

Области видимости переменных

В Python переменную можно объявить в любом месте кода. Но от места, где объявлена переменная, зависит возможность обратиться к этой переменной.

Правила поиска переменной

Когда вы обращаетесь к переменной, интерпретатор python ищет её, руководствуясь двумя правилами.

Первое правило: если переменная объявлена внутри функции, обратиться к этой переменной снаружи нельзя.

```
def my_func():
    my_var = 'some text'
    print(my_var)

print(my_var) #'my_variable' is not defined

my_func()

print(my_var) #'my_variable' is not defined
```

Область видимости внутри функции называется **областью видимости функции**. Область видимости вне любых функций называется **глобальной**.

```
# Здесь глобальная область видимости
my_var = 1

def fnc1():
    # Здесь область видимости функции fnc1
    print(my_var+2) # возьмет из
                    # глобальной области
                    # видимости

def fnc2():
    # Здесь область видимости функции fnc2
    print(my_var+5) # возьмет из
                    # глобальной области
                    # видимости

fnc1()
fnc2()
```

Второе правило: если обратиться к переменной из функции, интерпретатор Python сначала будет искать данную переменную в теле этой функции. Если не найдёт — начнёт искать снаружи.

```
my_var1 = 5
my_var2 = 10

def my_func():
    my_var1=15

    # внутри функции переменная my_var1 определена,
    # и это значение оказывается в консоли
    print(my_var1)

    # внутри функции переменная my_var2 не определена,
    # поэтому интерпретатор взял значение снаружи
    print(my_var2)

my_func()
```

Переменную называют **глобальной**, если она объявлена в глобальной области видимости, и **локальной** — если внутри функции.

```
# Здесь глобальная область видимости
a = 1 # это глобальная переменная

def fnc():
    # Здесь область видимости функции fnc1
    b = 5 # это локальная переменная

fnc()
```

Вложенные области видимости

Области видимости «вкладываются» друг в друга:

```
# Здесь глобальная область видимости
a = 1

def fnc1():
    # Здесь область видимости функции fnc1
    b=2

    # функция fnc2 вложена в fnc1
    def fnc2():
        # Здесь область видимости функции fnc2
        c=3
        print(a)
        print(b)
        print(c)

    fnc2()

fnc1()
```

Порядок поиска переменных: сначала внутри `fnc2`, потом внутри `fnc1`, затем в глобальной области видимости.

Удаление переменной

Удалить переменную можно только в той области видимости, в которой она была объявлена

```
a = 1

def fnc():
    b = 5
    del b # переменная удалится
    del a # будет ошибка

fnc()
```

Затенение переменных

Если в теле функции обратиться к переменной, Python интерпретатор сначала ищет её в теле этой функции, и уже потом снаружи. Поэтому если и снаружи, и внутри есть переменные с одинаковыми именами, движок остановится на внутренней, а внешнюю — проигнорирует.

```
# Здесь глобальная область видимости
a = 1

def fnc1():
    # Здесь область видимости функции fnc1
    a=2
    print(a) # будет выведено 2,
              # так как внутренняя переменная
              # затеняет внешнюю

fnc1()
```