Python Variables

Put it simple variables are the containers to store values in. In python variables of any type are object, we can say each and everything in python is an object, an object contains attributes/fields or data in variables and behaviour related to that data in methods. we can access attribute and methods related to object using dot opearator. let see an example,

In [1]:
```
x = 45.4
x.is_integer()
```

Out[1]:
```
False
```

Defining variable in python:

In [3]:
```
#we can't declare a variable without assigning it a value in python  as the following
#command will generate error
# var
```

In [3]:
```
#we can declare a variable without assigning it any value as,
value = None
print(value)
```

```
None
```

None is a specific type we can say a constant in python which indicate the absence of value, there are different scanerios when value can be None, for example when function returns nothing, or nothing passed in argument etc.

In [4]:
```
#we can assign value to a variable initialized with None as,
value = "Muhammad Nasir"
print(value)
```

```
Muhammad Nasir
```

Casting one type to another type

In [5]:
```
#we can convert one type into another type using different built-in appropriate function for example,
#convert string to int
val1 = int("99")
#coverting int to string
val2 = str(45)
#convert float to int
val3 = int(45.5)
print(val1)
print(val2)
print(val3)
```

```
99
45
45
```

In [6]:
```
#lets check type of above variables using type() function
print(type(val1))
print(type(val2))
print(type(val3))
```

```
<class 'int'>
<class 'str'>
<class 'int'>
```

In [8]:
```
#String variables can be defined using single and double quotes as well,
#in other languages single quotes are used for character data types but
#because in python there is no char type so '' or " " both used for string as,
fname = 'Muhammad'
lname = ' Nasir'
print(fname + lname)
```

```
Muhammad Nasir
```

Python is case sensitive

In [9]:
```
#Case Sensitive languages are ones which differ in lower case and upper case as the letter a is not
#same as capital A. python is case sensitive.
a = 34;
A = 56;
print(a)
print(A)
#a and A are two different variables for python
```

```
34
56
```

Rules for variable names in python

In [10]:
```
#1. variable name can contain only letter (lowercase + uppercase) digits (0-9) and underscore
#2. name can start with letters or underscore but can't start with a digit
```

Local and Global variable

Any variable which is created in program scope not in a function, it is known as global variable and can be access anywhere in program at any time during the lifetime of program.

In [13]:
```
x = 45
```

```
def myfunc():
 x=99
 print(x)
myfunc()

print("Python is " + str(x))
```

```
99
Python is 45
```

In above code the variable defined inside a function is local variable and can be accessed only inside a function and the variable defined outside function is global and it can be accessed any where inside function or out of function scope.

In [15]:
```
#to create a global variable inside a function use global keyword with variable as,
def myfunc():
  global x
```

In [ ]:
```

```