# Python Basics

## Datatypes
- **Numbers:** a=2(Integer), b=2.0(Float), c=1+2j(Complex)
- **Sets:** a={2,3,4,5}
- **List:** a=[1,2,3,’Word’]
- **Dictionary:** x={‘a’: [1,2],’b’: [4,6]}

## Operators
**Numeric Operator:** (Say, a holds 5, b holds 10)
- a + b = 15
- a - b = -5
- a * b = 50
- 7.0/2.0 = 3.0, -11/3 = -4
**Comparison Operator:**
- (a == b): not true
- (a > b): not true
- (a >= b): true
- (a < b) is not true
**Boolean Operator:**
- a and b
- a or b
- not a

## Operations
- **List Operations**
  - list[:]: Defines an empty list
  - list[i]: Stores a at the ith position
  - list[i]: Retrieves the character at the ith position
  - list[i:j]: Retrieves characters in the range i to j
  - list.append(val): Adds item at the end
  - list.pop(i): Removes and returns item at index i
- **String Operations**
  - String[i]: Retrieves the character at the ith position
  - String[i:j]: Retrieves characters in the range i to j
- **Dictionary Operations**
  - dict(): Defines an empty dictionary
  - dict[a]: stores “a” to the key “i”
  - dict[i]: Retrieves the item with the key “i”
  - dict.keys(): Gives all the keys items
  - dict.values(): Gives all the values

## Functions
- **lambda Function**
  - lambda a,b: a+b
- **def new_function():**
  - print(“Hello World”)
  - new_function()

## Comments
- **Single Line Comment**
  ```
  # Example of a single line comment
  ```
- **Multi-line comments**
  ```
  # Start of the multi-line comment
  # Example of multi-line comments
  # End of the multi-line comment
  ```

## Generic Operations
- **range(5):** 0,1,2,3,4
- **S=**input(“Enter:”)
- **len(a):** Gives item count in a
- **min(a):** Gives minimum value in a
- **max(a):** Gives minimum value in a
- **sum(a):** Adds up items of an iterable and returns sum
- **sorted(a):** Sorted list copy of a
- **importing modules:** import random

---

### Class/object
```python
Class:
  class Pen:
    obj=Pen()
  pass
```

### File Operations
```python
f= open("File Name","opening mode")
(Opening modes: r: read, w: write, a: append, r+: both read and write)
```

### Try & Except Block
```python
try:
  [Statement body block]
  raise Exception()
except Exception as e:
  [Error processing block]
```

---

**FURTHERMORE:**
Python for Data Science Certification Training Course

---

**OOPS**
- **Inheritance:**
  A process of using details from a new class without modifying existing class.
- **Polymorphism:**
  A concept of using common operation in different ways for different data input.
- **Encapsulation:**
  Hiding the private details of a class from other objects.