Generally any organization contains 2 parts.

1. Data part
2. Operations part.

For example , Bank contains two parts .

- In bank , Bank have Customers and Employees. Bank maintaining each person related data and providing the some services to each person.

For example,

1. Customer

a) data ------>> c.name, c.address,cbal,…….

b) operations------>> deposit(),withdraw(),bal.eng()……

2.Employee

a) data ----->> e.name, e.address,e.bal,…….

b) operations ------>> da(),hra(),pf(),tax(),tsal().

Data :

- Data of the any organisation we can represent by using any program language.
- To represent the data of organisations all programming languages are providing **datatypes** and **variables**.
- Every programming language supports datatypes and variables but the datatypes and variables of one language are not going to be same with another programming language.

Operations:

- Operations of the organisations we can represent by using any programming language.
- To represent the operations of the organisations every programming language is providing **functions** or **methods** or both.
- Every programming language supports functions or methods but the functions or methods of one language are not going to be same with another programming language.

Data types:

- Data types are nothing but some of the keywords of the programming languages, which are used to specify what type of data has to be store into the variables.
- Without data types we cannot store the data into the variables.
- Python supports the dynamic data types i.e, at the time of execution of program data type of the variable will be decided based on the data which is assigned to that variable.
- for example, a = 10 ---->> here a is maintaining integer value 10.
- At the time of writing the program, programmers should not specify data type to the variables explicitly otherwise we will get error.
- for example, int a = 10 ---->> it returns error because of datatype int.

Python data types are categorized into two types.

1) Fundamental data types.
2) Collections data types.

1. Fundamental data types.

- The variables which are represented with fundamental data types are stores the address of the object in which we can represent only one element.
- Python supports the following fundamental data types

1. Numbers

a) int -------->> a = 10
b) float -------->> b = 10.50
c) complex -------->> c = 2 + 3j
d) bool -------->> d = True / False

2. Strings

a. str ------>> name = “Virat”

2) Collection data Types.

- The objects which are maintaining the group of elements at a time are called collection data types.
- Python is providing somany collection datatypes.

For example,

a) list type

lst = [10,20,30]

b) tuple type
tup = (10,20,30)

c) set type
set1 = {10,20,30}

d) dict type
dict1 = {1:10, 2:20, “name”:”Virat”, “age”:30}

For example,

l = [1,2,3,4]

Immutable and Mutable Objects:

- Everything in Python is an object.
- All objects in Python can be either mutable or immutable.
- So here , above all data types are again devided into 2 types.

1. ImMutable

a) Numbers (int , float , bool , complex)
b) Strings (str)
c) Tuple (tuple)

2. Mutable

a) List (list)
b) Set (set)
c) Dictionary (dict)

Q. What is difference between Mutable and Immutable objects?

- **A mutable object can be changed after it is created, and an immutable object can not be changed after it is created**.

Built-In Functions:

- Python supports the so many builds in functions.

1. type():

- type() function is used to know the data types of variables.

For example,

```
>>> a = 10
>>> type(a)
<class 'int'>
>>>
```

2. id() :

- id() function is used to know the address of the object which is pointed by the variables.
- The built-in function id() returns the identity of an object as an integer.

for example,

```
>>> a = 10
>>> id(a)
1648190784
>>>
```

3. print():

- print() function is used to display the data on console.

For example,

```
>>> a = 10
>>> print(a)
10
>>> print('Hello')
Hello
```