Getting started with Python

Index

History of Python

=>Python Programming Language Conceived in the Year 1980.

=>Python Programming Language began its Implementation (Bring into action) in the year 1989.

=>Python Programming Language Officially Released in the year 1991 Feb 20

=>Python Programming Language Developed by "GUIDO VAN ROSSUM" (Father of Python)

=>Python Programming Language Developed at Centrum Winskunde Informatica(CWI) Institute in Nether Lands.

=>ABC Programming Language is the predecessor of Python Programming Language.

=>Python Programming Language maintained by a Non-Commercial Organization Python Software Foundation(PSF)

=>The official website of Python Software Foundation(PSF) is www.python.org

Versions of Python

=>Python Programming Language contains 3 types of Versions. They are

1) Python 1.x where 1 is called Major Version and x is called Minor Version
   here x represents 0 1 2 3 4 5 6 7...etc. (Outdated)

2) Python 2.x where 2 is called Major Version and x is called Minor Version
   here x represents 0 1 2 3 4 5 6 7 (Outdated 1-1-2020)
3) Python 3.x where 3 is called Major Version and x is called Minor Version
   here x represents 0 1 2 3 4 5 6 7 8 9 10 11 12

=>Python Programming Language does not provide Backward Compatibility, (Python 3.x does not support the features and syntaxes of Python 2.x).

==

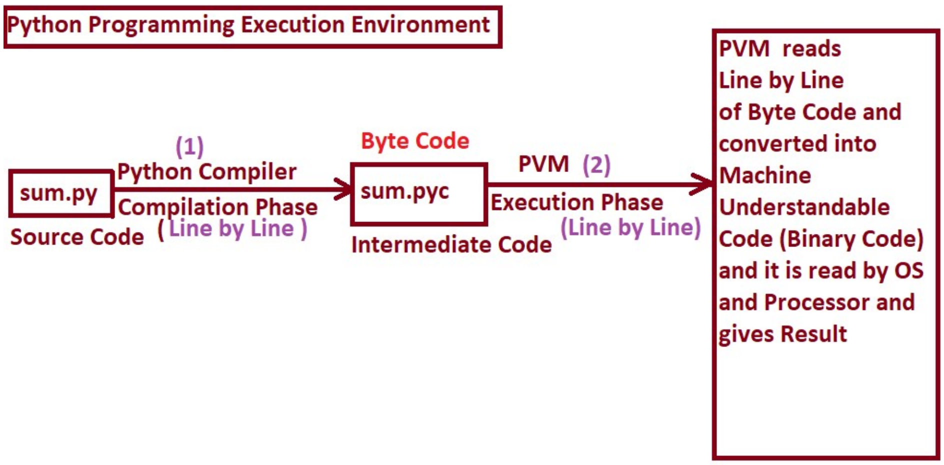
Features of Python Programming Language

==

=>Features of language are nothing but services or Facilities provided by Language Developers in Languages and those facilities are used by Programmers for developing Real Time Applications.

=>Python Programming Provides 11 Features. They are

1. Simple
2. Freeware and Open Source
3. Platform Independent
4. Dynamically Typed
5. Interpreted
6. High Level
7. Both Procedure Oriented and Object Oriented Programming Language
8. Robust(Strong)
9. Extensible
10. Embedded
11. Extensive Support for Third Party APIs such as Numpy, Pandas, matplotlib, scipy, scikit, keras, NLTK...etc.


Python Programming Execution Environment
sum.py
Source Code
(1)
Python Compiler
Compilation Phase
(Line by Line)
Byte Code
sum.pyc
Intermediate Code
PVM (2)
Execution Phase
(Line by Line)
PVM reads Line by Line of Byte Code and converted into Machine Understandable Code (Binary Code) and it is read by OS and Processor and gives Result

1. Simple

=>Python is one of the simple programming language because of 3 important Technical Features. They are

1. Python Programming Provides "Rich Set of APIs". So that Python programmer can re-use the pre-defined code without writing our code Definition of API (Application Programing Interface):

=>An API is a collection of MODULES.
=>A Module is a collection of Attributes, Functions and Class Names
Examples: calendar, math, cmath, random, gc ....etc.

2. Python Programming Provides In-built facility called "Garbage Collector" and whose role is to collect Un-Used Memory space and Improves the Performance of Python Based Applications.

=>Definition of Garbage Collector:

=>A Garbage Collector is one of the In-built program in python software and which is running in the background of our regular Python Program and whose role is collecting Un-Used Memory space and Improves the Performance of Python Based Applications.

=>Hence Garbage Collector takes care about Automatics Memory Management.

3. Python Programming Provides User-Friendly Syntaxes. So that we can develop error-free programs and Limited span of time.

2. Freeware and Open Source

Freeware:

=>If any software is said to be Freeware if It can be downloaded freely from official websites.
=>Python Software

Open Source:

=>The standard Name of Python is called “CPYTHON"
=>Some of company venders came forward and Customized "CPYTHON" and used the customized versions as In-house tools in there companies. Such Type of Customized versions of python are called "Distributions of Python"

=>Some of the Distributions of Python are

1) Jpython or Jython---->Used for Running Java Based Applications
2) Iron Python or Ipython---->Used to Run C#.net applications
3) Micro Python-------------->Used to develop micro controllers.
4) Ruby Python---------->Used to Ruby Based Applications
5) Anaconda Python---->Used to Run Hadoop Applications / Big Data Applications...............etc.

3. Platform Independent

=>In this context, we are having two types of Programming Languages. They are

1. Platform Dependent Languages
2. Platform Independent Languages

1. Platform Dependent Languages

=>If Language is said to be Platform Dependent if and only if whose applications runs on same Type of OS.
=>The property of Platform Dependent languages is that "Whose memory space differs from One OS to Another OS ".
=>Examples: C, CPP, .NET ......etc.

2. Platform Independent Languages

=>If Language is said to be Platform Independent if and only if whose applications runs on Any Type of OS.

=>The property of Platform Independent languages is that "Whose memory space remains on Types of Operating Systems

=>In Java Programming, all data types take same memory space and hence java is Platform Independent language. But java objects take multiple values with size restricted.

=>In PYTHON Programming, all values are stored in the form of OBJECTS.

=>All OBJECTS of PYTHON are Large Store Entities without Size Restrictions and they can execute on all types of Oses. Hence PYTHON is one of the Platform Independent Language.

Examples: Java, PYTHON.

4. Dynamically Typed

=>In the context programming languages, we have two types of languages. They are

1. Static Typed Programming Languages.
2. Dynamically Typed Programming Languages.

1. Static Typed Programming Languages.

=>In These Programming Languages, Programmer must define a variables with Data Type otherwise we get Compile Time Error

```
Examples: int a, b, c; -------Variable Declaration----Mandatory
                   a=10
                   b=20
                   c=a+b
```

Examples: C, CPP, JAVA, .net.........

2. Dynamically Typed Programming Languages.

=>In These Programming Languages, Programmer need not to specify the data type of the variables. So that data type of Variable decided by Python Execution Environment based value assigned by Programming

Examples:

```
>>> a=10
>>> b=1.2
>>> c=a+b
>>> print (a, type(a)) --------------------- 10 <class 'int'>
>>> print (b, type(b)) -------------------- 1.2 <class 'float'>
>>> print (c, type(c)) --------------------  11.2 <class 'float'>
```

Examples: Python

=>Hence All values in Python Programming are stored in the form of Objects.

5. Interpreted Programming

=>When we develop any python program, we must give some file name with an extension .py (File Name.py).

=>When we execute python program, two process taken place internally

a) Compilation Process
b) Execution Process.

=>In COMPILATION PROCESS, the python Source Code submitted to Python Compiler and It reads the source Code, Check for errors by verifying syntaxes and if no errors found then Python Compiler Converts into Intermediate Code called BYTE CODE with an extension.pyc (FileName.pyc). If errors found in source code then error displayed on the console.

=>In EXECUTION PROCESS, The PVM reads the Python Intermediate Code (Byte Code) and Line by Line and Converted into Machine Understable Code (Executable or binary Code) and It is read by OS and Processer and finally Gives Result.

=>Hence in Python Program execution, Compilation Process and Execution Process is taking place Line by Line conversion and It is one of the Interpretation Based Programming Language.

=>Definition of PVM (Python Virtual Machine)

=>PVM is one program in Python Software and whose role is to read LINE by LINE of Byte Code and Converted into Machine Under stable Code (Executable or binary Code).

6. High Level Programming

=>In general, we have two types of Programming languages. They are

a) Low Level Programming Languages.
b) High Level Programming Languages.

a) Low Level Programming Languages:

=>In These Programming Languages, we represent the data in lower level data like Binary, Octal and Hexa decimal and This type data is not by default understanble by Programmers and end users.

Examples: - a=0b1111110000111101010---binary data
b=0o23-----octal
c=0xface----Hexa Decimal

b) High Level Programming Languages.

In These Programming Languages, even we represent the data in lower level data like Binary, Octal and Hexa decimal, the High Level Programming Languages automatically converts into Decimal Number System data, which is understanble by Programmers and end-users and python is one High Level Programming Language.

Example: Python

8. Robust (Strong)

=>Python is one of the Robust Programming Language bcoz of there exist a programming Facility called " Exception Handling ".

=>Exception: Runtime Error is called Exception
Exceptions by default gives Tech Error Messages

=>Exception Handling: - The process of Converting Tech Error Messages into User Friendly Error Messages is called Exception Handling.

==

Extensible and Embedded

==

=>Extensible feature of Python makes us to understand "Python Code can be written inside other languages code."

=>In Other Words Python code can be written inside C, CPP and Java programs.

=>Embedded feature of Python makes us to understand " Python Code can include other language Code"

=>In Other words Python code can include the Code C, CPP and Java Languages.

===============================X================================

==

Extensive Support for Third Party APIs

==

=>As Python Libraries / API can do many tasks and Operations and unable perform complex operations and to solve such complex operations more easily and Quickly we use Third Party APIs such as

1) numpy----Numerical calculations
2) pandas---Analysis tool

----

3) matplotlib-----Data Visualization
4) scipy
5) scikit

================================X===============================