanimlama
9         int[][] c = new int[4][3];
10
11        // matris toplami icin ic ice dongu
12        for (int i = 0; i < 4; i++){
13            for (int j = 0; j < 3; j++){
14                c[i][j] = a[i][j] + b[i][j];
15                System.out.print(c[i][j] + " ");
16                //System.out.printf("%2d ",c[i][j]);
17            }
18            System.out.println();
19        }
20    }
21 }
```

```
C:\Program Files\Java\jdk-15.0.1\bin
\yeni>java MatrisToplami
```

```
2 4 6
8 10 12
14 16 18
20 22 24
```

```
C:\Program Files\Java\jdk-15.0.1\bin
\yeni>java MatrisToplami
```

```
2 4 6
8 10 12
14 16 18
20 22 24
```

# java.util.Arrays (Arrays Class)

- Karar yapıları
- Döngüler
- Diziler

Görüntü işleme  
Kriptoloji  
Yapay zeka  
Veri madenciliği  
...

# Ödev 1.a:

Tam sayılardan oluşan 20 elemanlı bir dizideki en küçük elemanı bulan ve konsolda gösteren Java programını yazınız.

# Ödev 1.b:

Kullanıcı tarafından tam sayı değerleri girilen 20 elemanlı bir dizideki en küçük elemanı ve o elemana ait indisi bulup konsolda gösteren Java programını yazınız.

# Ödev 1.c:

Kullanıcı tarafından tam sayı değerleri girilen 20 elemanlı bir diziyi küçükten büyüğe sıralayıp konsolda gösteren Java programını yazınız.

## Ödev 2:

Kullanıcı tarafından tam sayı değerleri girilen 20 elemanlı bir dizinin toplamını ve ortalamasını hesaplayıp konsolda gösteren Java programını yazınız.

# Ödev 3:

Aşağıdaki kodun çıktısı nedir?

```
int x = 30;  
int[] sayilar = new int[x];  
x = 60;  
System.out.println("x: " + x);  
System.out.println("sayilarin boyutu: " + sayilar.length);
```

# Ödev 4:

Aşağıdaki kodun çıktısı nedir?

```
public class Test {  
    public static void main(String[] args) {  
        int[] liste = {1, 2, 3, 4, 5, 6};  
  
        for (int i = 1; i < liste.length; i++)  
            liste[i] = liste[i - 1];  
  
        for (int i = 0; i < liste.length; i++)  
            System.out.print(liste[i] + " ");  
    }  
}
```

## Ödev 5.a:

Aşağıdaki matrisin en küçük elemanını ve o elemana ait indisleri bulup konsolda yazan Java programını yazınız?

|     |    |     |
|-----|----|-----|
| 100 | 20 | 300 |
| 44  | 55 | 666 |
| 7   | 80 | 9   |
| 10  | 11 | 12  |

## Ödev 5.b:

Aşağıdaki matrisin en küçük ve en büyük elemanlarını ve o elemanlara ait indisleri bulup konsolda yazan Java programını yazınız?

|     |    |     |
|-----|----|-----|
| 100 | 20 | 300 |
| 44  | 55 | 666 |
| 7   | 80 | 9   |
| 10  | 11 | 12  |

## Ödev 5.c:

Aşağıdaki matrisi tek boyutlu bir diziye 1) satır satır, 2) sütun sütun ekleyip konsolda yazan Java programını yazınız?

|     |    |     |
|-----|----|-----|
| 100 | 20 | 300 |
| 44  | 55 | 666 |
| 7   | 80 | 9   |
| 10  | 11 | 12  |

## Ödev 5.d:

Aşağıdaki matrisi küçükten büyüğe doğru sıralayıp yine matris formunda konsolda yazan Java programını yazınız?

|     |    |     |
|-----|----|-----|
| 100 | 20 | 300 |
| 44  | 55 | 666 |
| 7   | 80 | 9   |
| 10  | 11 | 12  |

# Ödev 6:

Aşağıdaki **a** matrisinin satır ve sütun toplamalarını döngüyle toplayan ve yeni bir **b** matrisi oluşturup bu değerleri ona atayan ve konsolda gösteren Java programını yazınız?

**a**

|     |    |     |
|-----|----|-----|
| 100 | 20 | 300 |
| 44  | 55 | 666 |
| 7   | 80 | 9   |
| 10  | 11 | 12  |

**b**

|     |     |     |      |
|-----|-----|-----|------|
| 100 | 20  | 300 | 420  |
| 44  | 55  | 666 | 765  |
| 7   | 80  | 9   | 96   |
| 10  | 11  | 12  | 33   |
| 161 | 166 | 987 | 2628 |