Introduction to Python

Chapter - 1 & 2

16.01.2024

Contents

Introduction

- Python
  - Write program without creating any class
  - Combines features of C & Java
  - Elegant style of C programming + OOP like Java (optional)
  - Developed by Guido Van Rossum (Netherlands),
    - 20.02.1991
    - Center of Mathematics and Computer Science – Dutch Govt.
    - https://www.python.org/

Features of Python

- Simple
  - Easy to read python programs
    - More clarity & less stress on understanding syntax and program.
      - Hence, program development becomes easy.
- Easy to learn
  - Python uses very few keywords.
    - Programs are very simple in structure.
- Open source
  - Don’t pay for Python.
- High level language
  - ?
- Dynamically typed
  - No need to declare variables
    - An assignment statement binds a name to an object of any type.
    - If a name is assigned to an object of one type, it may later be assigned to an object of a different type.
  - Unlike C & Java
- Platform independent
  - Python program -> Python compiler -> byte code.
  - Python Byte code: represents a fixed set of instructions that run in all operating systems and hardware on PVM.

Features of Python

- Portable
  - ?
- Procedure & object oriented
- Interpreted
  - Byte code -> PVM
- Extensible
  - C/C++ programs/pieces: can be integrated into Python program
  - Jython
  - Iron Python
- Embeddable
  - Insert Python programs into C/C++ programs
    - PHP, Java, Delphi, .Net, ..

Features of Python

- Scripting language
  - Compiler required?
- Database connectivity
- scalable
- Several packages already available
  - argparse: command-line parsing library
  - cherrypy: OO HTTP framework library
  - cryptography: cryptographic functions library
  - Fiona: read/write big data files
  - jellyfish: string processing
  - mysql-connector-python: mysql database connectivity

Features of Python

- Several packages already available
  - …
  - numpy: 1-D & multi-D array processing
  - pandas: data analytics, time series, statistics
  - matplotlib: graphs, electronic circuits
  - pillow: imaging library
  - scipy: scientific and engineering calculations
  - Sphinx: python documentation generator
  - sympy: computer algebra system
  - w3lib: web related functions

Availing Python (Linux)

- $ sudo apt update
- $ sudo apt upgrade
- $ python3 –V or
- $ python3 –version

Python program execution

- <filename>.py
- Python compiler
  - Converts the source code into bytecode
  - Byte code
    - Represents a fixed set of instructions that represent all operations

    (arithmetic operations, comparison operations, memory related operations, etc.)
    - Size: 1 byte (so, known as byte code)
    - <filename>.pyc : python compiled file
    - Can’t run directly on computer (computer can only run binary code)
      - Requires PVM Python Virtual Machine
        - PVM understands byte code, converts into binary code i.e. machine code (as per underlying platform)

Python program execution

- Interpreter
  - Translates program source code line by line; slow.
  - Interpreter that is found inside PVM runs the python program slowly.
  - Now JIT (just in time) compilers are available.
    - Not available in all Python environments.
    - The std. python s/w i.e. Cpython doesn't contain JIT.
    - Available in PyPy.
- C:\<path>\python <filename>.py
- C:\<path>\python –m py_compile <filename>.py
  - Bytecode available in: __py_cache__
  - C:\<path>__py_cache__\<filename>.cpython-38.pyc
- How to view the byte code?
  - python –m dis <filename>.py

Flavors of Python

- Cpython
- Jython
- IronPython
- PyPy
- RubyPython
- StacklessPython
- Pythonxy