

BBS515 Nesneye Yönelik Programlama

Ders 2

Zümra Kavafođlu
<https://zumrakavafoglu.github.io/>

print metodu

```
//Zumra Kavafoglu
//02.10.2017

//Text-printing with multiple statements
public class Welcome2 {

    //main method begins execution of Java application
    public static void main(String args[]) {

        System.out.print("Welcome ");
        System.out.println("to Java Programming");

    } //end method main
} //end class Welcome2
```

System.out.print imleci aynı satırda tutar, System.out.println imleci bir alt satıra indirir

Welcome to Java Programming!

Tamsayı toplama programı

Problem: Değerleri verilen iki tamsayının toplamını ekrana yazdırınız.

Neler öğreneceğiz:

- Değişken tanımlama ve değer atama
- Primitif veri tipleri
 - Nümerik veri tipleri
 - Boolean
 - char
- String veri tipi
- İkili nümerik operatörler
- Kısayol atama operatörleri
- Arttırma ve azaltma operatörleri
- printf
- Hafızada değişkenler

Tamsayı toplama programı

Addition.java ×

```
1 //Program for displaying the sum of two integers
```

```
2 public class Addition {
```

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

```
4 //first integer to add  
5 int number1;
```

```
6 //second integer to add  
7 int number2;
```

```
8 //sum of two integers  
9 int sum;
```

```
10 //assign number1  
11 number1 = 3;
```

```
12 //assign number2  
13 number2 = 8;
```

```
14 //Compute the sum  
15 sum = number1 + number2;
```

```
16 //Print the sum  
17 System.out.print("Sum is " + sum);
```

```
18 }
```

number1, number2 ve
sum değişkenlerini
tanımla

İyi deęişken isimleri için öneriler

- Deęişken isimleri küçük harfle başlar ve isimdeki takip eden her kelimenin baş harfi büyük yazılır
 - **örnek:** firstNumber
- Anlamli deęişken isimleri seçmek programın çok fazla yorum satırına veya dokümantasyona ihtiyaç olmadan anlaşılabilir olmasını sağlar. Deęişken ismini kısa tutmak için anlamından feragat edilmemelidir.

Tamsayı toplama programı

Addition.java ×

```
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
```

number1 değişkenine
3 değerini ata

number2 değişkenine
8 değerini ata

Değişken tanımlama ve değer atama

Değişken tanımlama

```
int number1;
```

veri tipi

değişken adı

Değer atama

```
number1 = 3;
```

değişken adı

atanmak istenen değer

atama operatörü
sağındaki değeri solundaki
değişkene atar

Değişken tanımlama ve değer atama

Değişken tanımlama

```
int number1;
```

veri tipi

değişken adı

Değer atama

```
number1 = 3;
```

değişken adı

atanmak istenen değer

İlk değer ile tanımlama

```
int number1 = 3;
```

veri tipi

değişken adı

atanmak istenen değer

Basit(Primitif) Veri Tipleri : Nümerik veri tipleri

- Basit (primitif) veri tipleri Java programlama dili tarafından önceden tanımlanmış veri tipleridir.
- Primitif veri tiplerini nümerik veri tipleri, boolean ve char primitif veri tipleri olarak sınıflandırabiliriz.

Basit(Primitif) Veri Tipleri : Nümerik veri tipleri

isim	değer aralığı	depolama boyutu
byte	$-2^7(-128)$ to $2^7-1(127)$	8-bit signed
short	$-2^{15}(-32768)$ to $2^{15}-1(32767)$	16-bit signed
int	$-2^{31}(-2147483648)$ to $2^{31}-1(2147483647)$	32-bit signed
long	-2^{63} to $2^{63}-1$	64-bit signed
float	negatif aralık: $-3.4028235E+38$ to $-1.4E-45$ pozitif aralık: $1.4E-45$ to $3.4028235E+38$	32-bit IEEE 754
double	negatif aralık: $-1.7976931348623157E+308$ to $-4.9E-324$ pozitif aralık: $4.9E-324$ to $1.7976931348623157E+308$	64-bit IEEE 754

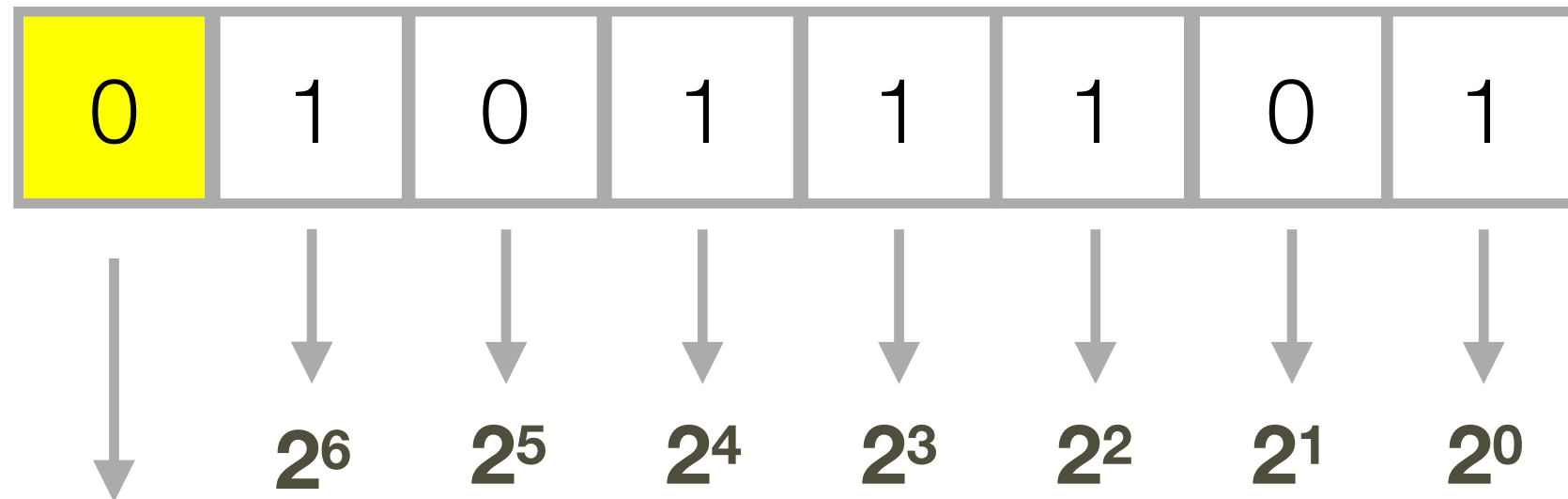
Nümerik veri tipleri : Tamsayılar

- **byte, short, int, long**: Hepsi tamsayı deęişkenleri tanımlamak için kullanılır.
- Farkları hafızada kapladıkları yer ve saklayabilecekleri deęerin büyüklüğüdür.

isim	deęer aralıęı	depolama boyutu
byte	$-2^7(-128)$ to $2^7-1(127)$	8-bit signed
short	$-2^{15}(-32768)$ to $2^{15}-1(32767)$	16-bit signed
int	$-2^{31}(-2147483648)$ to $2^{31}-1(2147483647)$	32-bit signed
long	-2^{63} to $2^{63}-1$	64-bit signed

Nümerik veri tipleri : Depolama

1 byte, 8 bit'e eşit ve her bit 1 veya 0 değerlerinden birini alıyor.



İşaret biti
pozitif : 0
negatif : 1

$$\begin{aligned} &= \mathbf{1*2^0} + 0*2^1 + \mathbf{1*2^2} + 1*2^3 + \mathbf{1*2^4} + 0*2^5 + \mathbf{1*2^6} \\ &= \mathbf{1} + 0 + \mathbf{4} + 8 + \mathbf{16} + 0 + \mathbf{64} \\ &= \mathbf{93} \end{aligned}$$

Nümerik veri tipleri : Rasyonel sayılar

- **double, float:** Rasyonel deęişkenleri tanımlamak için kullanılır. Aralarındaki en önemli fark hafızada kapladıkları yer ve hassasiyetleridir.
- float deęişkene deęer atarken deęerin sonuna f yazılmalıdır:

float x = 2.37f

isim	deęer aralığı	depolama boyutu
float	negatif aralık: $-3.4028235E+38$ to $-1.4E-45$ pozitif aralık: $1.4E-45$ to $3.4028235E+38$	32-bit IEEE 754
double	negatif aralık: $-1.7976931348623157E+308$ to $-4.9E-324$ pozitif aralık: $4.9E-324$ to $1.7976931348623157E+308$	64-bit IEEE 754

Rasyonel Sayıların bilimsel gösterimi

- Rasyonel sayılar bilimsel gösterim ile de ifade edilebilirler. Bilimsel gösterim bir rasyonel sayının üstel(exponential) kullanılarak gösterilmesidir.
- $7.45673E+2$ ifadesi $7.45673 \cdot 10^2 = 7.45673 \cdot 100 = 745.673$ sayısını ifade etmektedir.
- Benzer biçimde $7.45673E-2$ ifadesi $7.45673 \cdot 10^{-2} = 7.45673 / 100 = 0.0745673$ sayısını ifade etmektedir.

Tamsayı toplama programı

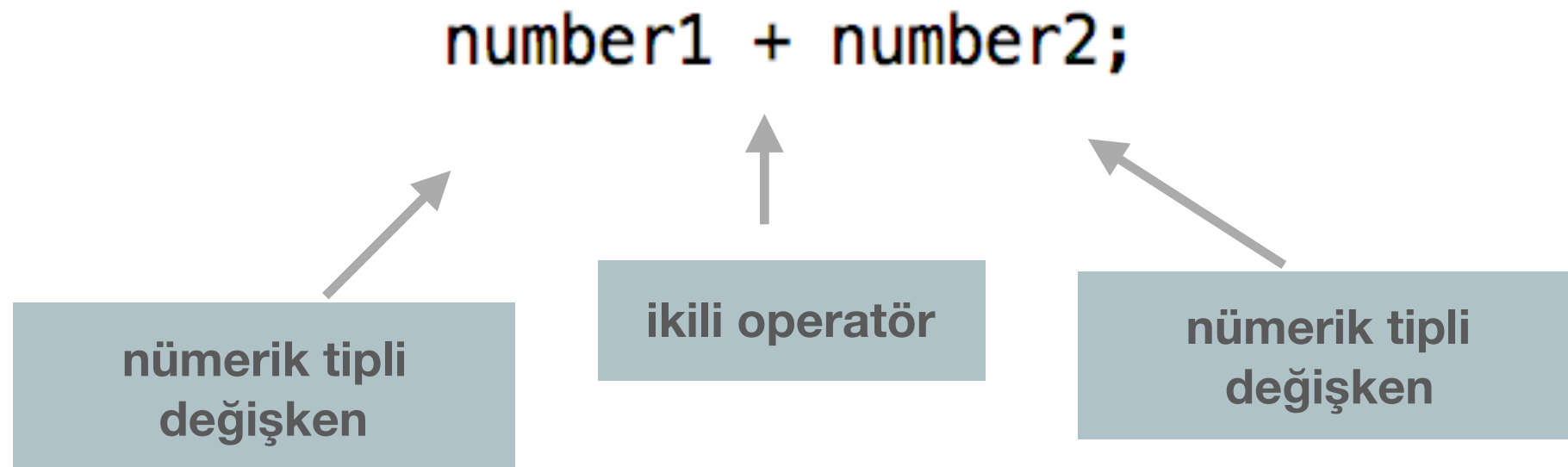
Addition.java ×

```
1 //Program for displaying the sum of two integers
2
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
```

number1 ve number2
değişkenlerinin toplamını
hesapla ve sum
değişkenine ata

İkili nümerik operatörler

İki nümerik tipli veriyi işleyen operatörlere ikili nümerik operatörler denir.



İkili nümerik operatörler

operatör	işlem
+	toplama
-	çıkarma
*	çarpma
/	bölme
%	kalan bulma

Aritmetik işlemlerin Java ile ifadesi

$$\frac{2 - 7x}{3.4} + \frac{8(y + 6)(x - 2)}{y} - \frac{\frac{x}{y} + \frac{10}{z}}{2}$$

aritmetik işleminin Java programlama diliyle yazımı aşağıdaki gibi olmalıdır:

$$(2-7*x)/3.4 + (8*(y+6)*(x-2))/y - ((x/y) + (10/z))/2.0$$

İkili nümerik operatörler : Tam sayı bölme

- int tipinde iki değişkenin bölme işleminin sonucu, bölüm tamsayı olmasa bile tamsayı çıkar.

```
int number1 = 7;  
int number2 = 2;  
int division = number1 / number2;  
System.out.print("Division is: " + division);
```

Çıktı

```
Division is: 3
```

İkili nümerik operatörler : Tam sayı bölme

Gerçek bölümü bulmak için **değişkenlerden en az biri** ve **sonuç tipi double** olarak tanımlanmalıdır.

```
double number1 = 7;  
int number2 = 2;  
double division = number1 / number2;  
System.out.println("Division is: " + division);
```

Çıktı

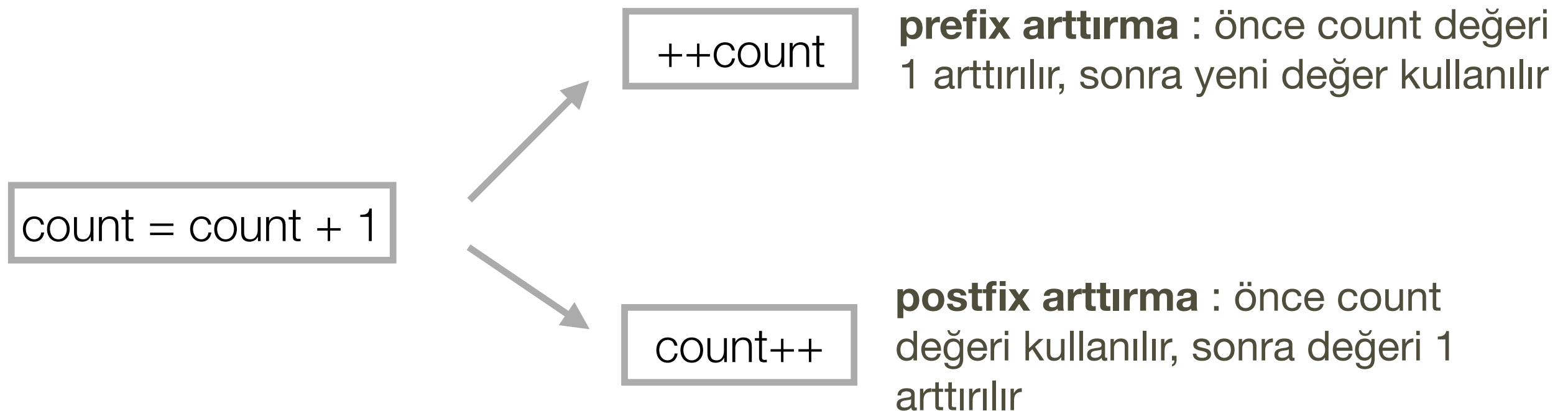
```
Division is: 3.5
```

Kısa yol atama operatörleri

operatör	örnek	işlem
+=	i+=5	i = i + 5
-=	n -= 4	n = n - 4
=	k=8	k = k * 8
/=	t/=3	t = t / 3
%=	i %= 7	i = i % 7

Arttırma ve Azaltma Operatörleri (Increment and Decrement Operators)

- Bir değişkenin 1 arttırıldığı ya da 1 azaltıldığı özel durumlar için arttırma ve azaltma işleçleri vardır.



Arttırma ve Azaltma Operatörleri (Increment and Decrement Operators)

`k = ++n`



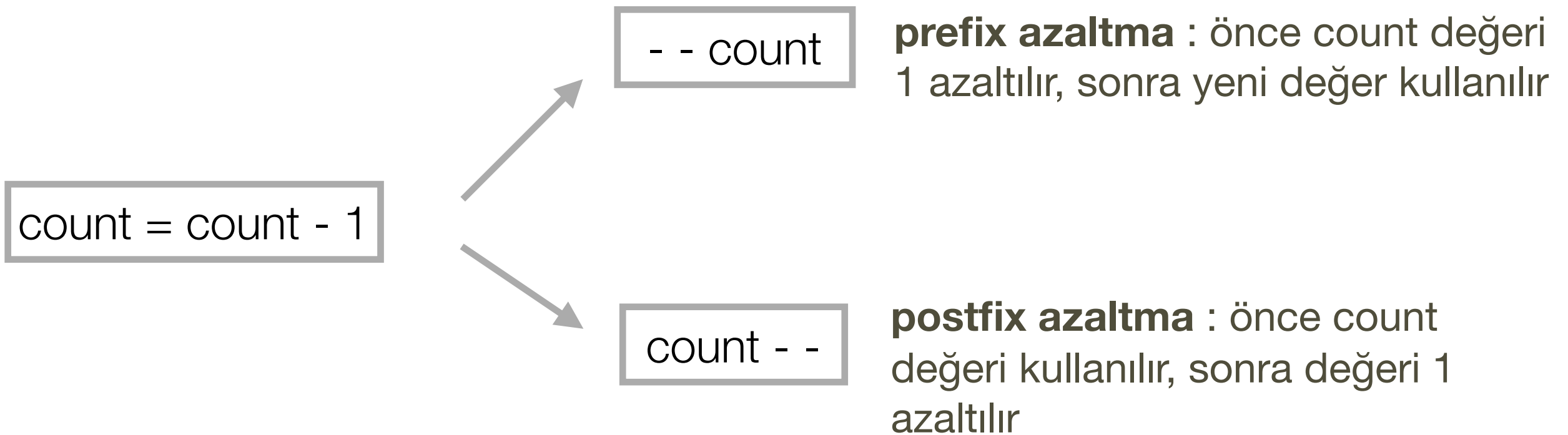
`n = n + 1`
`k = n`

`k = n++`



`k = n`
`n = n + 1`

Arttırma ve Azaltma Operatörleri (Increment and Decrement Operators)



Tamsayı toplama programı

Addition.java ×

```
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28
```

Toplamı ekrana yazdır

Tamsayı toplama programı

Addition.java ×

```
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28 }
```

Sum is 11

Toplamı printf ile yazdırma

```
1 //Program for displaying the sum of two integers
2 public class Addition {
3     public static void main(String args[]) {
4         //first integer to add
5         int number1;
6
7         //second integer to add
8         int number2;
9
10        //sum of two integers
11        int sum;
12
13        //assign number1
14        number1 = 3;
15
16        //assign number2
17        number2 = 8;
18
19        //Compute the sum
20        sum = number1 + number2;
21
22        //Print the sum
23        System.out.printf("Sum is %d", sum);
24    }
25 }
26
27
28
```

printf veriyi istenen formatta ekrana yazdırır

Sum is 11

Toplamı printf ile yazdırma

Yer tutucu argüman olarak verilen değişkenin metin içinde nerede yer alacağını ve formatını belirler.

```
System.out.printf("Sum is %d", sum);
```

yer tutucu



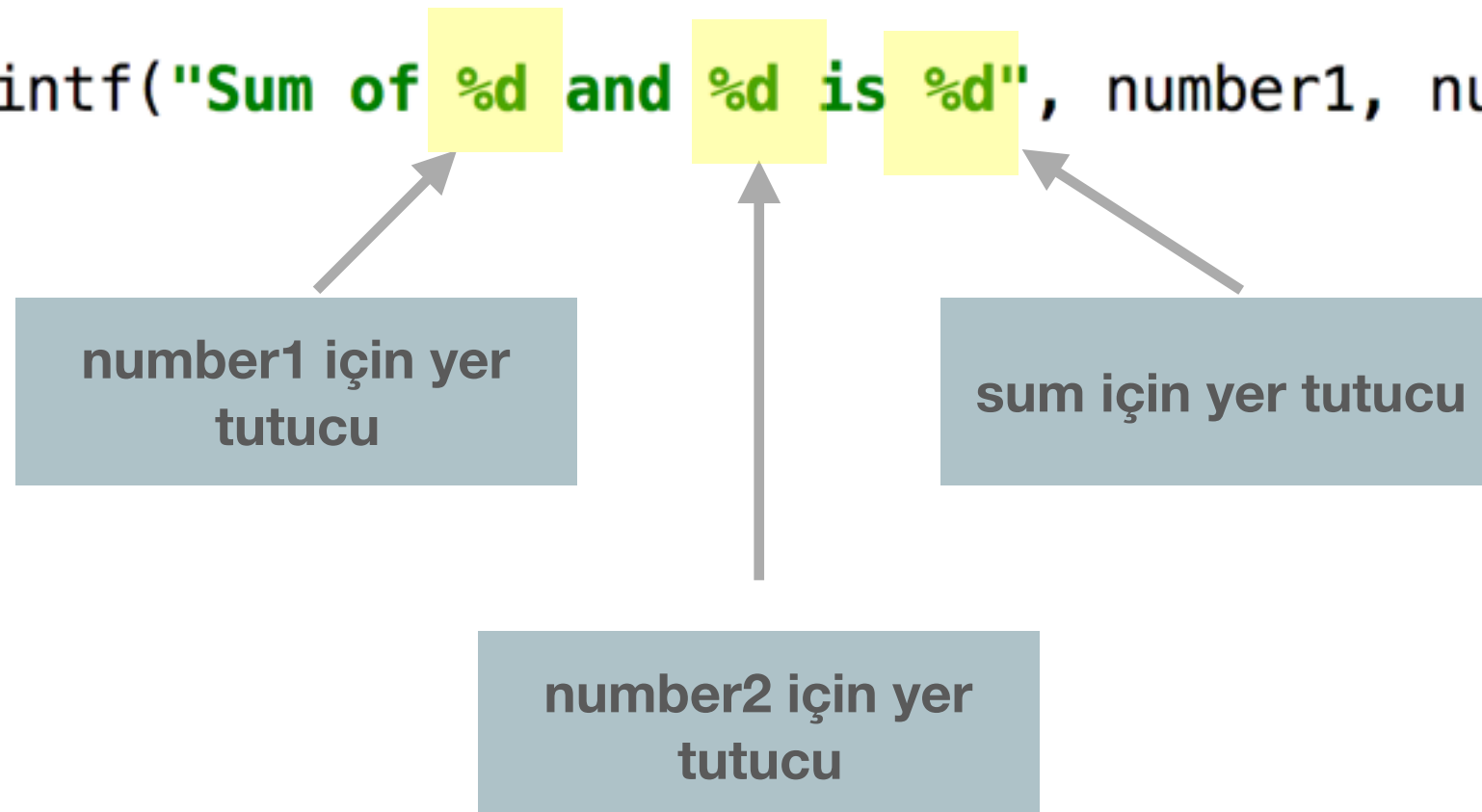
değişken

printf: dönüştürücüler

dönüştürücü	
d	tamsayı
f	rasyonel sayı
c	karakter
s	string

Toplamı printf ile yazdırma

```
System.out.printf("Sum of %d and %d is %d", number1, number2, sum);
```

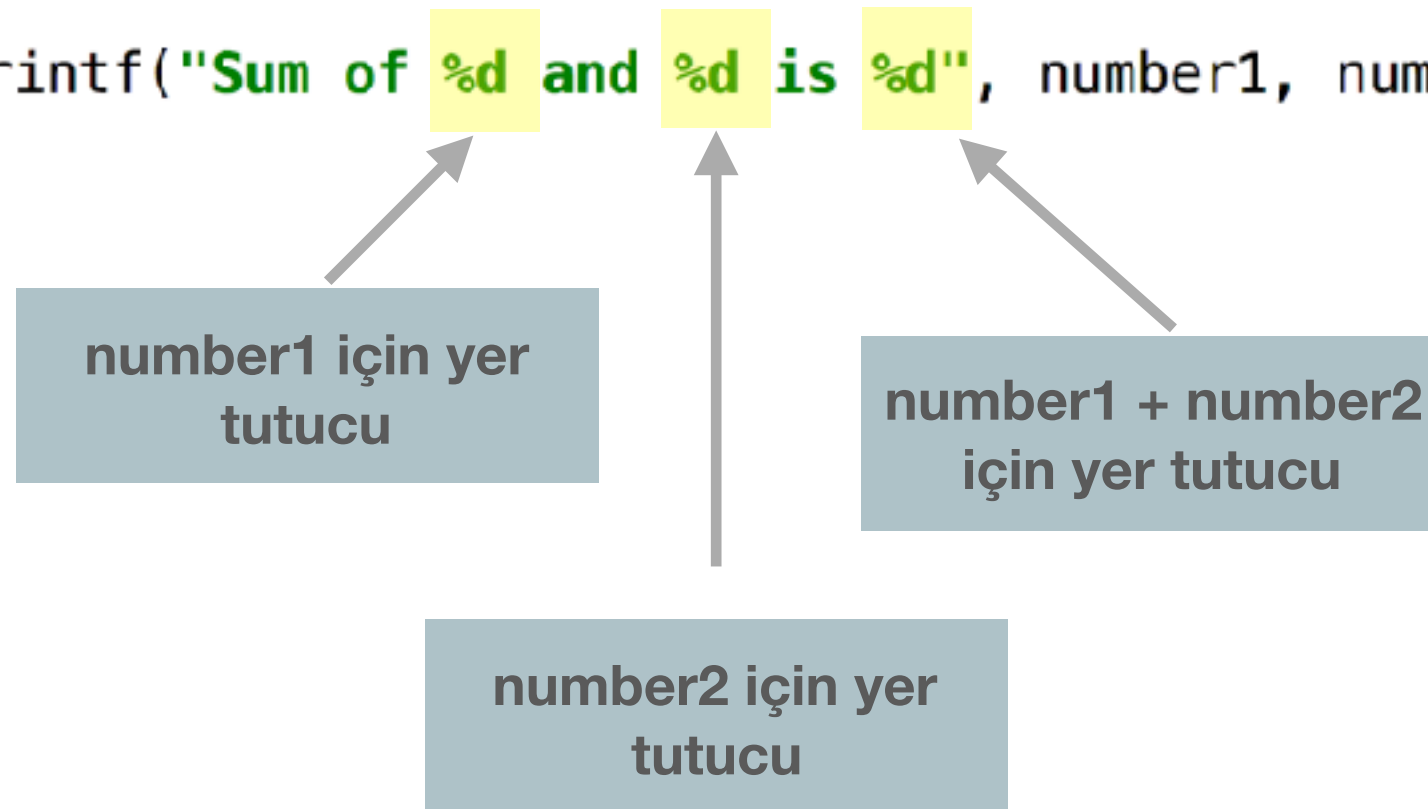


Çıktı

```
Sum of 3 and 8 is 11
```

printf: işlem printf'in içinde de yapılabilir

```
System.out.printf("Sum of %d and %d is %d", number1, number2, number1 + number2);
```



Çıktı

```
Sum of 3 and 8 is 11
```

printf: rasyonel sayıları formatlı yazdırma

```
double number = 12.367;  
System.out.printf("Number is %f", number);
```

Çıktı

```
Number is 12.367000
```

↑
default olarak virgülden
sonra 6 haneli yazdırır

printf: rasyonel sayıları formatlı yazdırma

```
double number = 12.367;  
System.out.printf("Number is %.2f", number);
```

Çıktı

```
Number is 12.37
```

↑
%.2f ile virgülden sonra 2 haneli yazdırır

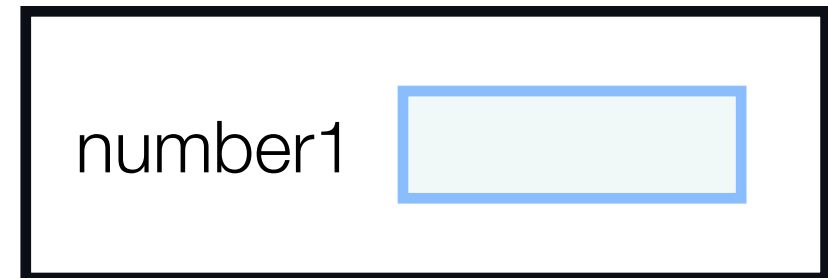
Hafızada deęişkenler

- Her deęişkenin bir adı, tipi, boyutu ve deęeri vardır.
- Deęişkenin adı hafızada bir yere karşılık gelir.
- Bir deęişkene yeni bir deęer atandığında bir önceki deęeri deęiştirir(ve yok eder).
- Deęişkenleri hafızadan okumak onların deęerini deęiştirmez.

Hafızada değişkenler

```
1 //Program for displaying the sum of two integers
2 public class Addition {
3     public static void main(String args[]) {
4         //first integer to add
5         int number1;
6         //second integer to add
7         int number2;
8
9         //sum of two integers
10        int sum;
11
12        //assign number1
13        number1 = 3;
14
15        //assign number2
16        number2 = 8;
17
18        //Compute the sum
19        sum = number1 + number2;
20
21        //Print the sum
22        System.out.print("Sum is " + sum);
23    }
24 }
25
26
27
28
```

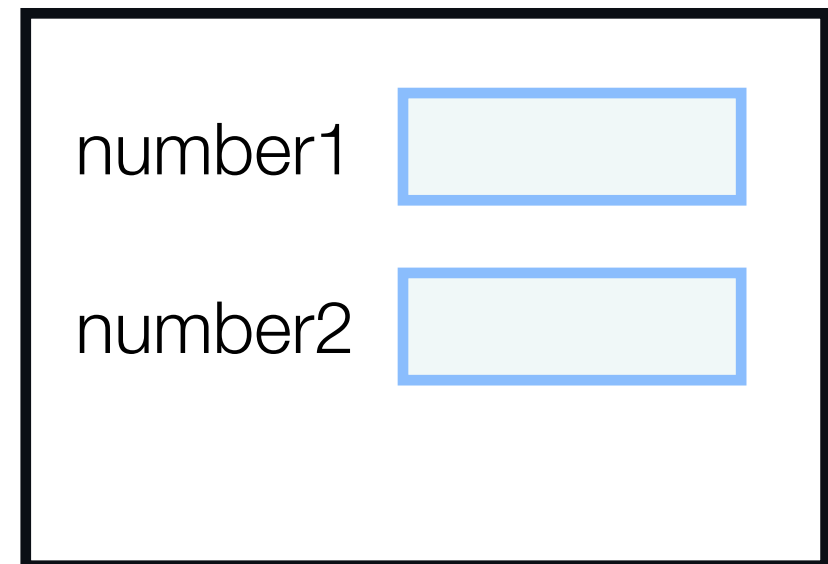
number1 için hafızadan yer ayır



Hafızada değişkenler

```
Addition.java x
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28 }
```

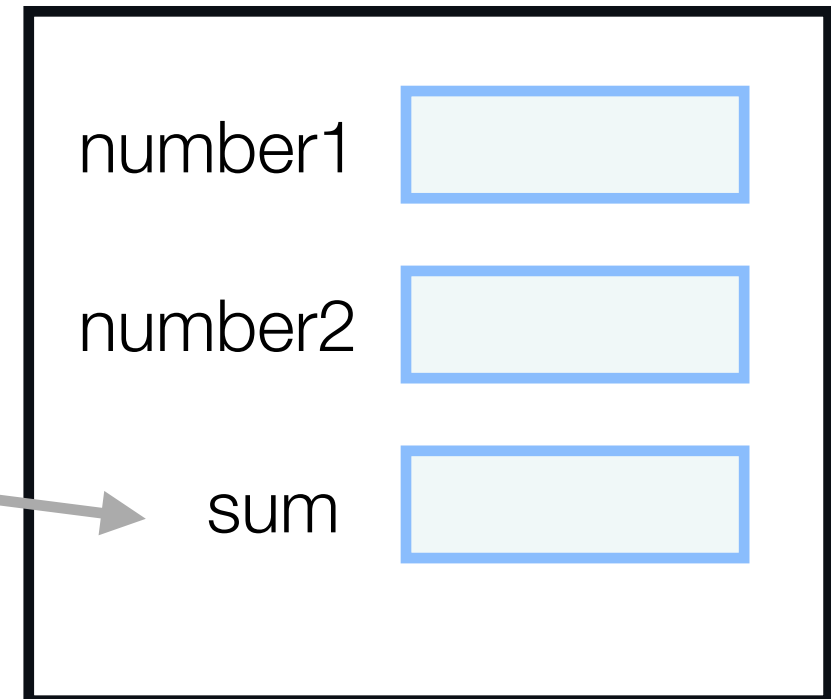
number2 için hafızadan yer ayır



Hafızada değişkenler

```
Addition.java x
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28 }
```

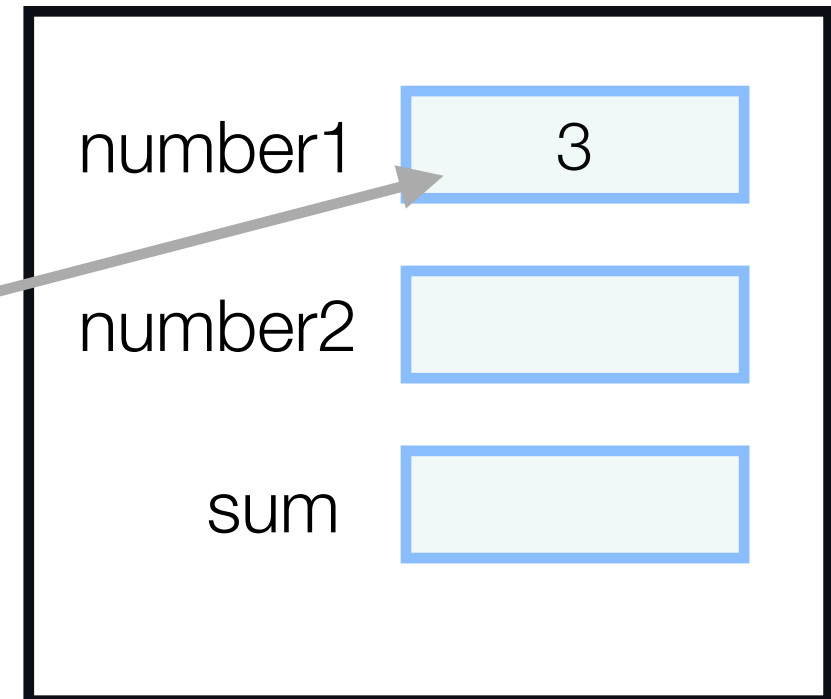
sum için hafızadan yer ayır



Hafızada değişkenler

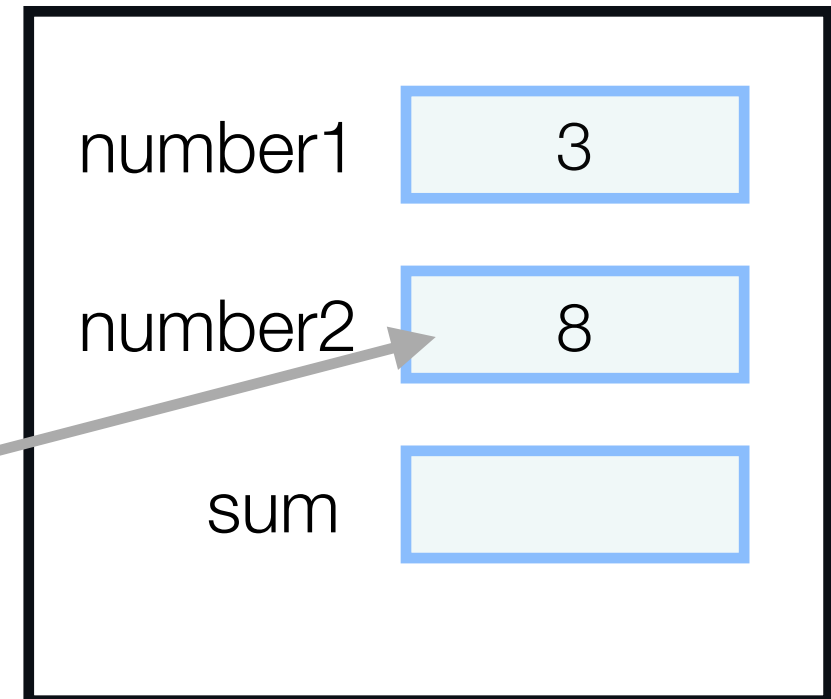
```
1 //Program for displaying the sum of two integers
2 public class Addition {
3     public static void main(String args[]) {
4         //first integer to add
5         int number1;
6
7         //second integer to add
8         int number2;
9
10        //sum of two integers
11        int sum;
12
13        //assign number1
14        number1 = 3;
15
16        //assign number2
17        number2 = 8;
18
19        //Compute the sum
20        sum = number1 + number2;
21
22        //Print the sum
23        System.out.print("Sum is " + sum);
24    }
25 }
26
27
28
```

3 değerini ata



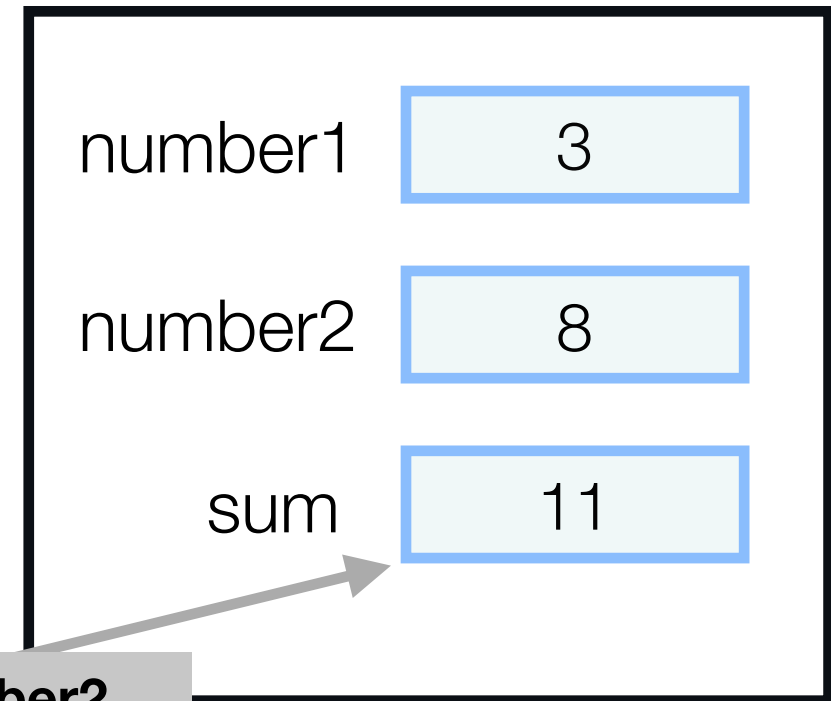
Hafızada değişkenler

```
Addition.java x
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28 }
```



Hafızada değişkenler

```
Addition.java x
1
2 //Program for displaying the sum of two integers
3 public class Addition {
4
5     public static void main(String args[]) {
6         //first integer to add
7         int number1;
8
9         //second integer to add
10        int number2;
11
12        //sum of two integers
13        int sum;
14
15        //assign number1
16        number1 = 3;
17
18        //assign number2
19        number2 = 8;
20
21        //Compute the sum
22        sum = number1 + number2;
23
24        //Print the sum
25        System.out.print("Sum is " + sum);
26
27    }
28 }
```



number1 + number2
değerini hesapla ve sum
değişkenine ata

Hafızada değişkenler

```
SuccessiveIncrease.java x
1
2 //Program for increasing the value of
3 // a number successively
4 public class SuccessiveIncrease {
5     public static void main(String args[]) {
6
7         int sum;
8
9         sum = 0;
10
11        sum = sum + 3;
12
13        sum += 7;
14
15        System.out.printf("Total sum is %d", sum);
16
17    }
18
19 }
```

sum için hafızadan yer ayır



Hafızada değişkenler

SuccessiveIncrease.java

```
1
2 //Program for increasing the value of
3 // a number successively
4 public class SuccessiveIncrease {
5     public static void main(String args[]) {
6
7         int sum;
8
9         sum = 0;
10
11        sum = sum + 3;
12
13        sum += 7;
14
15        System.out.printf("Total sum is %d", sum);
16
17    }
18
19 }
```

0 değerini ata

sum 0



Hafızada değişkenler

```
SuccessiveIncrease.java ×  
1  
2 //Program for increasing the value of  
3 // a number successively  
4 public class SuccessiveIncrease {  
5     public static void main(String args[]) {  
6  
7         int sum;  
8  
9         sum = 0;  
10  
11         sum = sum + 3;  
12  
13         sum += 7;  
14  
15         System.out.printf("Total sum is %d", sum);  
16  
17     }  
18  
19 }
```



3 değerini ata

sum = 0 + 3

Hafızada değişkenler

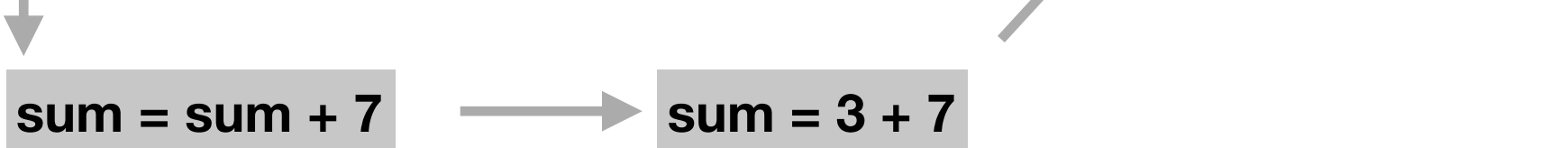
```
SuccessiveIncrease.java x
1
2 //Program for increasing the value of
3 // a number successively
4 public class SuccessiveIncrease {
5     public static void main(String args[]) {
6
7         int sum;
8
9         sum = 0;
10
11        sum = sum + 3;
12
13        sum += 7;
14
15        System.out.printf("Total sum is %d", sum);
16
17    }
18
19 }
```

sum = sum + 7

sum = 3 + 7

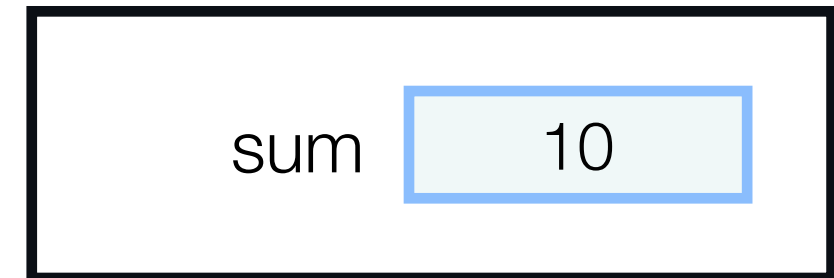


10 değerini ata



Hafızada değişkenler

```
SuccessiveIncrease.java x
1
2 //Program for increasing the value of
3 // a number successively
4 public class SuccessiveIncrease {
5     public static void main(String args[]) {
6
7         int sum;
8
9         sum = 0;
10
11        sum = sum + 3;
12
13        sum += 7;
14
15        System.out.printf("Total sum is %d", sum);
16
17    }
18
19 }
```



Total sum is 10

Farklı nümerik veri tipleri arasında işlemler

- İki değişken arasında yapılan işlemlerde
 - Eğer değişkenlerden biri double ise diğeri de double'a çevirilir.
 - Değil ise; eğer biri float ise diğeri de float'a çevirilir
 - Değil ise; eğer biri long ise diğeri de long'a çevirilir.
 - Değil ise ikisi de int'e çevirilir.

Veri tipi çevirme (Casting)

aralık genişler



byte, short, int, long, float, double

Veri tipi genişletme: Daha dar bir veri tipine sahip değişken daha geniş bir veri tipine otomatik olarak çevirilir.

Aşağıdaki çevirme ifadeleri doğru mu yanlış mı?

```
int x = 2;
```

```
double y = x;
```

```
double a = 2.0;
```

```
int b = a;
```

Veri tipi çevirme (Casting)

aralık genişler



byte, short, int, long, float, double

Veri tipi genişletme: Daha dar bir veri tipine sahip değişken daha geniş bir veri tipine otomatik olarak çevirilir.

Aşağıdaki çevirme ifadeleri doğru mu yanlış mı?

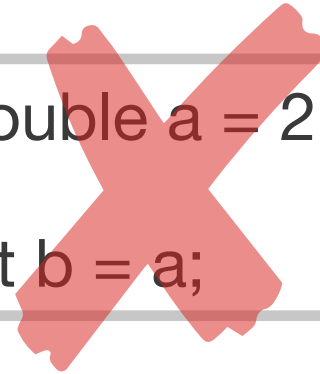
```
int x = 2;
```

```
double y = x;
```



```
double a = 2.0;
```

```
int b = a;
```



Veri tipi çevirme (casting)

aralık genişler



byte, short, int, long, float, double

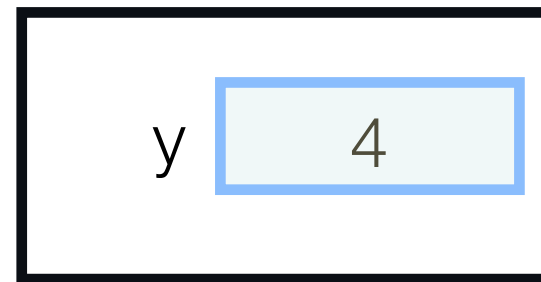
Veri tipi daraltma:

```
double a = 2.0;  
int b = a
```

```
double a = 2.0;  
int b = (int)a;
```



```
int y = (int)4.8;
```



virgülden sonrası yok olur.

Basit(Primitif) Veri Tipleri : char

Karakter değişken tipi:

- Alfabenin harfleri
- rakamlar
- + \$. , - ! gibi semboller

için kullanılır.

```
char letter = 'A';
```

Farklı karakter kodlama yolları (character encodings)

Değeri A olan char tipinde bir değişken tanımlamak istediğimizde farklı yollar kullanabiliriz:

- `char letter = 'A'` (ASCII)
- `char letter = '\u0041'` (UNICODE)

ASCII karakter kodlama

- ASCII kodlamasına göre, sıklıkla kullanılan karakterlere karşılık gelen tamsayı değerler vardır.

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040	 	Space	64	40	100	@	@
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C
4	4	004	EOT (end of transmission)	36	24	044	$	\$	68	44	104	D	D
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G
8	8	010	BS (backspace)	40	28	050	((72	48	110	H	H
9	9	011	TAB (horizontal tab)	41	29	051))	73	49	111	I	I
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[[
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135]]
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_
									127	7F	177		DEL

ASCII karakter kodlama

- Farklı karakterler de ASCII'nin genişletilmiş kodlama sistemiyle kodlanabilir. ASCII toplamda 2⁸ farklı karakter kod

128	Ç	144	É	160	á	176	☼	192	⌞	208	⌚	224	α	240	≡
129	ü	145	æ	161	í	177	☽	193	⌟	209	⌛	225	β	241	±
130	é	146	Æ	162	ó	178	☼	194	⌠	210	⌜	226	Γ	242	≥
131	â	147	ô	163	ú	179		195	⌡	211	⌝	227	π	243	≤
132	ä	148	ö	164	ñ	180	†	196	—	212	⌞	228	Σ	244	∫
133	à	149	ò	165	Ñ	181	‡	197	+	213	⌟	229	σ	245	∫
134	â	150	û	166	ª	182	‡	198	†	214	⌠	230	μ	246	+
135	ç	151	ù	167	º	183	π	199	‡	215	‡	231	τ	247	≈
136	ê	152	ÿ	168	¿	184	‡	200	⌞	216	‡	232	Φ	248	°
137	ë	153	Ö	169	¡	185	‡	201	⌟	217	∫	233	⊖	249	·
138	è	154	Ü	170	¬	186	‡	202	⌞	218	∫	234	Ω	250	·
139	ì	155	◊	171	½	187	‡	203	‡	219	■	235	δ	251	√
140	î	156	£	172	¼	188	‡	204	‡	220	■	236	∞	252	∞
141	ï	157	¥	173	¡	189	‡	205	=	221	■	237	φ	253	²
142	Ä	158	Ⓜ	174	«	190	∫	206	‡	222	■	238	ε	254	■
143	Å	159	f	175	»	191	∫	207	⌞	223	■	239	∩	255	

Source: www.LookupTables.com

- Ancak Dünyadaki dil ve karakter çeşitliliğini düşününce ASCII yetersiz kalmaktadır.

UNICODE karakter kodlama

- UNICODE karakter kodlaması çok daha geniş bir karakter aralığı sunar:
 - <https://unicode-table.com/en/#control-character>
- Java Dünya dillerinin çoğunu destekleyecek 16 bitlik Unicode karakter kümesini kullanır.
- Örneğin ekrana aşağıdaki Çince karakteri yazdırmak isterseniz, karakterin Unicode karşılığını bulup aşağıdaki ifadedeki gibi kullanabilirsiniz:

```
System.out.println( '\u5F08' );
```

弈

UNICODE karakter kodlama

- UNICODE karakter kodlaması çok daha geniş bir karakter aralığı sunar:
 - <https://unicode-table.com/en/#control-character>
- Örneğin ekrana aşağıdaki Çince karakteri yazdırmak isterseniz, karakterin Unicode karşılığını bulup aşağıdaki ifadedeki gibi kullanabilirsiniz:

```
System.out.println( '\u5F08' );
```

弈

Karakterlerin nümerik karşılıkları

- `int value = 'A'; // ifadesi value tamsayı değişkenine 65 değerini atar.`
- `char letter = 65; // ifadesi letter karakter değişkenine A değerini atar.`
- char tipindeki değişkenler arttırma/azaltma operatörleri ve kısayol atama operatörleriyle kullanılabilirler.
- Örneğin
 - `char letter = 'a';`
`letter++; // bu ifade letter değişkenin değerini 'a' dan bir sonraki karakter yani 'b' yapar`
 - `char letter = 'e';`
`letter-=2; // bu ifade letter değişkenin değerini 'e' den iki önceki karakter yani 'c' yapar`

Özel karakterler (Escape characters)

```
1
2 //Text-printing with multiple lines with a single statement
3 ▶ public class Welcome3 {
4
5     //main method begins execution of Java application
6 ▶     public static void main(String args[]) {
7
8         System.out.println("Welcome\nJava\nProgramming");
9
10    } //end method main
11 } //end class Welcome
```

```
welcome
to
Java
Programming!
```

Her \n özel karakterinden sonra çıktıda yeni satıra geçilir.

Özel karakterler (Escape sequences)

Özel Karakter	Tanımı
<code>\n</code>	yeni satır
<code>\t</code>	tab
<code>\\</code>	backslash
<code>\'</code>	tek tırnak
<code>\"</code>	çift tırnak
<code>\b</code>	backspace

Dizgi(String) veri tipi

- Karakter dizileridir.
- Primitif veri tipi değildir.

```
String message = "Hello World";
```

- Stringler birbirleriyle ve diğer veri tipleriyle + operatörü kullanılarak toplanabilirler.

```
int order = 1;  
String message = order+"st message";  
System.out.println(message);
```

Çıktı

```
1st message
```

Basit(Primitif) Veri Tipleri : boolean

boolean veri tipleri

- Mantıksal değerleri temsil etmek için vardır.
- İki farklı değer alabilir: true ve false

```
boolean var = true;
```

```
boolean var = false;
```

- boolean veri tiplerinin kullanımını ve boolean operatörleri bir sonraki ders detaylı öğreneceğiz.

Daire alanı hesaplama programı

Problem: Yarıçapı kullanıcı tarafından girilen dairenin alanını hesaplayıp, yarıçapıyla birlikte ekrana yazdırma.

Neler öğreneceğiz:

- final niteleyicisi
- Scanner sınıfıyla kullanıcı girdisi okuma

Daire alanı hesaplama programı

```
DiskArea.java x
1
2 import java.util.Scanner;
3
4 /*
5 Program that reads a radius from the user
6 and calculates the disk area with that radius
7 */
8
9 public class DiskArea {
10
11     public static void main(String args[]) {
12
13         //radius of the disk
14         double radius;
15
16         //area of the disk
17         double area;
18
19         //constant pi value
20         final double PI = 3.14;
21
22         //create Scanner to obtain input from command window
23         Scanner input = new Scanner(System.in);
24
25         System.out.print("Enter the radius: ");
26
27         //read radius from user
28         radius = input.nextDouble();
29
30         //calculate disk area
31         area = PI * radius * radius;
32
33         //Print radius and area to the command window
34         System.out.printf("\n Area of the disk with radius %.2f is %.2f ",radius,area);
35
36     }
```

final niteleyicisiyle
double tipinde sabit
PI değişkenini
tanımla

Daire alanı hesaplama programı

```
DiskArea.java x
1
2 import java.util.Scanner;
3
4 /*
5  Program that reads a radius from the user
6  and calculates the disk area with that radius
7  */
8
9 public class DiskArea {
10
11     public static void main(String args[]) {
12
13         //radius of the disk
14         double radius;
15
16         //area of the disk
17         double area;
18
19         //constant pi value
20         final double PI = 3.14;
21
22         //create Scanner to obtain input from command window
23         Scanner input = new Scanner(System.in);
24
25         System.out.print("Enter the radius: ");
26
27         //read radius from user
28         radius = input.nextDouble();
29
30         //calculate disk area
31         area = PI * radius * radius;
32
33         //Print radius and area to the command window
34         System.out.printf("\n Area of the disk with radius %.2f is %.2f ",radius,area);
35
36     }
```

Scanner nesnesi oluşturmak için bu paketin import edilmesi gerekli

Kullanıcıdan girdi okumak için input isimli Scanner nesnesini tanımla

Daire alanı hesaplama programı

```
DiskArea.java x
1
2 import java.util.Scanner;
3
4 /*
5 Program that reads a radius from the user
6 and calculates the disk area with that radius
7 */
8
9 public class DiskArea {
10
11     public static void main(String args[]) {
12
13         //radius of the disk
14         double radius;
15
16         //area of the disk
17         double area;
18
19         //constant pi value
20         final double PI = 3.14;
21
22         //create Scanner to obtain input from command window
23         Scanner input = new Scanner(System.in);
24
25         System.out.print("Enter the radius: ");
26
27         //read radius from user
28         radius = input.nextDouble();
29
30         //calculate disk area
31         area = PI * radius * radius;
32
33         //Print radius and area to the command window
34         System.out.printf("\n Area of the disk with radius %.2f is %.2f ",radius,area);
35
36     }
```

*Kullanıcıyı gireceği
değerle ilgili
bilgilendiren mesajı
yazdır*

Daire alanı hesaplama programı

```
DiskArea.java x
1
2 import java.util.Scanner;
3
4 /*
5 Program that reads a radius from the user
6 and calculates the disk area with that radius
7 */
8
9 public class DiskArea {
10
11     public static void main(String args[]) {
12
13         //radius of the disk
14         double radius;
15
16         //area of the disk
17         double area;
18
19         //constant pi value
20         final double PI = 3.14;
21
22         //create Scanner to obtain input from command window
23         Scanner input = new Scanner(System.in);
24
25         System.out.print("Enter the radius: ");
26
27         //read radius from user
28         radius = input.nextDouble();
29
30         //calculate disk area
31         area = PI * radius * radius;
32
33         //Print radius and area to the command window
34         System.out.printf("\n Area of the disk with radius %.2f is %.2f ",radius,area);
35
36     }
```

*Kullanıcının girdiği
double değeri radius
değişkenine ata*

Daire alanı hesaplama programı

```
DiskArea.java x
1
2 import java.util.Scanner;
3
4 /*
5 Program that reads a radius from the user
6 and calculates the disk area with that radius
7 */
8
9 public class DiskArea {
10
11     public static void main(String args[]) {
12
13         //radius of the disk
14         double radius;
15
16         //area of the disk
17         double area;
18
19         //constant pi value
20         final double PI = 3.14;
21
22         //create Scanner to obtain input from command window
23         Scanner input = new Scanner(System.in);
24
25         System.out.print("Enter the radius: ");
26
27         //read radius from user
28         radius = input.nextDouble();
29
30         //calculate disk area
31         area = PI * radius * radius;
32
33         //Print radius and area to the command window
34         System.out.printf("\n Area of the disk with radius %.2f is %.2f ",radius,area);
35
36     }
```

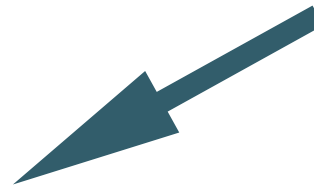
Alanı hesaplayıp area değişkenine ata



Daire alanı hesaplama programı

```
DiskArea.java x
1
2 import java.util.Scanner;
3
4 /*
5 Program that reads a radius from the user
6 and calculates the disk area with that radius
7 */
8
9 public class DiskArea {
10
11     public static void main(String args[]) {
12
13         //radius of the disk
14         double radius;
15
16         //area of the disk
17         double area;
18
19         //constant pi value
20         final double PI = 3.14;
21
22         //create Scanner to obtain input from command window
23         Scanner input = new Scanner(System.in);
24
25         System.out.print("Enter the radius: ");
26
27         //read radius from user
28         radius = input.nextDouble();
29
30         //calculate disk area
31         area = PI * radius * radius;
32
33         //Print radius and area to the command window
34         System.out.printf("\n Area of the disk with radius %.2f is %.2f ",radius,area);
35
36     }
```

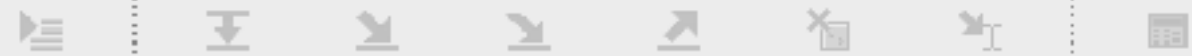
Yarıçapı ve alanı
ekrana yazdır



Daire alanı hesaplama programı

1

Debugger Console



```
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/  
Connected to the target VM, address: '127.0.0.1:51435', tr  
Enter the radius:
```



Daire alanı hesaplama programı

1

Debugger Console

```
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/  
Connected to the target VM, address: '127.0.0.1:51435', tr  
Enter the radius:
```

2

Debugger Console

```
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/  
Connected to the target VM, address: '127.0.0.1:51435', tr  
Enter the radius: 4
```

Daire alanı hesaplama programı

1

```
Debugger Console →  
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/  
Connected to the target VM, address: '127.0.0.1:51435', tr  
Enter the radius:
```

2

```
Debugger Console →  
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/  
Connected to the target VM, address: '127.0.0.1:51435', tr  
Enter the radius: 4
```

3

```
Debugger Console →  
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/  
Connected to the target VM, address: '127.0.0.1:51460', tr  
Enter the radius: 4  
Disconnected from the target VM, address: '127.0.0.1:51460'  
Area of the disk with radius 4.00 is 50.24  
Process finished with exit code 0
```

Scanner ile kullanııcıdan girdi okuma

```
Scanner input = new Scanner(System.in);  
radius = input.nextDouble();
```

- next(), nextByte(), nextShort(), nextInt(), nextLong(), nextFloat(), nextDouble() veya nextBoolean() metodlarını kullanarak byte, short, int, long, float, double veya boolean tipinde veri okuyabiliriz.

Scanner ile kullanııcıdan girdi okuma

```
Scanner input = new Scanner(System.in);  
radius = input.nextDouble();
```

- next(), nextByte(), nextShort(), nextInt(), nextLong(), nextFloat(), nextDouble() veya nextBoolean() metodlarını kullanarak byte, short, int, long, float, double veya boolean tipinde veri okuyabiliriz.

Alıřtırmalar

1. Kullanıcıdan saniye cinsinden zaman girmesini isteyen ve girilen zamanı saat, dakika ve saniyeye çevirip ekrana yazdıran bir Java programı yazınız. Örneğın kullanıcı 3668 girdiğinde program ekrana H:1 M:1 S:8 yazdırmalı.
2. Kullanıcıdan derece cinsinden bir açı isteyen ve bu açıyı radian cinsine çevirip ekrana yazdıran bir program yazınız. (derece = $180 \cdot \text{Pi} / \text{radyan}$; Pi deęerini sabit tanımlayınız(3,14))
3. Kullanıcıdan iki kelimeli bir cümle isteyen ve girilen cümlenin kelimelerini alt alta komut penceresine yazdıran bir program yazınız.
4. Verilen TL cinsinden parayı 50, 20, 10, 5, 1 cinsinden adet olarak en az para olacak şekilde bozduran programı yazınız. Program verilen miktarın 50TL cinsinden en fazla kaç para ile ödenebileceğini, arta kalanı 20 TL - >10 TL -> 5TL ve en son kalan parayı 1 TL olarak hesaplamalıdır

Alıştırmalar

5. Kullanıcıdan tek tek iki düzlemsel noktanın koordinatlarını isteyen ve girilen bu noktalar arasındaki uzaklığı hesaplayıp ekrana yazdıran bir program yazınız. (Noktaların koordinatları ve uzaklık virgülden sonra 2 basamaklı formatta yazılmalıdır, bunun için printf komutunu kullanınız). Programın çıktısı aşağıdaki gibi olmalıdır.

```
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/bin/java -agentlib:jdwp=transport=dt_socket,address=localhost:51017,server=y -jar DistanceCalculator.jar  
Connected to the target VM, address: '127.0.0.1:51017', transport: 'socket'  
Enter first point coordinates seperated with comma:
```

```
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/bin/java -agentlib:jdwp=transport=dt_socket,address=localhost:51017,server=y -jar DistanceCalculator.jar  
Connected to the target VM, address: '127.0.0.1:51017', transport: 'socket'  
Enter first point coordinates seperated with comma: 1 , 2  
Enter second point coordinates seperated with comma: |
```

```
/Library/Java/JavaVirtualMachines/jdk-9.jdk/Contents/Home/bin/java -agentlib:jdwp=transport=dt_socket,address=localhost:51017,server=y -jar DistanceCalculator.jar  
Connected to the target VM, address: '127.0.0.1:51017', transport: 'socket'  
Enter first point coordinates seperated with comma: 1 , 2  
Enter second point coordinates seperated with comma: 3 , 7  
Distance between (1.00,2.00) and (3.00,7.00) is 5.39  
Disconnected from the target VM, address: '127.0.0.1:51017', transport: 'socket'  
  
Process finished with exit code 0
```