Python Programing

1. What is Python

   Python is general purpose programing language which is used in various fields such as like wed development, mobile development, desktop application, data science, machine learning...

   Python is a general-purpose high level, dynamically typed, interpreted programing language which support for the procedural, functional and Object-oriented programing paradigms.

   a. High level language: Human readable code
   b. Dynamically typed: Variable declaration not required
   c. Programing paradigms: Python will support for the procedural programing (It will follow the step-by-step process) functional programing (Functions will play vital role) Object oriented programing (Objects and classes use while programing)
   d. Interpreted language: Python, the source code is executed line by line by an interpreter at runtime. The Python code is not directly translated into machine code (like compiled languages) before execution. Instead, it is first converted into bytecode by the Python interpreter, which is then executed by the Python Virtual Machine (PVM).

2. How to Install Python

   Python is a cross plat-form application we can install in any operating system. Basically, we can install python 2 ways

   a. From official web site we can download exe file and install the application website: https://www.python.org/
   b. From micro store we install python directly.

3. How to execute python program

   After installation of python, Python comes with the Default editor Python IDLE. which is works on the REPL (Read, Evaluation, Print, Loop) it won't useful for much complex situations. In this situation we need to depend on IDE such as Like PyCharm, VsCode... In IDE or IDLE we can run the program and get output from the console. We can command line execution as well by calling the particular script.

4. Basics of the Programing

a. Variable: Variable is a container which is used to store the value with the reference.

Name = "Apple", - Name is a Variable Reference and Apple is a Variable.

```
# name = "Apple"
  Print(name)
```

When ever we want to use that variable, we will call with the variable reference.

b. Data types: Data types will define the type of an object or variable. To make any operations we will know which data type that particular object then only we can perform some operations. Here in this kind of scenarios data types are really important.

- Integer: Allow only whole numbers. Ex: 1, 100, -5, 0
- Float: Allow only Decimal values. Ex: 12.5, 10.0, 11.7
- String: Allow only sequence of characters. Ex: "Apple", "Box", "C"
- Boolean: Allow only True or False. Ex: True, False

In some other programing languages character is different data type in python it will be string.

```
# num = 6

  Print(type(num)) This will give us the variable data type
```

c. Conditional Statements: Conditional statements will define the which block of code need to be executed. We have three conditions over there if, elif, else.

How it will work?

First, we will define the condition with an if statement once if statement not satisfied it will check for the elif block if elif block in the program it will go and check for that condition based on that it will decision other wise it will go to the else block.

```
#  a = 7
   If a < 5:
         Print ("a is less than five")
   Elif a < 10:
         Print ("a is greater than 5 and less than 10")
   Else:
         Print ("a is greater than 10")
```

d. Looping statements: Loops are help us to run a task multiple times without code repetition. In python we have 2 types of loops are there while loop and for loop.
While loop: While loop work on the condition is true or not until condition become false loop will execute. Some times if we not provided proper condition it leads to infinite loop

```
#  a = 5
   While  a < 5:
          Print(a)
          a +=1
```

For loop: For loop works on based on the length of iterable or condition. For we won't face any infinite loop issue.

```
#  list_of_fruits = ["apple", "box", "cat", "dog",
   For i in list_of_fruits:
          Print(i)
```

e. Mathematical operations: If we want perform some actions like addition, subtraction we will use mathematical operations.

```
# a, b = 5, 7
# Addition   a + b
# Subtraction  a - b
# Multiplication a * b
# Division a / b, this will give us floating value
# Floor division a // b, this will give us the integer value
# Modulo a % b, this will provide the remainder value
# Exponent a ** 2
# we can perform multiple operations on the same line of code
# we can do round the decimal values round(value, decimal precision)
```

f. Console actions: Console is used to display the output of the program with help of print function and if we want to provide some thing to program use input function.

```
# name = input ("Enter your name: ") entered value will store in the name variable
# print (a, b, c, end = " ", sep = ",")
```

Print function will take different arguments end argument used to print output in the same line and sep argument used for the separate the variables with defined separator and file argument used to print something inside the file and flush function used print something forcefully.

g. Strings: As we discussed earlier strings are data types and it will support for the various operations.
   # Name = "Bangalore"
   # Accessing string character = Name [1]
   # String concatenation = Adding string to some other string
   # String will support for the various methods.
h. Comparison operators: Comparison useful when we are comparing two values
   # Equal: ==
   # Less than: <
   # Greater than: >
   # Less than or equal: <=
   # Greater than or equal: >=
   # Not Equal: !=
i. String formatting: Any integer value or decimal value need to be print inside the string we will get the error in this condition we need to do string formatting
   # print ("My phone no:", 1234567890)

5. Intermediate python:
   a. Functions: Function is the block of code we can reuse as many times we required.
      # def greet(name):
            Print (f"Hello {name}")
   b. Built in functions: There are some built functions are available in python we can use them as per requirement.
      # len (iterable) This will give us the length of the iterable.
      # map (function, iterable1, iterable2, ...) Transform
      # Filter (function, iterable) Filter
      # Reduce () It will take 2 elements as the input and provide final output.
      # Lamada
      # Zip
      # Enumerate
   c. Decorators: Decorators are the functions which can change the behavior of existing function without change the function code while run time.
   d. Generators: Generators are the functions which are execute the iterable values without storing it. Which are more efficient.