
Python Variables: How to Define/Declare String Variable Types

By Steve Campbell ⌚ Updated May 3, 2022

What is a Variable in Python?

A Python variable is a reserved memory location to store values. In other words, a variable in a python program gives data to the computer for processing.

Python Variable Types

Every value in Python has a datatype. Different data types in Python are Numbers, List, Tuple, Strings, Dictionary, etc. Variables in Python can be declared by any name or even alphabets like a, aa, abc, etc.

In this tutorial, we will learn,

- [How to Declare and use a Variable](#)
- [Re-declare a Variable](#)
- [Concatenate Variables](#)
- [Local & Global Variables](#)
- [Delete a variable](#)

How to Declare and use a Variable

Let see an example. We will define variable in Python and declare it as “a” and print it.

This code is editable. Click Run to Execute

```
1 a=100
2 print (a)
```

Run

100

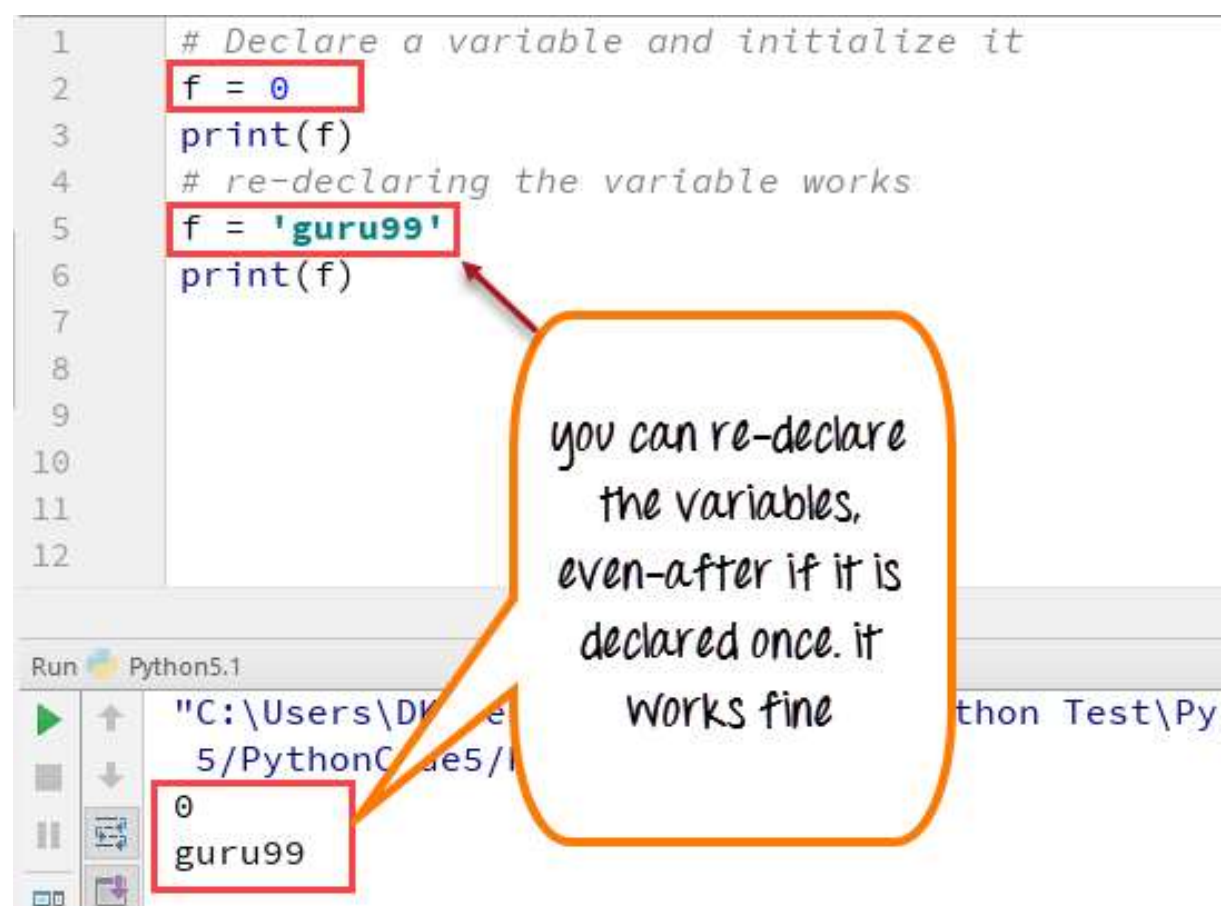
Re-declare a Variable

You can re-declare Python variables even after you have declared once.

Here we have Python declare variable initialized to f=0.

Later, we re-assign the variable f to value “guru99”





Python 2 Example

This code is editable. Click Run to Execute

```
1 # Declare a variable and initialize it
2 f = 0
3 print f
4 # re-declaring the variable works
5 f = 'guru'
6 print f
```

Run

Python 3 Example

This code is editable. Click Run to Execute

```
1 # Declare a variable and initialize it
2 f = 0
3 print(f)
4 # re-declaring the variable works
5 f = 'guru'
6 print(f)
```

Run

0

guru99



Python String Concatenation and Variable

Let's see whether you can concatenate different data types like string and number together. For example, we will concatenate "Guru" with the number "99".

Unlike Java, which concatenates number with string without declaring number as string, while declaring variables in Python requires declaring the number as string otherwise it will show a `TypeError`



For the following code, you will get undefined output –

```
a="Guru"
b = 99
print a+b
```

Once the integer is declared as string, it can concatenate both "Guru" + `str("99")`= "Guru99" in the output.

This code is editable. Click Run to Execute

```
1 a="Guru"
2 b = 1
3 print(a+str(b))
```

Run

Guru1

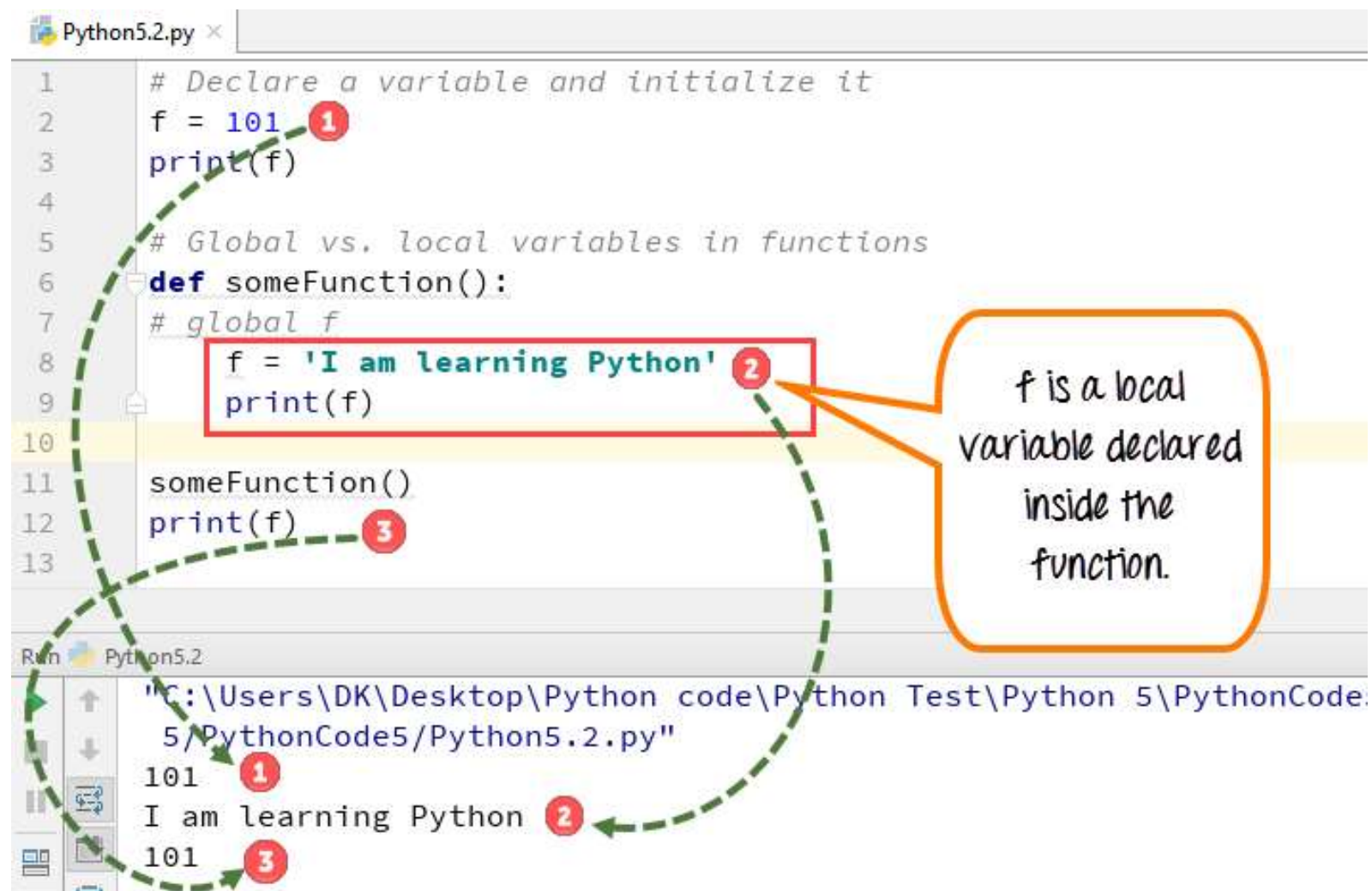
Python Variable Types: Local & Global

There are two types of variables in Python, Global variable and Local variable. When you want to use the same variable for rest of your program or module you declare it as a global variable, while if you want to use the variable in a specific function or method, you use a local variable while Python variable declaration.



Let's understand this Python variable types with the difference between local and global variables in the below program.

1. Let us define variable in Python where the variable “f” is **global** in scope and is assigned value 101 which is printed in output
2. Variable f is again declared in function and assumes **local** scope. It is assigned value “I am learning Python.” which is printed out as an output. This Python declare variable is different from the global variable “f” defined earlier
3. Once the function call is over, the local variable f is destroyed. At line 12, when we again, print the value of “f” is it displays the value of global variable f=101



Python 2 Example

This code is editable. Click Run to Execute

```
1 # Declare a variable and initialize it
2 f = 101
3 print f
4 # Global vs. local variables in functions
5 def someFunction():
6     # global f
7     f = 'I am learning Python'
8     print f
9 someFunction()
10 print f
```

Run

101

I am learning Python

101

Python 3 Example



This code is editable. Click Run to Execute

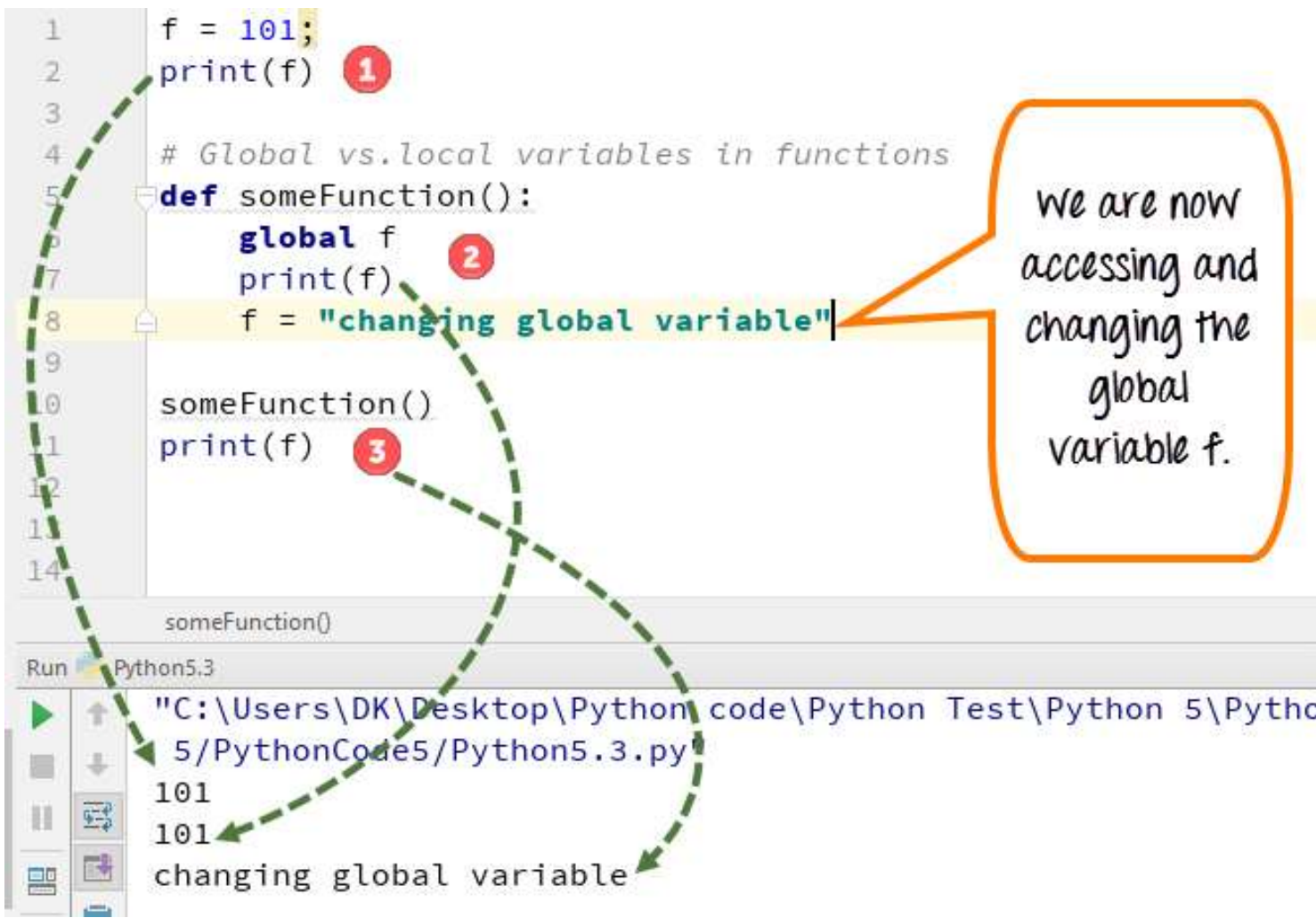
```
1 # Declare a variable and initialize it
2 f = 101
3 print(f)
4 # Global vs. local variables in functions
5 def someFunction():
6     # global f
7     f = 'I am learning Python'
8     print(f)
9 someFunction()
10 print(f)
```

Run

101
I am learning Python
101

While Python variable declaration using the keyword **global**, you can reference the global variable inside a function.

- 1. Variable “f” is **global** in scope and is assigned value 101 which is printed in output
- 2. Variable f is declared using the keyword **global**. This is **NOT** a **local variable**, but the same global variable declared earlier. Hence when we print its value, the output is 101
- 3. We changed the value of “f” inside the function. Once the function call is over, the changed value of the variable “f” persists. At line 12, when we again, print the value of “f” is it displays the value “changing global variable”



Python 2 Example

This code is editable. Click Run to Execute

```
1 f = 101;
2 print f
3 # Global vs.local variables in functions
4 def someFunction():
```



```
5 global f
6 print f
7 f = "changing global variable"
8 someFunction()
9 print f
```

Run

101

101

changing global variable

Python 3 Example

This code is editable. Click Run to Execute

```
1 f = 101;
2 print(f)
3 # Global vs.local variables in functions
4 def someFunction():
5     global f
6     print(f)
7     f = "changing global variable"
8 someFunction()
9 print(f)
```

Run

101

101

changing global variable

Delete a variable

You can also delete Python variables using the command **del** “variable name”.

In the below example of Python delete variable, we deleted variable f, and when we proceed to print it, we get error “**variable name is not defined**” which means you have deleted the variable.



```
1  #Declare a variable and initialize it
2  f = 11;
3  print(f)
4
5
6
7  del f
8  print(f)
9
```

Run Python5.4

"C:\Users\DK\Desktop\Python code\Python 5\PythonCode5\Python5.4.py"

11

Traceback (most recent call last):

File "C:/Users/DK/Desktop/Python code/Python 5/PythonCode5/Python5.4.py", line 8, in <module>

print(f)

NameError: name 'f' is not defined

Once you delete variable f and print f it will show this comment, which means your variable is now deleted

Example of Python delete variable or Python clear variable :

This code is editable. Click Run to Execute

```
1 f = 11;
2 print(f)
3 del f
4 print(f)
```

Run

11

Summary:

- Variables are referred to “envelop” or “buckets” where information can be maintained and referenced. Like any other programming language Python also uses a variable to store the information.
- Variables can be declared by any name or even alphabets like a, aa, abc, etc.
- Variables can be re-declared even after you have declared them for once
- Python constants can be understood as types of variables that hold the value which can not be changed. Usually Python constants are referenced from other files. Python define constant is declared in a new or separate file which contains functions, modules, etc.
- Types of variables in Python or Python variable types : Local & Global
- Declare local variable when you want to use it for current function
- Declare Global variable when you want to use the same variable for rest of the program
- To delete a variable, it uses keyword “del”.

