

متغیرها و دستورات کنترلی

متغیرها

نام متغیرها در پایتون فقط باید از حروف الفبا، اعداد و زیر خط تشکیل شده باشد و نمی‌تواند با اعداد شروع شود. هر متغیر باید قبل از استفاده تعریف شود. ولی متغیرها در پایتون نوع ثابتی ندارند و دارای نوع دینامیک هستند؛ به این معنی که می‌توان پس از تعریف، نوع آنها را تغییر داد و با کلمه‌ی کلیدی `del` می‌توان متغیرها را از حافظه پاک کرد؛ انگار اصلا وجود نداشته‌اند. پس از پاک کردن اگر آنها را فراخوانی کنید با همان خطایی مواجه می‌شوید که برای متغیرهای تعریف نشده رخ می‌دهد.

```
1 >>> a #not defined
2 Traceback (most recent call last):
3   File "<stdin>", line 1, in <module>
4 NameError: name 'a' is not defined
5 >>> a = 10 #defined of type int
6 >>> a
7 10
8 >>> type(a)
9 <class 'int'>
10 >>> a = "hi" #changed type to string
11 >>> a
12 'hi'
13 >>> type(a)
14 <class 'str'>
15 >>> del a #deleted from storage
16 >>> a #not it's not defined again
17 Traceback (most recent call last):
18   File "<stdin>", line 1, in <module>
19 NameError: name 'a' is not defined
```

سیستم نوشتاری پایتون

- پایتون بر خلاف زبان‌هایی مثل `C++` و `Java` سیستم نوشتاری خود را بر مبنای *indent* بنا کرده است؛ به این معنی که داخل `if` ، `for` یا تابع به جای استفاده از آکولاد با *indent* مشخص می‌شود. یعنی باید مقداری جلوتر از چهارچوب خود باشد.
- منظور از *tab* در اینجا تعدادی فاصله است و این مقدار می‌تواند در هر برنامه فرق کند، اما باید در یک فایل پایتون تعداد کاراکترهای *tab* برابر باشند و استاندارد این است که این مقدار ۴ فاصله باشد.

```
1 if a == b:
2     #if block
3
4 for i in range(10):
5     #for block
6
7 def a():
8     #function
```

شرط‌ها در پایتون غیر از تفاوت نوشتاری (*Syntax*) تفاوت دیگری با بقیه زبان‌ها ندارد. باید جلوی `if` یا `elif` یک گزاره باشد که جواب دودویی دارد. ابتدا شرط جلوی `if` بررسی می‌شود اگر درست بود وارد بلوک `if` می‌شود. در غیر این صورت به ترتیب برای `elif` ها اینکار را انجام می‌دهد و اگر هیچکدام درست نبودند وارد بلوک `else` می‌شود.

- **نکته:** به دلیل سیستم نوشتار پایتون، به جای `else if` در زبان‌های دیگر، باید از `elif` استفاده کرد.
- وجود `else` و `elif` ها اختیاری است.
- اگر بخواهید یک بلوک را خالی بگذارید، باید از کلمه‌ی `pass` استفاده کنید.

```
1 | if a == b :
2 |     print("they are equal")
3 | elif a < b:
4 |     pass
5 | else:
6 |     print("a is more than b")
```

شروط تک خطی

در پایتون دو نوع شرط تک خطی وجود دارد.

۱. شرط تک خطی به عنوان *statement* که شبیه شرط معمولی است.

```
1 | if condition : statement
```

```
1 | >>> a = 10
2 | >>> if a == 10 : print("a is 10")
3 | ...
4 | a is 10
```

این نوع فقط نوشتار تک خطی شروط معمولی است.

۲. شرط تک خطی جای *expression*

```
1 | expression_if_true if condition else expression_if_false
```

```
1 | >>> print("a is 10" if a==10 else "a is not 10")
2 | a is 10
3 | >>> b = "hi" if 5 > 10 else "bye"
4 | >>> b
5 | 'bye'
```

این نوع شرط فقط یک مقدار برمی‌گرداند که مثلا می‌توان آن را داخل متغیر ریخت.

