Variables

In [ ]:

```
Variables --> Variables are used to store a value
          --> Variables is something which is going to hold the memory location.
```

Example of Variable

In [39]:

```
x=10 # Here x is a variable in which is going to hold the memory location of 10
x
```

Out[39]:

```
10
```

In [40]:

```
y="Python" # Here y is a variable in which is going to hold the memory location of "Python"
y
```

Out[40]:

```
'Python'
```

Data Types

In [ ]:

```
Data + Types --> Represent which kind of data we are going to use in our program that we are going to
                 stored inside a variable.

Example:

    Name = "Pratyush"  --> Strings

    Mob  = 9876543210  --> Integer

    IsValid = True or False --> Boolean


Note --> In Python we need not to Declare or define the datatype explictly.Internally PVM will
         automatically Determine which kind of datatype  we are going to use in our Programming
         at the RunTime.

PVM --> PVM Stand for Python Virtual Machine(Python Interpretor.)
```

Three Important Functions

In [ ]:

```
type() --> return which kind of data we are going to use in our program.

id() --> Return the memory location of a variable.

print() --> Display the output.
```

Example of type,id and print Function

In [41]:

```
x = 10
print("Data Type is",type(x))
print("Memory Location is",id(x))
```

```
Data Type is <class 'int'>
Memory Location is 2221268232784
```

In [42]:

```
x = True
print("Data Type is",type(x))
print("Memory Location is",id(x))
```

```
Data Type is <class 'bool'>
Memory Location is 140725274466408
```

In [43]:

```
x = 10.5
print("Data Type is",type(x))
print("Memory Location is",id(x))
```

```
Data Type is <class 'float'>
Memory Location is 2221387523056
```

Types of Datatype in Python

In [ ]:

```
Types of Datatypes in Python?

1.Numeric Datatype --> int,float,complex
2.Sequence Datatype --> List,String and Tuple
3.Boolean Datatype --> True and False
4.Dictionary
5.Set
```

Numeric Datatype

Integer Datatype

In [ ]:

```
Integer Datatype --> Represents Integeral Value

Intergral Values:

    Positive Intergral Value --> 0,1,2,3,..... Infinity
    Negative Intergral Value --> -1,-2,-3 ..... -infinity.

Integer Datatype --> if we want to represent any number without decimal then such datatypes are known as
                     Integer Datatype.
```

Example of Integer Datatypes

In [49]:

```
x = -99
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable Data is",x)
```

```
Data Type is <class 'int'>
Memory Location is 2221391179056
Variable Data is -99
```

In [50]:

```
x = 0
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable Data is",x)
```

```
Data Type is <class 'int'>
Memory Location is 2221268232464
Variable Data is 0
```

In [51]:

```
x = 9999
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable Data is",x)
```

```
Data Type is <class 'int'>
Memory Location is 2221391179184
Variable Data is 9999
```

Important Point Related to Integer Datatype

In [ ]:

```
Note 1: In Python we don't have any short and long datatypes.
Note 2: If we want represent a  large Value(Long Datatype) then that value will also be considered as
        int datatype.
Note 3: Till Python 2.0 We are having long datatype in python for handling large values but From Python
        3.0 We don;t have any long dataype whatever long values are there that will also be considered
        as Int Datatype

Note 4 : In Int Datatype we don't have any limit you can numerical data in form of int.
```

Example of Integer Datatype

In [48]:

```
x = 9999999999999999999999999999999999999999999999999999999999999999999999999999999999999999999999999999
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable Data is",x)
```

```
Data Type is <class 'int'>
Memory Location is 2221381374400
Variable Data is 9999999999999999999999999999999999999999999999999999999999999999999999999999999999999999999999999999
```

Float Datatype

In [ ]:

```
Float Datatype --> If we want to represent a floating point number(Decimal Number) then we are going to
                   use float Numbers.

Float Positive Number --> 0.1,0.2....infinity
Float Negative Number --> Any float number with negative symbol
```

Example of Float Datatype

In [52]:

```
x = 10.87
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable Data is",x)
```

```
Data Type is <class 'float'>
Memory Location is 2221350915056
Variable Data is 10.87
```

In [53]:

```
x = 0.0
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable Data is",x)
```

```
Data Type is <class 'float'>
Memory Location is 2221391015440
Variable Data is 0.0
```

In [54]:

```
x = 10.00
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable Data is",x)
```

```
Data Type is <class 'float'>
Memory Location is 2221350915056
Variable Data is 10.0
```

In [55]:

```
x = 988888888888888888888888888888888888888888888888888888888888888888888888888888888888.0
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable Data is",x)
```

```
Data Type is <class 'float'>
Memory Location is 2221391178352
Variable Data is 9.888888888888889e+83
```

Complex Datatype

In [ ]:

```
A number which is in the form of
      a+bj

a--> real part(Real Number)
b--> Imaginary Part(Imaginary Number)

j ==> squareroot(-1)
j^2 ==> -1

Note: There are Two Attributes which will help us to access the imaginary and real part of a complex
      number.

    --> imag  is used to access imaginary part in form of floating Number

    --> real is used to access real part in form floating Number
```

Example of Complex Number

In [57]:

```
x = 10+20j  #a+bj
#real ==> 10
#imaginary ==> 20
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Real part is ",x.real)
print("Imaginary Part is ",x.imag)
print("Variable Data is",x)
```

```
Data Type is <class 'complex'>
Memory Location is 2221391179376
Real part is  10.0
Imaginary Part is  20.0
Variable Data is (10+20j)
```

In [58]:

```
x = 10-20j  #a+bj
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Real part is ",x.real)
print("Imaginary Part is ",x.imag)
print("Variable Data is",x)
```

```
Data Type is <class 'complex'>
Memory Location is 2221391178512
Real part is  10.0
Imaginary Part is  -20.0
Variable Data is (10-20j)
```

In [59]:

```
x = 10.23-20.34j  #a+bj
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Real part is ",x.real)
print("Imaginary Part is ",x.imag)
print("Variable Data is",x)
```

```
Data Type is <class 'complex'>
Memory Location is 2221391179056
Real part is  10.23
Imaginary Part is  -20.34
Variable Data is (10.23-20.34j)
```

Where we use Complex Numbers?

In [ ]:

```
Note --> Complex number are very useful for scientific Applications(Scientific Computation).
     --> ISRO and NASA are preferying python as a core programming language.
```

Boolean Datatype

In [ ]:

```
Boolean means True or False.

Internally True is equal to 1
           False is equal to 0
```

Example of Boolean Datatype

In [60]:

```
x = False
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable Data is",x)
```

```
Data Type is <class 'bool'>
Memory Location is 140725274466440
Variable Data is False
```

In [61]:

```
x = False+True+True #0+1+1
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable value is",x)
print("Variable Data is",x)
```

```
Data Type is <class 'int'>
Memory Location is 2221268232528
Variable value is 2
Variable Data is 2
```

In [62]:

```
x = False+True+True+False-False+True+True+False
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable value is",x)
print("Variable Data is",x)
```

```
Data Type is <class 'int'>
Memory Location is 2221268232592
Variable value is 4
Variable Data is 4
```

String Datatype

In [ ]:

```
Strings --> Any sequence of character either in single quotes,double quotes or triple quotes.

        --> If we are going to represent anything with single quotes,double quotes or triple quotes
            then that will be considered as a String.

        --> In Python we don;t have char datatype(If we are representing a single character inside
                                                  single quotation)

        --> If we are representing a single character inside single quotation then that will also
            considered as a string in python.
```

Example of Strings

In [28]:

```
x = "Pratyush Srivastava"
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable value is",x)
```

```
Data Type is <class 'str'>
Memory Location is 2221379449440
Variable value is Pratyush Srivastava
```

In [29]:

```
x = 'Pratyush Srivastava'
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable value is",x)
```

```
Data Type is <class 'str'>
Memory Location is 2221385040560
Variable value is Pratyush Srivastava
```

In [30]:

```
x = '''Pratyush Srivastava'''
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable value is",x)
```

```
Data Type is <class 'str'>
Memory Location is 2221351044320
Variable value is Pratyush Srivastava
```

In [31]:

```
x = 'A'
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable value is",x)
```

```
Data Type is <class 'str'>
Memory Location is 2221272226608
Variable value is A
```

In [32]:

```
x = "10"
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable value is",x)
```

```
Data Type is <class 'str'>
Memory Location is 2221277653424
Variable value is 10
```

In [33]:

```
x = "10.15"
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable value is",x)
```

```
Data Type is <class 'str'>
Memory Location is 2221383355504
Variable value is 10.15
```

In [34]:

```
x = '10+20j'
print("Data Type is",type(x))
print("Memory Location is",id(x))
print("Variable value is",x)
```

```
Data Type is <class 'str'>
Memory Location is 2221383397104
Variable value is 10+20j
```

In [63]:

```
a="0"
print("Data Type is",type(a))
print("Memory Location is",id(a))
print("Variable value is",a)
```

```
Data Type is <class 'str'>
Memory Location is 2221274079792
Variable value is 0
```