

# عملگرهای مقایسه

در این درس، دربارهٔ عملگرهای مقایسه در پایتون آموزش خواهیم دید. این عملگرها به ما امکان می‌دهند متغیرها را مقایسه کرده و یک مقدار منطقی (True یا False) را خروجی دهند.

اگر شما تا حدی تجربه در ریاضیات داشته باشید، این عملگرها بسیار ساده خواهند بود.

ابتدا جدولی از عملگرهای مقایسه را ارائه می‌دهیم و سپس برخی مثال‌ها را بررسی خواهیم کرد:

## جدول عملگرهای مقایسه

در جدول زیر،  $a=3$  و  $b=4$  است.

عملگر	توضیحات	مثال
==	اگر مقادیر دو عملوند برابر باشند، شرط صحیح می‌شود.	$(a == b)$ درست نیست.
!=	اگر مقادیر دو عملوند برابر نباشند، شرط صحیح می‌شود.	$(a != b)$ درست است.
<	اگر مقدار عملوند چپ از مقدار عملوند راست بزرگتر باشد، شرط صحیح می‌شود.	$(a > b)$ درست نیست.
>	اگر مقدار عملوند چپ از مقدار عملوند راست کمتر باشد، شرط صحیح می‌شود.	$(a < b)$ درست است.
<=	اگر مقدار عملوند چپ از مقدار عملوند راست بزرگتر یا مساوی باشد، شرط صحیح می‌شود.	$(a >= b)$ درست نیست.
>=	اگر مقدار عملوند چپ از مقدار عملوند راست کمتر یا مساوی باشد، شرط صحیح می‌شود.	$(a <= b)$ درست است.
is	اگر علاوه بر برابر بودن مقدار عملوند چپ و عملوند آدرس ذخیره سازی مقادیر نیز با هم برابر باشد، شرط صحیح می‌شود.	$a = [1, 2, 3]$ $b = a$ $(a \text{ is } b)$ درست است
in	اگر مقدار عملوند چپ داخل دنباله (رشته، لیست، دیکشنری، تاپل، مجموعه) سمت راست باشد، شرط صحیح می‌شود.	$a = [1, 2, 3]$ $b = 3$ $(b \text{ in } a)$ درست است.

بیا ببینیم چند مثال با هم

برابری Equal

In [1]:

```
1 2 == 2
```

Out[1]:

True

In [2]:

```
1 1 == 0
```

Out[2]:

False

## بررسی برابری در اعداد اعشاری

In [1]:

```
1 x = 0.1 + 0.2
2
3 x == 0.3
```

Out[1]:

False

In [2]:

```
1 x
```

Out[2]:

0.30000000000000004

عجب !! دقت داشته باشید که محاسبات اعشاری در تمام زبان های برنامه نویسی به دلیل تبدیل خطای تبدیل اعداد اعشاری حقیقی به مبنای دو، دارای خطا خواهند بود. راه صحیح مقایسه برابر بودن اعداد اعشاری بصورت زیر می باشد.

In [6]:

```
1 tolerance = 0.001
2
3 x = 0.1 + 0.2
4
5 abs(x - 0.3) < tolerance
```

Out[6]:

True

توجه کنید که `==` یک عملگر مقایسه است، در حالی که `=` یک عملگر تخصیص است.

## Not Equal نابرابری

In [5]:

```
1 2 != 1
```

Out[5]:

True

In [6]:

```
1 2 != 2
```

Out[6]:

False

بزرگتر Greater Than

In [7]:

```
1 2 > 1
```

Out[7]:

True

In [8]:

```
1 2 > 4
```

Out[8]:

False

کوچکتر Less Than

In [9]:

```
1 2 < 4
```

Out[9]:

True

In [10]:

```
1 2 < 1
```

Out[10]:

False

بزرگتر مساوی Greater Than or Equal to

In [11]:

```
1 2 >= 2
```

Out[11]:

True

In [12]:

```
1 2 >= 1
```

Out[12]:

True

Less than or Equal to کوچکتر مساوی

In [13]:

```
1 2 <= 2
```

Out[13]:

True

In [14]:

```
1 2 <= 4
```

Out[14]:

True

عملگر عضویت در یک دنباله in

In [15]:

```
1 1 = [1,3,5,7,9]
```

In [16]:

```
1 1 in 1
```

Out[16]:

True

In [17]:

```
1 2 in 1
```

Out[17]:

False

In [18]:

```
1 2 not in 1
```

Out[18]:

True

In [19]:

```
1 l = [1,2,3,4]
2
3 l1 = [1,2,3,4]
4
5 l2 = l
```

In [20]:

```
1 id(l)
```

Out[20]:

2797747660160

In [21]:

```
1 id(l1)
```

Out[21]:

2797746714240

In [22]:

```
1 id(l2)
```

Out[22]:

2797747660160

In [23]:

```
1 l is l2
```

Out[23]:

True

In [24]:

```
1 l is l1
```

Out[24]:

False

In [25]:

```
1 l is not l1
```

Out[25]:

True

در ادامه درباره عملگرهای منطقی و زنجیره سازی عملگرهای مقایسه ای یاد میگیریم!