

**Aritmetik Operatörler (işlemler):**

Toplama : +  
 Çıkarma : -  
 Çarpma : \*  
 Bölme : /  
 Üs Alma : \*\*

**+ operatörü:** Toplama ve birleştirme için kullanılır.

10+20      Çıktı: 30  
 5.5+3.8    Çıktı: 9.3  
 9+4.0      Çıktı: 13.0

**Not:** Sayıların çift tırnak içine alınmadığına dikkat edin. Eğer çift tırnak içine alınsaydı veri türü bir integer değil string olacaktı. Bu durumda matematiksel işlem değil birleştirme işlemi yapılacaktır.

**Örnekler:**

"10"+"20"      Çıktı: '1020'  
 "5" + str(10)   Çıktı: '510'  
 5 + int("10")   Çıktı: 15

**- operatörü:**

50-30      Çıktı: 20

**\* operatörü:** Çarpma işleminin yanında stringleri belli sayıda tekrar etmek için kullanılır.

10\*5      Çıktı: 50  
 "w" \* 3    Çıktı: www  
 "-" \* 10   Çıktı: -----      # - işareti 10 kere çoğalttı.

**Not:** - eksi ve / bölü işlemlerini karakter dizileri ile birlikte kullanamayız.

**/ operatörü:** Bölme işlemi gerçekleştirir. Sonuç daima float veri türüdür.

21/3      Çıktı: 7.0  
 int(21/3)   Çıktı: 7      #burada veri integer'a dönüştürüldü.  
 21/0      Çıktı: hata      #bir sayı sifira bölünemez.

**// operatörü:** Bölme işleminde kalan sayı göz ardı edilir.(Taban Bölme)

25//6      Çıktı: 4

**\*\* operatörü:** Sayıların üssünü alır.

5\*\*2      Çıktı: 25

**Üs Alma İkinci Yol:**

pow(5,2)      Çıktı: 25

**% operatörü:** Mod işlemi. Bölme işlemi sonucunda kalan sayıyı verir.

25%7      Çıktı: 4

**Yuvarlama İşlemi:**

round(5.9)   Çıktı: 6  
 round(5.3)   Çıktı: 5

**Not:** Yuvarlama işleminde eğer 5 değeri ile karşılaşırsak en yakın çift sayıya yuvarlama yapmamız gerekir. Ör:

round(5.5) **Çıktı: 6** #en yakın çift sayı olan 6'ya yuvarlıyor.

round(4.5) **Çıktı: 4** #en yakın çift sayı kuralından dolayı sonuç 4 oluyor.

### Karşılaştırma Operatörleri:

Eşittir: ==  
Eşit değildir: !=  
büyüktür : >  
küçüktür: <  
büyük eşittir: >=  
küçük eşittir: <=

### Ör:

a=1		
a==1	<b>Çıktı: True</b>	#a 1'e eşittir?
a==2	<b>Çıktı: False</b>	#a 2'ye eşittir?
a!=5	<b>Çıktı: True</b>	#a 5'ye eşit değildir?
a!=1	<b>Çıktı: False</b>	#a 1'ye eşit değildir?
a>2	<b>Çıktı: False</b>	#a 2'den büyüktür?
8>a	<b>Çıktı: True</b>	#a 8'den büyüktür?
a>=1	<b>Çıktı: True</b>	#a 1'e eşit veya 1'den büyüktür?
a>=2	<b>Çıktı: False</b>	#a 2'ye eşit veya 1'den büyüktür?

**Bool Operatörleri:** and, or ve not operatörleridir. and-ve, or-veya, not-değil anlamı taşır.

### Ör:

a = 23  
b = 10  
a == 23 and b == 10  
**Çıktı: True**  
a == 23 and b == 56  
**Çıktı: False**  
a == 23 or b == 56  
**Çıktı: True**

**Not:** Kullanıcı tarafından bir değişkene veri girilip girilmediğini denetlemek için not operatörü kullanılabilir.

### Ör:

a = 19  
not a **#a'nın içinde değer yoktur (yanlış bir ifade)**  
**Çıktı: False**

a = ""  
not a  
**Çıktı: True**

```
a=0  
not a
```

**Çıktı: True**

**Aitlik Operatörü:** Aitlik operatörü, bir karakter dizisi ya da sayının, herhangi bir veri içinde bulunup bulunmadığını sorgular. Python 'da bir tane aitlik işleci bulunur. Bu işleç de "in" işlecidir.

**Ör:**

```
a = "Onur"  
"r" in a
```

# "b" karakteri a değişkeninin içinde var mı?

**Çıktı: True**

```
"f" in a
```

# "f" karakteri a değişkeninin içinde var mı?

**Çıktı: False**