

# NESNE YÖNELİMLİ PROGRAMLAMA(Object Oriented Programming/OOP)

**Öğr. Gör. Celil ÖZTÜRK**

Marmara Üniversitesi

Teknik Bilimler Meslek Yüksekokulu

# İçerik

- ✓ Giriş-Çıkış(I/O) İşlemleri Komutları
- ✓ Koşul İfadeleri
- ✓ If-Else
- ✓ Switch Case
- ✓ Ternary(Üçlü Şart Operatörü)
- ✓ Döngüler/Tekrarlı Yapılar

# Giriş Çıkış (I/O) Komutlar

- **Scanner** sınıfının yöntemlerini kullanabilmek için programın başında; **import java.util.Scanner** veya **import java.util.\*** ifadesi kullanılır.

## **Bilgi giriş-çıkışı için genelde kullanılan 3 yöntem:**

- **System.in:** Klavyeden bilgi girişi için kullanılır.
- **System.out:** İşlem sonuçlarını, mesajları ekranda göstermek için kullanılır.
- **System.err:** Hata mesajlarını verir.

# Giriş Çıkış (I/O) Komutlar

## Scanner sınıfı yöntemleri

İşlevi	Yöntemi
Klavyeden girilen ifadeyi tamsayı (integer) türüne çevirir.	<code>nextInt()</code>
Klavyeden girilen ifadeyi kesirli (gerçek) sayı (float) türüne çevirir.	<code>nextFloat()</code>
Klavyeden girilen ifadeyi uzun kesirli (gerçek) sayı (double) türüne çevirir.	<code>nextDouble()</code>
Klavyeden girilen ifadenin ilk kelimesini okur. {boşluk (space) karakterine kadar olan kısım} tarar.	<code>next()</code>
Klavyeden girilen tüm satırı okur.	<code>nextLine()</code>
Tek bir karakteri okur. (Girilen ifadeyi karakter karakter okur.)	<code>findInLine( "." ).charAt(0)</code>

# Giriş Çıkış (I/O) Komutlar

## System.out.print() Deyimi

- Ekranaya yazı yazmak için System.out.println() veya System.out.print() şeklinde bir ifade kullanılır. Bu işlem, System adlı sınıfın out adlı özelliğinin print() veya println() adlı yöntemini çağırarak gerçekleşir. print() deyimi ile mesajlar yan yana yazılırken, println() deyimi ile mesaj yazıldıktan sonra bir alt satıra geçilir. (alt alta yazılır)

Örneğin aşağıdaki kod satırlarının;

```
System.out.print("Marmara");  
System.out.print("Universitesi");  
System.out.println("MYO");  
System.out.println("Istanbul");
```

# Giriş Çıkış (I/O) Komutlar

## `System.out.printf()` Deyimi

Bu deyim `print()` deyimi ile aynı işleve sahip olmakla birlikte ek özellikler içermektedir.

Kullanım şekli: `System.out.printf (çıkış formatı, ifadeler);`

`printf()` metodu, çıkış formatı olarak `print()` metodunda bahsedilen çıkış karakterlerine ek olarak aşağıdaki karakterleri de kullanabilir

# Giriş Çıkış (I/O) Komutlar

Basılacak Karakter veya İşlevi	Çıkış Karakteri	Örnek Kod	Ekran Çıktısı
Ondalıklı (Decimal) tamsayı	%d	System.out.printf("%d\n", +26); System.out.printf("%d\n", -26);	26 -26
Sekizli (Octal) tamsayı	%o	System.out.printf("%o", +26);	32
Onaltılı (Hexadecimal) tamsayı	%x veya %X	System.out.printf("%x\n", 26); System.out.printf("%X", 26);	1a 1A
Tek Karakter	%c	System.out.printf("%c", 'A');	A
String İfade	%s veya %S	System.out.printf("%s\n", "Ali"); System.out.printf("%S\n", "Veli"); System.out.printf("%3.2s\n", "bade"); (2 karakter boşluk bıraktıktan sonra 3 karakterden itibaren 2 karakter gösterir.)	Ali VELİ ØØba
Gerçek (float) sayı, Standart gösterim	%f	System.out.printf("%f", +26.0);	26,000000
Gerçek (float) sayı, Bilimsel gösterim	%e veya %E	System.out.printf("%e", +26.0); System.out.printf("%E", +26.0);	2.600000e+01 2.600000E+01

# Giriş Çıkış (I/O) Komutlar

- Date nesnesi farklı türevlerini elde etmek için printf() kullanımı.
- Örnek:

```
import java.util.*;
class TarihSaat {
    public static void main (String [] args){
        Date bugun =new Date();
        System.out.printf("Date nesnesinin ürettiği Tarih-Saat..|%s|\n", bugun);
        System.out.printf("Formatlı Tarih-Saat..|%tc|\n", bugun);
        System.out.printf("(YYYY-AA-GG) Formatında Tarih |%tF|\n", bugun);
        System.out.printf("(AA-GG-YY) Formatında Tarih..|%tD|\n", bugun);
        System.out.printf("12 Saat Formatında..|%tr|\n", bugun);
        System.out.printf("24 Saat Formatında..|%tR|\n", bugun);
        System.out.printf("24 Saat Formatında Saniyeli..|%tT|\n", bugun);
    }
}
```



# Giriş Çıkış (I/O) Komutlar

## Program Çıktısı

```
Date nesnesinin ürettiği Tarih-Saat..|Thu Oct 16 09:04:29 EEST 2008|  
Formatlı Tarih-Saat..|Per Eki 16 09:04:29 EEST 2008|  
(YYYY-AA-GG) Formatında Tarih |2008-10-16|  
(AA-GG-YY) Formatında Tarih..|10/16/08|  
12 Saat Formatında..|09:04:29 AM|  
24 Saat Formatında..|09:04|  
24 Saat Formatında Saniyeli..|09:04:29|
```

# Giriş Çıkış (I/O) Komutlar

**Örnek:** Klavyeden girilen A ve B gibi iki sayının ortalamasını alan programı kodlayınız.

## **PROGRAM:**

```
import java.util.Scanner;
class Ortalama {
    public static void main(String[] args) {
        Scanner tara=new Scanner (System.in);
        int a,b;
        float ortalama;
        System.out.println("A sayısını giriniz:");
        a=tara.nextInt();
        System.out.println("B sayısını giriniz:");
        b=tara.nextInt();
        ortalama=(a+b)/2;
        System.out.println("Ortalama= " + ortalama);
    }
}
```

# Giriş Çıkış (I/O) Komutlar

**Örnek:** Klavyeden girilen 5 harfli bir kelimeyi ters çevirip ekranda gösteren (“melek” girildiğinde ekranda “kelem” gösteren) programı kodlayınız.

## PROGRAM:

```
import java.util.Scanner;
public class Terscevir {
    public static void main(String[] args) {
        Scanner tara=new Scanner(System.in);
        char c1, c2, c3,c4,c5;
        c1=tara.findInLine(".").charAt(0);
        c2=tara.findInLine(".").charAt(0);
        c3=tara.findInLine(".").charAt(0);
        c4=tara.findInLine(".").charAt(0);
        c5=tara.findInLine(".").charAt(0);
        System.out.print(c5);
        System.out.print(c4);
        System.out.print(c3);
        System.out.print(c2);
        System.out.print(c1);
    }
}
```

# Koşul İfadeleri

- Koşul ifadeleri, programlama dilinde bir durum hakkında karar verme ve bir kod parçacığının çalışmasının bir şarta bağlandığı durumlarda kullanılır.
- Örnek olarak bir sayının negatif bir sayı olduğunu anlamak için, sayının 0'dan küçük olması gerekmektedir. Bu durumda şart olarak sayının 0 ile kıyaslanma işleminin yapılması gereklidir.
- Koşul ifadeleri olarak, if, else if ve switch case kullanılır.

# If ifadesi 1/3

## if DEYİMİ

Koşula bağlı olarak tek bir işlemi yerine getiren yani şartlı dallanma işlemini gerçekleştiren yapıdır. Koşul doğru (true) ise işlem yapılır. Koşul yanlış (false) ise herhangi bir işlem yapılmaz, program işleyişi bir alt satırdan devam eder.

### Kullanım şekli:

```
if (koşul)
{
    <işlemler>;
}
```

**NOT:** if yerine IF ya da If yazılırsa hata verir.

**NOT:** if'den sonra yalnızca bir deyim görünebilir. Daha fazla deyim yerleştirmek istediğiniz zaman, küme parantezlerini kullanmanız gerekir.

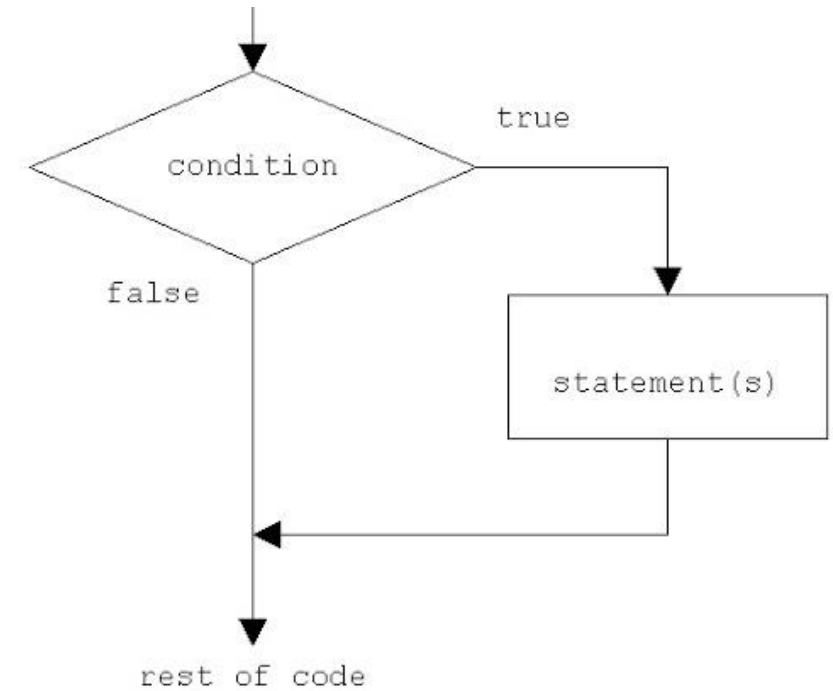
## If İfadesi 2/3

- If komutu içindeki şart, boolean ifadeden oluşmaktadır.
- `if(şart) { } / if(boolean) { }`
- Boolean ifade değeri true olur ise, if komutu içindeki kodlar çalışmaktadır.

```
if(şart)  
{  
    //Şart sağlandığında çalışacak kodlar..  
}
```

# If ifadesi 3/3

```
int sayi=-3;  
  
if(sayi < 0)  
{  
    System.out.print("sayi negatiftir.");  
}
```



```
if(true)  
{  
    System.out.print("sayi negatiftir.");  
}
```

# If Else İfadesi

- If else ifadesi, if ifadesinde ek olarak boolean olan şart değeri false olduğunda da çalışacak kod bloğunu barındırır.

```
if(sart) //true ise,
{
    //Şart sağlandığında çalışacak kodlar..
}
else //false ise,
{
}
```

```
int sayi=4;

if(sayi < 0)
{
    System.out.print("sayi negatiftir.");
}
else
{
    System.out.print("sayi pozitiftir.");
}
```



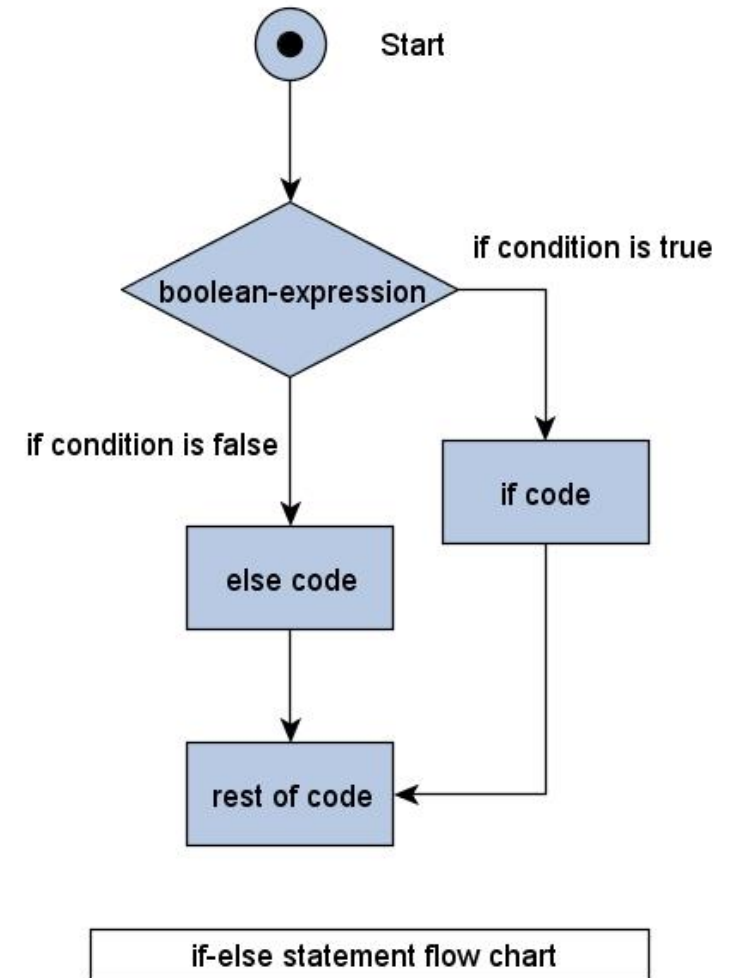
# If, Else if, else İfadesi 1/2

- If ifadesinde bulunan şart işlemini, ek olarak farklı şartlar ile kullanılmak istendiğinde else if şart blokları if deyiminden sonra eklenir. Durumların tamamının sağlanamaması ile oluşan false boolean değeri için en sona else ifadesi eklenmelidir.

```
if($art1) //true ise,  
{  
    //Şart sağlandığında çalışacak kodlar..  
}  
else if($art2) //true ise,  
{  
  
}  
  
....  
else if($artn) //true ise  
else //false ise,  
{  
  
}
```

## If, Else if, else İfadesi 2/2

```
if(sayi < 0)
{
    System.out.print("sayi negatiftir.");
}
else if(sayi > 0)
{
    System.out.print("sayi pozitifdir.");
}
else
{
    System.out.print("sayi sıfır'dır.");
}
```

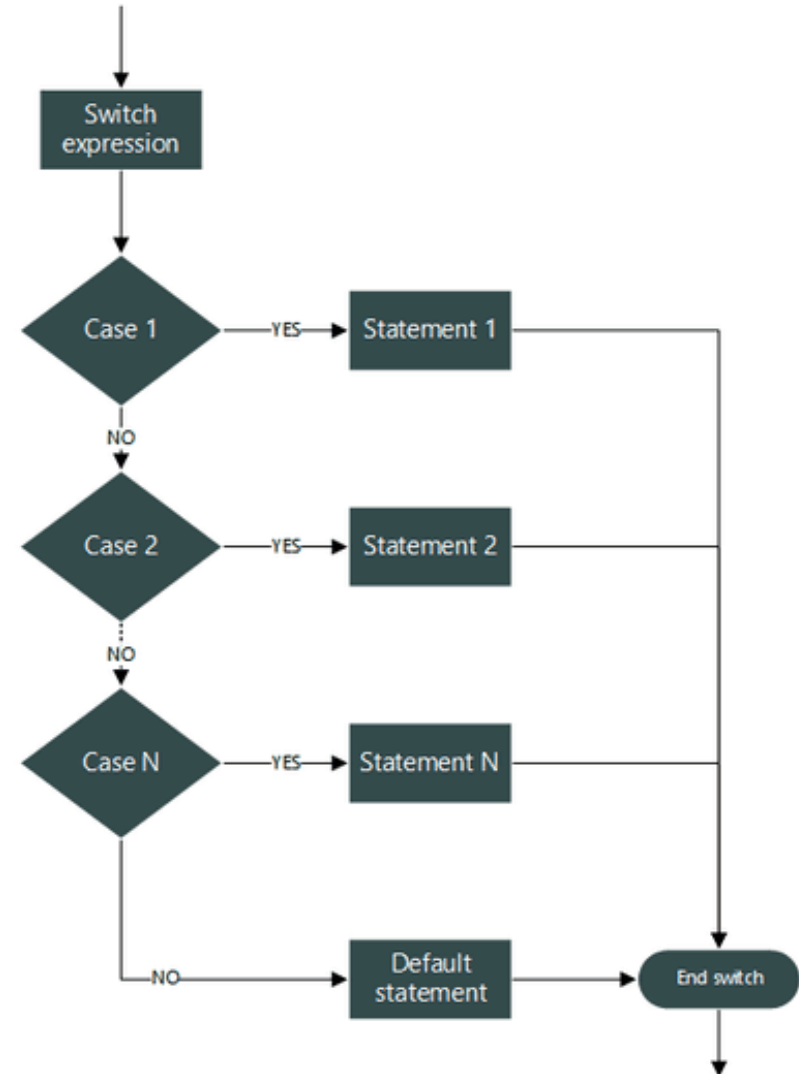


# Switch Case 1/5

- Switch-Case, koşul ifadeleri için kullanılan bir başka yapıdır. Tek bir şarta bağlı olarak, bu şartı karşılayan çok çeşitli değerlere göre kod blokları çalıştırılabilir.
- Karmaşık yapıdaki if-else blokları yerine, daha okunabilir oldukları için kullanılırlar.
- Switch-Case yapısının çalışması, bir deyimin yada şartın değeri, sabitlerden oluşan bir listede test edilir. Deyimin değeri sabitlerden birisi ile eşleşince, bu eşleşme için tanımlanan işlemler yapılır.

# Switch Case 2/5

```
switch(ifade)
{
    case sabit1:
        Yapılacak işlemler;
        break;
    case sabit2:
        Yapılacak işlemler;
        break;
    case sabit3:
        Yapılacak işlemler;
        break;
    .
    .
    .
    default:
        Yapılacak işlemler;
        break;
}
```

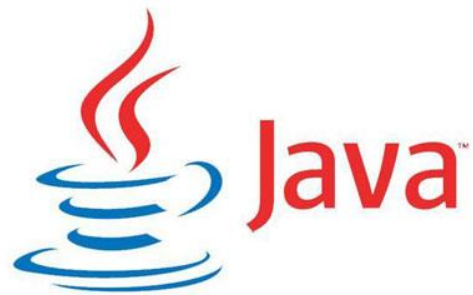


# Switch Case 3/5

## Kurallar

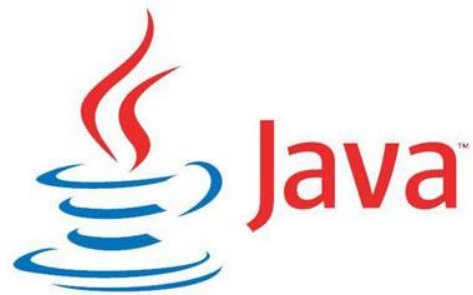
- Case anahtar sözcüğünün yanındaki ifadeler sabit olmak zorundadırlar. Bu ifadeler içerisinde değişken bulunamaz.
- Case ifadeleri herhangi bir tam sayı sabiti, karakter veya string sabiti olabilir.
- Bir switch bloğunda default case olmak zorunda değildir.
- Bir switch bloğunda iki veya daha fazla sayıda aynı değere sahip case ifadesi bulunamaz.
- Default durumunu istediğimiz yere yazabiliriz. Aynı şekilde case ifadelerini de istediğimiz sırada yazabiliriz.

# Switch Case 4/5



```
int deger=5;
switch (deger)
{
    case 1:
        System.out.println("Deger 1'dir.");
        System.out.println(1+5);
        break;
    case 2:
        System.out.println("Deger 2'dir.");
        break;
    case 3:
        System.out.println("Deger 3'tür.");
        break;
    default :
        System.out.println("Deger aralık dışındadır.");
        System.out.println("*****");
        break;
}
```

# Switch Case 5/5



```
String num="two";  
switch (num)  
{  
    case "one":  
        System.out.println("Deger 1'dir.");  
        System.out.println(1+5);  
        break;  
    case "two":  
        System.out.println("Deger 2'dir.");  
        break;  
    case "three":  
        System.out.println("Deger 3'tür.");  
        break;  
    default :  
        System.out.println("Deger aralık dışındadır.");  
        System.out.println("*****");  
        break;  
}
```

# Koşul Operatörü ( ? : ) (Ternary) 1/2

- `booleanExpression ? expression1 : expression2`
- `ifade1 ? ifade2 : ifade3`
- `ifade1` doğru ise `ifade2`, değil ise `ifade3` çalışır.
- Örnek;

```
int x=10;
```

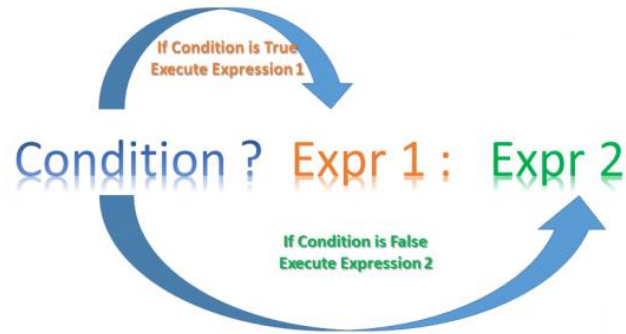
```
int y=4;
```

```
int buyukSayi = (x<=y) ? y : x;
```

```
System.out.println(max);
```



# Koşul Operatörü ( ? : ) (Ternary) 2/2



```
*/  
public static void main(String[] args) {  
    // TODO code application logic here  
    boolean durum=false;  
    String sonuc=(durum)? "Başarılı" : "Başarısız";  
    System.out.println(sonuc);  
}
```

```
public static void main(String[] args) {  
    // TODO code application logic here  
    int durum=1;  
    String sonuc=(durum == 1)? "Başarılı" : "Başarısız";  
    System.out.println(sonuc);  
}
```

Görsel Linki

<https://www.mycertnotes.com/en/nice-view-with-java-ternary-operator/>

# Döngüler/For

## for DÖNGÜSÜ

### Kullanım şekli:

for (başlangıç değeri; koşul; artım ya da azalım miktarı)  
    tekrarlanacak işlem

ya da

```
for (başlangıç değeri; koşul; artım ya da azalım miktarı)
{
    tekrarlanacak işlemler
}
```

**Örnek:** 0'dan 20'ye kadar olan çift sayıları ekranda yan yana gösteren programı for döngüsü ile yazınız.

### Program

```
public class Ciftsayilar {
    public static void main(String[] args) {
        for (int i=0; i<=20; i+=2)
            System.out.print(i+ " ");
    }
}
```

# Döngüler/For

Örnek: 10 defa ekrana MUTBMYO yazan programı for döngüsü ile yapınız.

## Program

```
public class Mesaj {  
    public static void main(String[] args) {  
        int sayac;  
        for (sayac=1;sayac<=10;sayac++)  
            System.out.println("MUTBMYO");  
    }  
}
```

# Döngüler/For

**Örnek:** Aşağıdaki programın çıktısı ne olur?

```
public class Cikti {  
    public static void main(String[] args) {  
        int sayac;  
        for (sayac=1; sayac<=10; sayac++)  
        {System.out.print(sayac + " ");  
        if (sayac==6)break;  
        }  
    }  
}
```



**ÇIKTI**

1 2 3 4 5 6

# Döngüler/For

Örnek: Dışardan girilen sayının faktöriyelini alan programı for döngüsü ile yapınız.

## Ekran Çıktısı

Faktöriyeli alınacak sayıyı giriniz

5

5!=120

## Program

```
import java.util.Scanner;
public class Faktoriyel {
    public static void main(String[] args) {
        Scanner tara=new Scanner(System.in);
        int n,fakt,i;
        System.out.println ("Faktöriyeli alınacak sayıyı giriniz");
        n=tara.nextInt();
        fakt=1;

        for (i=1; i<=n;i++)
            fakt=fakt*i;

        System.out.println(n+ "!=" + fakt);
    }
}
```

# Döngüler/For

<b>1'den 100'e kadar sayılar, artım miktarı 1</b>	<b>for(int i=1; i&lt;=100; i++)</b>
<b>100'den 1'e kadar sayılar, azalım miktarı 1</b>	<b>for(int i=100; i&gt;=1; i--)</b>
<b>7'den 77'e kadar olan sayılar, artım miktarı 7</b>	<b>for(int i=7; i&lt;=77; i+=7)</b>
<b>20'den 2'ye kadar olan sayılar, azalım miktarı 2</b>	<b>for(int i=20; i&gt;=2; i-=2)</b>
<b>2,5,8,11,14,17,20 serisi elde etmek için</b>	<b>for(int i=2; i&lt;=20; i+=3)</b>
<b>71,61,51,41,31,21,11,1 serisi elde etmek için</b>	<b>for(int i=71; i&gt;=1; i-=10)</b>

# Döngüler/While

## While döngüsü

Verilen koşul doğru olduğu sürece belirtilen işlemler tekrar edilir. Koşul kontrol işlemi, döngünün başında yapılır.

### Kullanım şekli:

```
while (koşul)
{
    işlemler;
}
```

# Döngüler/Do-While

## do/while döngüsü

Bir defa işlem yapıldıktan sonra koşul doğru olduğu sürece belirtilen işlemler tekrar edilir. Koşul kontrol işlemi döngünün sonunda yapılır.

### Kullanım şekli:

```
do  
{  
    işlemler;  
}while(koşul);
```



# Döngüler

## Örnek

```
int a=5;  
do  
{  
a=a+1;  
}while(a<3);  
System.out.println(a);
```

İlk önce işlem gerçekleşir sonra koşula bakılır. Koşul gerçekleşmediği için döngüden çıkılır. Buna göre ekran çıktısı; **6**'dır.

```
int a=5;  
while (a<3)  
{  
a=a+1;  
}  
System.out.println(a);
```

İlk önce koşula bakılır, koşul gerçekleşmediği için döngüden çıkılır. Buna göre programın ekran çıktısı; **5**'dir.

# Döngüler/Break Kullanımı

**Örnek:** Aşağıdaki programın ekran çıktısı ne olur?

```
public class Cikti {  
    public static void main(String[] args) {  
        int a=0;  
        double top=0;  
        dongu1:  
        while (a<10)  
        {  
            a++;  
            top=top+a;  
            if(a==2)  
                break dongu1;  
            System.out.println(top);  
        }  
    }  
}
```

**CEVAP:** Program normalde 1'den 10'a kadar olan sayıların toplamını yapıyor ancak a değişkeni ilk (a=1) aldıktan sonra, toplama işlemi (top=top+a) işlemi gerçekleşiyor ve ekrana bu değer yazılıyor. A değişkeni 2 değerini aldıktan sonra toplama işlemi gerçekleşiyor ama

**if (a==2) break git;**

ifadesi döngüden çıkmamıza neden olur ve ekrana top değişkeninin değeri 1.0 olarak yazılır. Programdan

**// if (a==2) break git;**

ifadesini çıkarırsak ekrana 1.0 dan 55.0 kadar olan toplama değerini yazar.

**NOT:** while, do-while, for döngülerinden çıkmak için break deyimi kullanılır.

# Döngüler/Break

## break Deyimi:

break deyimi **etiketli (labeled)** ve **etiketsiz (unlabeled)** olmak üzere iki farklı şekilde kullanılabilir. Etiketsiz break deyimi ile en içteki while, do-while, for döngüleri sonlandırılırken, etiketli break deyimi etiket hangi döngünün başına konulmuş ise o döngü sistemini sona erdirir

# Döngüler/Continue

## continue Deyimi:

continue deyimi, döngü içerisinde o anki işlemin atlanmasını (göz ardı edilmesini) ve bir sonraki işlemde devam edilmesini sağlayan bir komuttur. continue deyimini de break deyimi gibi while, do-while ve for döngülerinde etiketli veya etiketsiz olarak kullanılabilir. Etiketsiz continue deyimi en içteki döngü içerisinde etkili olurken, etiketli continue deyimi ise başına konulduğu döngü yapısını etkiler.

# Örnek Uygulama

- Dışarıdan girilen nota göre öğrencinin o dersten geçip, geçmediğini ekranda gösteren programı yazınız.

## Program (1.YOL)

```
import java.util.Scanner;
public class Not {
    public static void main(String[] args) {
        Scanner tara=new Scanner (System.in);
        int not;
        System.out.println("Notunuzu giriniz");
        not=tara.nextInt();
        if (not>=50)
            System.out.println("Geçtiniz");
        else
            System.out.println("Kaldınız");
    }
}
```

## Program (2.YOL)

```
import java.util.Scanner;
public class Not
{
    public static void main(String[] args)
    {
        Scanner tara=new Scanner (System.in);
        int not;
        System.out.println("Notunuzu giriniz");
        not=tara.nextInt();
        System.out.println(not >=50 ? "Geçtiniz" : "Kaldınız");
    }
}
```

**Not:** Aynı program if/else yapısı kullanılmadan ? operatörü ile yapıldı.

# Örnek Uygulama Switch-Case

**Örnek:** 100'lük sisteme göre girilen başarı notunu harfli sistemde gösteren programı switch/case ve if/else if deyimleri kullanarak ayrı ayrı yazınız.

## **Kriter**

<b>90 ve üzeri</b>	<b>AA</b>
<b>80-89</b>	<b>BA</b>
<b>70-79</b>	<b>BB</b>
<b>60-69</b>	<b>CB</b>
<b>50-59</b>	<b>CC</b>
<b>50'den aşağı</b>	<b>FF</b>

# 100'lük Sistem

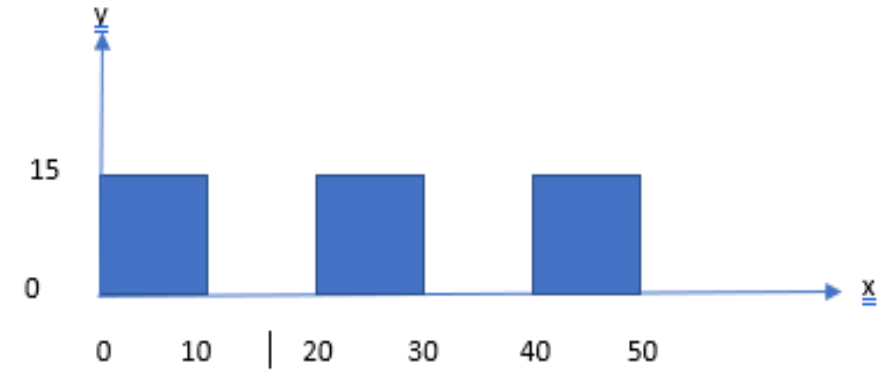
## Ders Notu Çözüm

```
import java.util.Scanner;
public class Basarinotu {
    public static void main(String[] args) {
        Scanner tara=new Scanner(System.in);
        int not;
        System.out.println ("Başarı notunuzu giriniz");
        not=tara.nextInt();
        if (not>=90 && not<=100)
            System.out.println("AA");
        else if (not>=80 && not<=89)
            System.out.println("BA");
        else if (not>=70 && not<=79)
            System.out.println("BB");
        else if (not>=60 && not<=69)
            System.out.println("CB");
        else if (not>=50 && not<=59)
            System.out.println("CC");
        else
            System.out.println("FF");
    }
}
```

```
import java.util.Scanner;
public class Basarinotu {
    public static void main(String[] args) {
        Scanner tara=new Scanner(System.in);
        int not;
        System.out.println ("Başarı notunuzu giriniz");
        not=tara.nextInt();
        switch (not/10) {
            case 10:
            case 9:
                System.out.print("AA");
                break;
            case 8:
                System.out.print("BA");
                break;
            case 7:
                System.out.print("BB");
                break;
            case 6:
                System.out.print("CB");
                break;
            case 5:
                System.out.print("CC");
                break;
            default:
                System.out.print("FF");
                break;
        }
    }
}
```

# Uygulamalar

- 3 Sayıyı karşılaştıran ve en büyük olanı bulan program kodlarını yazınız.(if Else yapısı kullanılarak yapılacak !)
- Grafikte bulunan x Değerine göre, y değerini hesaplayan kodları yazınız.(X değeri kullanıcıdan alınacak !)





# 3 Sayının En Büyüğü Çözüm !

- Çözüm 1 (Maybe Spagetti Code !)

```
int sayi1,sayi2,sayi3,enBuyuk;
sayi1=5;
sayi2=7;
sayi3=2;

if (sayi1 > sayi2)
    if(sayi1>sayi3)
        enBuyuk=sayi1;
    else
        enBuyuk=sayi3;
else
    if(sayi2>sayi3)
        enBuyuk=sayi2;
    else
        enBuyuk=sayi3;

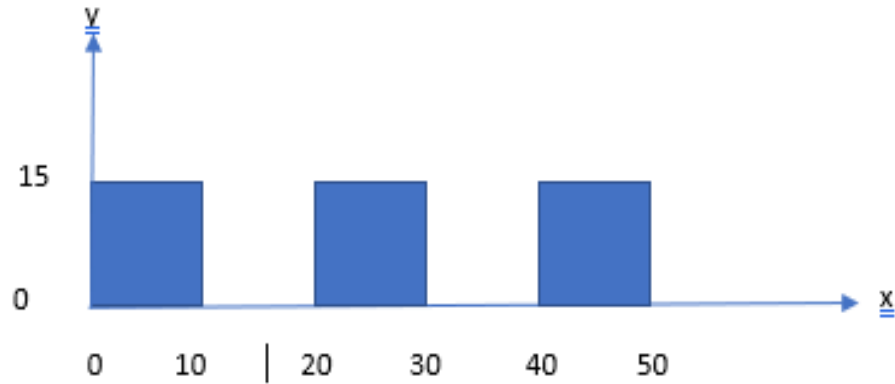
System.out.println(enBuyuk);
}
```

- Çözüm 2(Maybe Clean Code)

```
int sayi1,sayi2,sayi3,enBuyuk;
sayi1=5;
sayi2=7;
sayi3=2;
if(sayi1>sayi2 && sayi1>sayi3)
    enBuyuk = sayi1;
else if(sayi2>sayi1 && sayi2>sayi3)
    enBuyuk = sayi2;
else
    enBuyuk=sayi3;

System.out.println(enBuyuk);
```

# Grafik Sorusu Çözüm !



```
Scanner scan = new Scanner(System.in);
int x,y;
x=scan.nextInt();
if(x<=0)
{
    y=0;
}
else
{
    if((x>0 && x<=10) || (x>20 && x<=30) || (x>40 && x<=50))
        y=15;
    else
        y=0;
    System.out.println("y:"+y);
}
```

\*\*\*???

**Örnek:** Girilen yaş değerine kişinin durumunu aşağıdaki kriterlere göre bulan programı yazınız.

Yaş Değeri	Durum
0-5	BEBEK
6-14	ÇOCUK
15-35	GENÇ
36-45	ORTA YAŞLI
46 ve üstü	İHTİYAR

# Örnek İç İçe Döngü

Soru: Çarpım Tablosunu oluşturup ekranda gösteren programı yazınız.

1*1=1	1*2=2	1*3=3	1*4=4	1*5=5	1*6=6	1*7=7	1*8=8	1*9=9	1*10=10
2*1=2	2*2=4	2*3=6	2*4=8	2*5=10	2*6=12	2*7=14	2*8=16	2*9=18	2*10=20
3*1=3	3*2=6	3*3=9	3*4=12	3*5=15	3*6=18	3*7=21	3*8=24	3*9=27	3*10=30
4*1=4	4*2=8	4*3=12	4*4=16	4*5=20	4*6=24	4*7=28	4*8=32	4*9=36	4*10=40
5*1=5	5*2=10	5*3=15	5*4=20	5*5=25	5*6=30	5*7=35	5*8=40	5*9=45	5*10=50
6*1=6	6*2=12	6*3=18	6*4=24	6*5=30	6*6=36	6*7=42	6*8=48	6*9=54	6*10=60
7*1=7	7*2=14	7*3=21	7*4=28	7*5=35	7*6=42	7*7=49	7*8=56	7*9=63	7*10=70
8*1=8	8*2=16	8*3=24	8*4=32	8*5=40	8*6=48	8*7=56	8*8=64	8*9=72	8*10=80
9*1=9	9*2=18	9*3=27	9*4=36	9*5=45	9*6=54	9*7=63	9*8=72	9*9=81	9*10=90
10*1=10	10*2=20	10*3=30	10*4=40	10*5=50	10*6=60	10*7=70	10*8=80	10*9=90	10*10=100

# Örnek İç İçe Döngü

## Program

```
public class Carpim {  
    public static void main(String[] args) {  
        for (int i=1; i<=10; i++){  
            for (int j=1; j<=10; j++){  
                {  
                    System.out.print(i+ "*" + j + "=" + i*j + "\t");  
                }  
                System.out.println();  
            }  
        }  
    }  
}
```

```
1*1=1  1*2=2  1*3=3  1*4=4  1*5=5  1*6=6  1*7=7  1*8=8  1*9=9  1*10=10  
2*1=2  2*2=4  2*3=6  2*4=8  2*5=10 2*6=12 2*7=14 2*8=16 2*9=18 2*10=20  
3*1=3  3*2=6  3*3=9  3*4=12 3*5=15 3*6=18 3*7=21 3*8=24 3*9=27 3*10=30  
4*1=4  4*2=8  4*3=12 4*4=16 4*5=20 4*6=24 4*7=28 4*8=32 4*9=36 4*10=40  
5*1=5  5*2=10 5*3=15 5*4=20 5*5=25 5*6=30 5*7=35 5*8=40 5*9=45 5*10=50  
6*1=6  6*2=12 6*3=18 6*4=24 6*5=30 6*6=36 6*7=42 6*8=48 6*9=54 6*10=60  
7*1=7  7*2=14 7*3=21 7*4=28 7*5=35 7*6=42 7*7=49 7*8=56 7*9=63 7*10=70  
8*1=8  8*2=16 8*3=24 8*4=32 8*5=40 8*6=48 8*7=56 8*8=64 8*9=72 8*10=80  
9*1=9  9*2=18 9*3=27 9*4=36 9*5=45 9*6=54 9*7=63 9*8=72 9*9=81 9*10=90  
10*1=10 10*2=20 10*3=30 10*4=40 10*5=50 10*6=60 10*7=70 10*8=80 10*9=90 10*10=100  
BUILD SUCCESSFUL (total time: 0 seconds)
```

## Örnek: Fahrenheit'tan Santigrat dönüşüm(Fahrenheit - Celsius)

- 200 f'den 220 f'ye kadar olan sıcaklık değerini C'ye çeviren program...

100 Santigrat =  
212 Fahrenheit

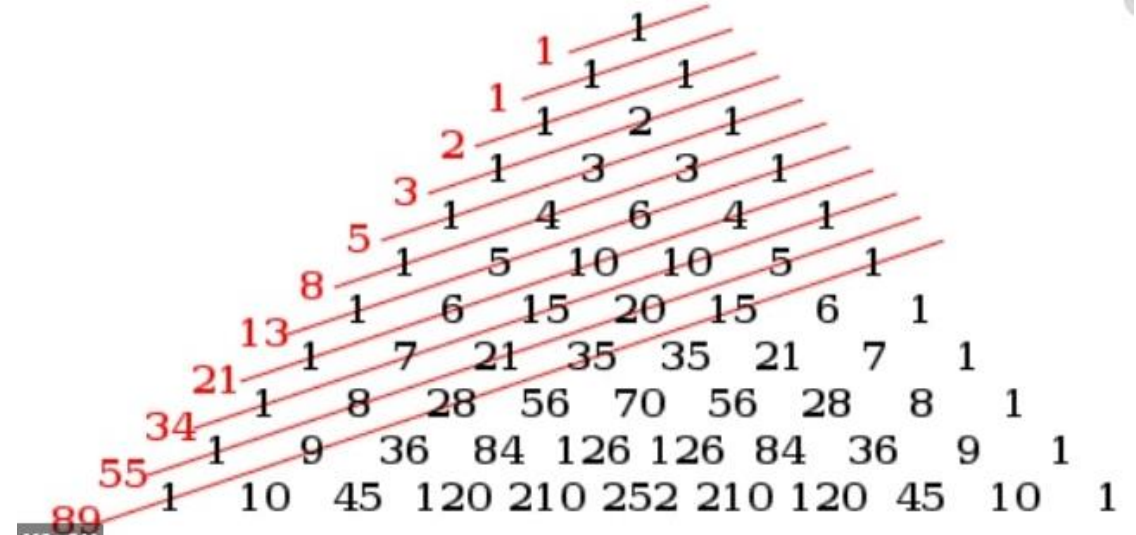


$$C = \frac{5}{9} (F - 32)$$

```
public class Donguler {  
  
    public static void main(String[] args) {  
        int f=200;  
        double c;  
        System.out.println("\tF\t\tC");  
        System.out.println("\t__ \t\t__")  
        while (f<=220) {  
            c=5.0/9.0*(f-32);  
            System.out.println("\t"+f+"\t\t"+c);  
            f++;  
        }  
    }  
}
```

# Örnek: Döngüler/ Fibonacci(1,2,3,5,8,13,21,34...)

```
public class Donguler {  
  
    public static void main(String[] args) {  
        int sayac=0;  
        int a=0,b=1;  
  
        while (sayac<10) {  
            a=a+b;  
            b=a-b;  
            System.out.println(a);  
            sayac++;  
        }  
    }  
}
```



# Kaynaklar

- Java ve Java Teknolojileri, *Tevfik KIZILÖREN* – Kodlab Yayınları
- Yrd. Doç. Dr. Deniz KILINÇ-Celal Bayar Üniversitesi / Nesneye Yönelik Programlama Ders Notları
- Yrd. Doç. Dr. Erbil Akbay –Marmara Üniversitesi / Object Oriented Programming 1-2 Ders notları
- [http://javayaz.com/?page\\_id=84](http://javayaz.com/?page_id=84)
- <http://mail.baskent.edu.tr/~tkaracay/etudio/ders/prg/java/ch06/privilegesOfOperators.pdf>
- [http://www.megep.meb.gov.tr/mte\\_program\\_modul/moduller\\_pdf/Kontrol%20Deyimleri.pdf](http://www.megep.meb.gov.tr/mte_program_modul/moduller_pdf/Kontrol%20Deyimleri.pdf)