**VARIABLES**

**What is variable?**

variable is a name given to a memory location. It is the basic unit of storage in a program.

* The value stored in a variable can be changed during program execution.
* A variable is only a name given to a memory location, all the operations done on the variable effects that memory location.
* Variable also known as **identifier** and used to hold value.
* A variable, as the name indicates is something whose value is changeable over time. X = 10
* In Python, **we don't need to specify the type of variable (int, float)** because Python is a type infer language and smart enough to get variable type

**What are the rules for creating variables in Python:**

* A variable name must start with a letter or the underscore character.
* A variable name cannot start with a number.
* A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and \_ ).
* Variable names are case-sensitive (name, Name and NAME are three different variables).
* The reserved words(keywords) cannot be used naming the variable.

Eg:age=45 (age is variable,45 is int value)

Salary=1458.23 (salary is variable,1458.23 is float value)

Name=john (name is variable,john is string)

**What are the individual data types ?**

1. **Numbers:**

**integer :** 10, 123, 432

**float :** 32.43, 543.56, 3456.34

**long :** 7406900500, 324324324324(Removed)

**complex :** 4+5j(Will never use)

**binary 2 octa 8 deca 10 hex 16**

* 1. **Boolean:**

**True - 1 bit 1 ,**

**False - 1 bit 0**

**X=10 (write program)**

Steps:

1.load the statatement

2.Execution starts from R.H.S TO L.H.S

3.finds the type of variable

4 if expression ,then perform operation to get value.

5.final value turns into binary format and allocate the memory location.

6.memory location address given to the variable x.

Eg: X=10

Type int

Binary format 10=00001010

Memory location =16821827

**Print (x) Read program**

1.python will go the variable reffered address of x

2. Take the binary format of that value and converts to decimal format.

3. Gives (prints) the vaue to console.

## Variables

Variables are containers for storing data values.

## Creating Variables

Python has no command for declaring a variable.

A variable is created the moment you first assign a value to it.

### Example

x = 5  
y = "John"  
print(x)  
print(y)

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variables1)

Variables do not need to be declared with any particular type, and can even change type after they have been set.

### Example

x = 4       # x is of type int  
x = "Sally" # x is now of type str  
print(x)

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variables2)

## Casting

If you want to specify the data type of a variable, this can be done with casting.

### Example

x = str(3)    # x will be '3'  
y = int(3)    # y will be 3  
z = float(3)  # z will be 3.0

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variables_casting)

## Get the Type

You can get the data type of a variable with the type() function.

### Example

x = 5  
y = "John"  
print(type(x))  
print(type(y))

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variables_type)

You will learn more about [data types](https://www.w3schools.com/python/python_datatypes.asp) and [casting](https://www.w3schools.com/python/python_casting.asp) later in this tutorial.

## Single or Double Quotes?

String variables can be declared either by using single or double quotes:

### Example

x = "John"  
# is the same as  
x = 'John'

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variables7)

## Case-Sensitive

Variable names are case-sensitive.

### Example

This will create two variables:

a = 4  
A = "Sally"  
#A will not overwrite a

## Variable Names

A variable can have a short name (like x and y) or a more descriptive name (age, carname, total\_volume). Rules for Python variables:

* A variable name must start with a letter or the underscore character
* A variable name cannot start with a number
* A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and \_ )
* Variable names are case-sensitive (age, Age and AGE are three different variables)

### Example

Legal variable names:

myvar = "John"  
my\_var = "John"  
\_my\_var = "John"  
myVar = "John"  
MYVAR = "John"  
myvar2 = "John"

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variable_names_legal)

### Example

Illegal variable names:

2myvar = "John"  
my-var = "John"  
my var = "John"

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variable_names_error)

Remember that variable names are case-sensitive

## Multi Words Variable Names

Variable names with more than one word can be difficult to read.

There are several techniques you can use to make them more readable:

## Camel Case

Each word, except the first, starts with a capital letter:

myVariableName = "John"

## Pascal Case

Each word starts with a capital letter:

MyVariableName = "John"

## Snake Case

Each word is separated by an underscore character:

my\_variable\_name = "John"

## Many Values to Multiple Variables

Python allows you to assign values to multiple variables in one line:

### Example

x, y, z = "Orange", "Banana", "Cherry"  
print(x)  
print(y)  
print(z)

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variables8)

**Note:** Make sure the number of variables matches the number of values, or else you will get an error.

## One Value to Multiple Variables

And you can assign the same value to multiple variables in one line:

### Example

x = y = z = "Orange"  
print(x)  
print(y)  
print(z)

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variables6)

## Unpack a Collection

If you have a collection of values in a list, tuple etc. Python allows you extract the values into variables. This is called unpacking.

### Example

Unpack a list:

fruits = ["apple", "banana", "cherry"]  
x, y, z = fruits  
print(x)  
print(y)  
print(z)

## Output Variables

The Python print statement is often used to output variables.

To combine both text and a variable, Python uses the + character:

### Example

x = "awesome"  
print("Python is " + x)

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variables3)

You can also use the + character to add a variable to another variable:

### Example

x = "Python is "  
y = "awesome"  
z =  x + y  
print(z)

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variables4)

For numbers, the + character works as a mathematical operator:

### Example

x = 5  
y = 10  
print(x + y)

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variables5)

If you try to combine a string and a number, Python will give you an error:

### Example

x = 5  
y = "John"  
print(x + y)

## Global Variables

Variables that are created outside of a function (as in all of the examples above) are known as global variables.

Global variables can be used by everyone, both inside of functions and outside.

### Example

Create a variable outside of a function, and use it inside the function

x = "awesome"  
  
def myfunc():  
  print("Python is " + x)  
  
myfunc()

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variables_global)

If you create a variable with the same name inside a function, this variable will be local, and can only be used inside the function. The global variable with the same name will remain as it was, global and with the original value.

### Example

Create a variable inside a function, with the same name as the global variable

x = "awesome"  
  
def myfunc():  
  x = "fantastic"  
  print("Python is " + x)  
  
myfunc()  
  
print("Python is " + x)

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variables_global2)

## The global Keyword

Normally, when you create a variable inside a function, that variable is local, and can only be used inside that function.

To create a global variable inside a function, you can use the global keyword.

### Example

If you use the global keyword, the variable belongs to the global scope:

def myfunc():  
  global x  
  x = "fantastic"  
  
myfunc()  
  
print("Python is " + x)

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_variables_global3)

Also, use the global keyword if you want to change a global variable inside a function.

### Example

To change the value of a global variable inside a function, refer to the variable by using the global keyword:

x = "awesome"  
  
def myfunc():  
  global x  
  x = "fantastic"  
  
myfunc()  
  
print("Python is " + x)