

Bölüm 6

Java Operatörleri

Java, değişkenleri işlemek için çok sayıda operatör bulundurur. Java operatörlerini aşağıdaki gibi gruplandırabiliriz.

- Aritmetik operatörler
- İlişkisel operatörler
- Binary operatörler
- Mantıksal operatörler
- Atama operatörleri
- Diğer Operatörler

6.1 Aritmetik Operatörler

Aşağıdaki tablo aritmetik operatörleri listeler. İnteger A değişkeninin 10, B değişkenin 20 değerini aldığını varsayalım.

Operator	Description	Example
+	Toplama: Operatörün iki tarafındaki değeri birbiriyle toplar	A + B 30 değerini verecektir.
-	Çıkartma: : Operatörün iki tarafındaki değeri birbirinden çıkartır.	A – B -10 değerini verecektir
*	Çarpma: Operatörün iki tarafındaki değeri birbiriyle çarpar	A * B 200 değerini verecektir
/	Bölme: Operatörün iki tarafındaki değeri birbirine böler	B / A 2 değerini verecektir.
%	Mod alma: Operatörün solundaki değişkenin; sağındakine göre modunu alıp kalanı verir.	B % A 0 değerini verecektir.
++	Arttırma: Değişkenin değerini 1 arttırır	B++ 21 değerini verecektir.

--	Azaltma: Değişkenin değerini 1 azaltacaktır.	B-- 19 değerini verecektir
----	--	----------------------------

6.2 İlişkisel Operatörler

Aşağıdaki ilişkisel operatörler Java tarafından desteklenmektedir. İnteger A değişkeninin 10, B değişkeninin 20 değerini aldığını varsayalım.

Operator	Description	Example
==	İki değişkenin değerini kontrol eder,eğer eşitse true değerini döndürür.	(A == B) False değer döner..
!=	İki değişkenin değerini kontrol eder,eğer eşit değilse true değerini döndürür.	(A != B) True değer döner..
>	Operatörün solundaki değişkenin, sağındakinden büyük olup olmadığını kontrol eder. Eğer büyükse true değer döner.	(A > B) False değer döner.
<	Operatörün solundaki değişkenin, sağındakinden küçük olup olmadığını kontrol eder. Eğer küçükse true değer döner.	(A < B) true değer döner
>=	Operatörün solundaki değişkenin, sağındakine eşit veya büyük olup olmadığını kontrol eder. Eğer öyleyse true değer döner.	(A >= B) False değer döner.
<=	Operatörün solundaki değişkenin, sağındakine eşit veya küçük olup olmadığını kontrol eder. Eğer öyleyse true değer döner.	(A <= B) true değer döner

6.3 Binary operatörler

- Java integer,long,short,char ve byte tipleri için uygulanabilen birkaç Binary operatör sunar.
- Binary operatörler bitler üstünde çalışır ve bit-bit işlem yapar. a=60 ve b=13 diyelim. Bunların binary formları aşağıdaki gibidir.

a = 0011 1100 b = 0000 1101

$a \& b$	=	0000 1100
$a b$	=	0011 1101
$a \wedge b$	=	0011 0001
$\sim a$	=	1100 0011

Aşağıdaki tablo Binary operatörleri göstermektedir. int A'nın 60 , B'nin 13 değerlerini aldığı varsayalım.

Operator	Description	Example
&	Binary AND operatörü iki tarafta da bulunan biti sonuca kopyalar.	(A & B) 12 değerini verecektir. Bu da 0000 1100
	Binary OR operatörü iki taraftan birinde bulunan biti sonuca kopyalar.	(A B) 61 değerini verecektir. Bu da 0011 1101
^	Binary XOR operatörü bir tarafta bulunup diğer tarafta bulunmayan biti sonuca kopyalar.	(A ^ B) 49 değerini verecektir. Bu da 0011 0001
~	Binary 1'in tümleyeni (Ones Complement) operatörü. Sayının bütün 1 lerini sıfır, bütün sıfırlarını 1 yapar.	(~A) -60 değerini verecektir. Bu da 1100 0011
<<	Binary sola kaydırma operatörü. Operatörün solundaki değişkenin değeri sağdakine göre sola kaydırılır	A << 2 240 değerini verecektir. Bu da 1111 0000
>>	Binary sağa kaydırma operatörü. Operatörün solundaki değişkenin değeri soldakine göre sağa kaydırılır	A >> 2 15 değerini verecektir. Bu da 1111

6.4 Mantıksal Operatörler

Aşağıdaki tablo mantıksal operatörleri göstermektedir. boolean A'nın **true** , B'nin **false** değerlerini aldığını varsayalım.

Operator	Description	Example
&&	Mantıksal AND operatörü. Eğer iki taraftaki değişken de false değilse sonuç true döner	(A && B) false değer döner..
	Mantıksal OR operatörü. Eğer operatörün iki tarafındaki değişkenlerden birisi true ise sonuç true döner.	(A B) true değer döner.
!	Mantıksal NOT operatörü. Operasyonun mantıksal cevabını terse çevirir. Eğer true değer dönüyorsa, sonucu false değere çevirir.	!(A && B) true değer döner..

6.5 Atama Operatörleri

Aşağıdaki atama operatörleri Java dili tarafından desteklenmektedir.

Operator	Description	Example
=	Basit atama operatörüdür. Operatörün sağındaki değişkenin değeri solundaki(lere)'ne atanır.	C = A + B; A+B nin değerini C'ye atayacaktır.
+=	Operatörün sağındaki değeri solundaki değerle toplayıp tekrar solundaki değere atar.	C += A ; C = C + A değerine eşittir.
-=	Operatörün sağındaki değeri solundaki değerden çıkartıp tekrar solundaki değere atar.	C -= A; C = C - A değerine eşittir.
*=	Operatörün sağındaki değeri solundaki değerle çarpıp tekrar solundaki değere atar.	C *= A; C = C * A değerine eşittir.
/=	Operatörün sağındaki değeri solundaki değere bölüp tekrar solundaki değere atar.	C /= A; C = C / A değerine eşittir.
%=	Operatörün sağındaki değer solundakine göre modunu alıp tekrar solundaki değere atar.	C %= A; C = C % A değerine eşittir.

6.6 Koşul Operatörü (? :)

Koşul operatörü aynı zamanda üçlü operatörler olarak da bilinir. Bu operatör 3 işlenen kullanır ve boolean ifadelerin çözümleri için kullanılır. Bu operatörün amacı değişkene hangi değer atanacağına karar vermektir. Şu şekilde kodlanmaktadır:

```
variable x = (expression) ? value if true : value if false
```

ÖRNEK:

```
public class Test {  
  
    public static void main(String args[]){  
        int a , b;  
        a = 10;  
        b = (a == 1) ? 20: 30;  
        System.out.println( "Value of b is : " + b );  
  
        b = (a == 10) ? 20: 30;  
        System.out.println( "Value of b is : " + b );  
    }  
}
```

Aşağıdaki sonuç üretilecektir.

```
Value of b is : 30  
Value of b is : 20
```

6.7 “instanceOf” Operatörü

Bu operatör sadece nesne referans değişkenleri için kullanılır. Nesnenin belirtilen tipte olup olmadığını kontrol eder.(sınıf tipi veya arayüz tipi.) instanceof operatörü aşağıdaki gibi yazılır.

```
( Object reference variable ) instanceof (class/interface type)
```

ÖRNEK:

```
String name = "James";  
boolean result = name instanceof String;  
// This will return true since name is type of String
```