



Python Variables, Constants and Literals



B. Ramar(Ph.D)

Resource Faculty,

AU TVS Centre for Quality Management, Anna University.

Content



1. Variable

- a) **Declaring Variables in Python**
- b) **Assigning value to a Variable in Python**

2. Constants

- a) **Assigning value to a constant in Python**
- b) **Rules and Naming convention of Variables and Constants**

3. Literals

- a) **Numeric Literals**
- b) **String Literals**
- c) **Boolean Literals**
- d) **Special Literals**
- e) **Literal Collection**

Variable

Variable is a named location used to store data in the memory.

Statements:

if statement, for statement, while statement etc.

Declaring Variables in Python

Variables do not need declaration to reserve memory space. The "variable declaration" or "variable initialization" happens automatically when we assign a value to a variable.

Assigning value to a Variable in Python

You can use the assignment operator = to assign the value to a variable.

- 1. Declaring and assigning a value to a variable**
- 2. Changing value of a variable**
- 3. Assigning multiple values to multiple variables**

Constants

A constant is a type of variable whose value cannot be changed. It is helpful to think of constants as containers that hold information which cannot be changed later.

Assigning value to a constant in Python

Constants are usually declared and assigned on a module. Here, the module means a new file containing variables, functions etc which is imported to main file.

Declaring and assigning value to a constant

Create a constant.py



Create a main.py

Rules and Naming convention for variables and constants

1. Create a name that makes sense. Suppose, vowel makes more sense than v.
2. Use camelCase notation to declare a variable. It starts with lowercase letter.
3. Use capital letters where possible to declare a constant
4. Never use special symbols like !, @, #, \$, %, etc.
5. Don't start name with a digit.
6. Constants are put into Python modules and meant not be changed.
7. Constant and variable names should have combination of letters in lowercase (a to z) or uppercase (A to Z) or digits (0 to 9) or an underscore (_).

Numeric Literals

Literal is a raw data given in a variable or constant. In Python, there are various types of literals they are

Numeric Literals are immutable (unchangeable). Numeric literals can belong to 3 different numerical types,

1. Integer
2. Float
3. Complex.

```
a = 0b1010 #Binary Literals
b = 100 #Decimal Literal
c = 0o310 #Octal Literal
d = 0x12c #Hexadecimal Literal
```

```
#Float Literal
float_1 = 10.5
float_2 = 1.5e2
```

```
#Complex Literal
x = 3.14j
print(a, b, c, d)
print(float_1, float_2)
print(x, x.imag, x.real)
```

String Literals

We can use both single, double or triple quotes for a string.

How to use string literals in Python?

```
strings = "This is Python"
char = "C"
multiline_str = """This is a multiline string with more than
one line code."""
unicode = u"\u00dcnic\u00f6de"
raw_str = r"raw \n string"

print(strings)
print(char)
print(multiline_str)
print(unicode)
print(raw_str)
```

Boolean Literals

A Boolean literal can have any of the two values: True or False.

In the above program, we use boolean literal True and False. In Python, True represents the value as 1 and False as 0. The value of x is True because 1 is equal to True. And, the value of y is False because 1 is not equal to False.

```
x = (1 == True)
```

```
y = (1 == False)
```

```
a = True + 4
```

```
b = False + 10
```

```
print("x is", x)
```

```
print("y is", y)
```

```
print("a:", a)
```

```
print("b:", b)
```


Special Literals

special literal i.e. None. We use it to specify to that field that is not created.

```
drink = "Available"  
food = None
```

```
def menu(x):  
    if x == drink:  
        print(drink)  
    else:  
        print(food)
```

```
menu(drink)  
menu(food)
```

Literal Collections

There are four different literal collections List literals, Tuple literals, Dict literals, and Set literals.

```
fruits = ["apple", "mango", "orange"] #list
numbers = (1, 2, 3) #tuple
alphabets = {'a':'apple', 'b':'ball', 'c':'cat'} #dictionary
vowels = {'a', 'e', 'i', 'o', 'u'} #set

print(fruits)
print(numbers)
print(alphabets)
print(vowels)
```

Content



1. Variable

- a) **Declaring Variables in Python**
- b) **Assigning value to a Variable in Python**

2. Constants

- a) **Assigning value to a constant in Python**
- b) **Rules and Naming convention of Variables and Constants**

3. Literals

- a) **Numeric Literals**
- b) **String Literals**
- c) **Boolean Literals**
- d) **Special Literals**
- e) **Literal Collection**