

# Python: print(), переменные и константы | Я.Шпора

## Функция `print()`

Функция `print()` выводит на экран информацию, которая была ей передана, например, строки, числа, результаты вычислений и так далее:

```
print('Привет! Это текст!')
```

```
# Вывод в терминал: Привет! Это текст!
```



Строки кода, которые начинаются с символа `#`, — это комментарии: примечания, которые автор программы пишет для других разработчиков. Python игнорирует любые символы, расположенные на строчке после `#`.

## Переменные

Переменная в Python — имя, связанное с определённым значением, которое можно использовать или менять в коде. В Python нет объявления переменных, официально это называется *assignment* — «назначение, связывание».

Чтобы создать переменную, надо указать имя переменной, поставить знак присваивания `=`, указать присваиваемое значение:

```
jar_of_vegetables = 'Огурцы'
```

## Как называть переменные

- Используйте осмысленные слова на английском языке и не применяйте транслитерацию слов с родного языка:

```
# Некорректное имя переменной.  
banka_ovoshchei = 'Огурцы'
```

```
# Хорошее имя переменной.  
jar_of_vegetables = 'Огурцы'
```

- Если название переменной состоит из нескольких слов, отделяйте эти слова друг от друга символом подчёркивания — `_`:

```
jar_of_vegetables = 'Огурцы'
```

- Старайтесь избегать однобуквенных имён и сокращений:

```
# Неудачные названия переменных.  
a = 1  
nm = 100  
  
# Хорошие названия переменных.  
amount = 1  
number = 100
```

## Как задать значение переменной

- Явно, с помощью литерала:

```
quantity_of_jars = 15
```

- Используя выражение:

```
total_quantity = quantity_of_jars + 10
```

## Тип данных str: строка

Python считает строкой любой набор символов, заключённый в двойные `"..."` или одинарные `'...'` кавычки, тройные двойные `"""..."""` или тройные одинарные `'''...'''` кавычки:

```
# Строка в одинарных кавычках.  
profession = 'Фермер'  
# Строка в двойных кавычках.  
name = "Тоня"
```

```
# Число в кавычках тоже становится строкой.  
age = '31'
```

## Тип данных int: целое число

Целочисленные значения записываются без кавычек:

```
quantity_of_jars = 15
```

## Преобразование типов

Для преобразования числа в строку используется функция `str()`:

```
int_number = 100  
str_number = str(int_number)
```

Для преобразования строки из цифр в целое число используется функция `int()`:

```
str_number = '100'  
int_number = int(str_number)
```

Преобразование в другой тип можно выполнить, не задействуя новую переменную, например, прямо в функции `print()`:

```
number = 100  
text = ' лет одиночества'  
  
print(str(number) + text)  
  
# Вывод в терминал: 100 лет одиночества
```

## Константы

Константа — это переменная, значение которой известно заранее и не должно меняться при выполнении программы.

В Python имя константы принято обозначать заглавными буквами:

```
PI = 3.14
```

