Introduction to Python Programming

By Sujay Goswami

Invention of Python Language

- Developed by Guido van Rossum in 1991 at the National Research Institute for Mathematics and Computer Science.
- Python creator Guido van Rossum named it after the BBC comedy series Monty Python's Flying Circus.

Why to study Python?

- Development productivity
- Software Quality
- Program portability
- Support libraries
- Component integration
- Enjoyment

What can you do with python?

- Graphical User Interface
- Internet Scripting
- Database Programming
- Numeric and Scientific programming
- Components integration

Python’s Technical Strength

- It’s Free
- It’s Object Oriented
- It’s easy to learn
- It’s mixable

Features of Python's

- Easy to Learn and Use
- Interpreted Language
- Cross-Platform Language
- Free and Open Source
- Object-Oriented Language
- Large Standard Library
- Intergrated

Application of Python's

- Web Application
- Desktop GUI Application
- Software Development
- 3D-CAD Application
- Console Based Application
- Audio/Video Based Application
- Artificial Intelligence and Machine Learning
- Research

Definition of Python's

- Python is a General-purpase dynamic , High level and Interpreted Programming Language.

- It Supports Object Oriented Programming Features to develope various type of Application.

Python’s Data Types

- Python takes data in the form of objects
- Objects, either built-in object provided by Python or created using Python
- Objects are most fundamental notion of Python programming

Python's Core Data Types

Object Type	Examples
Numbers	1234, 3.145,3+6j
Strings	'spam', "guido"
Lists	[1,2,3],[1,[2,3]]
Dictionaries	{a:1,b:3,4:c}
Tuples	(1,2,"ram")
Files	Flptr=open("test.txt", 'r')
Sets	Set('abc'),{'a','b','c'}

Python's Statement

Statements are the things written to tell Python what your program should do

Statement	Role	Example
Assignment	Creating references	a,b='good','bad'
Calls and other expression	Running functions	log.write("spam,ham")
Print calls	Printing objects	print("hi')
If/elif/else	Selecting actions	if a>0: print('hi')
For/else	Sequence iteration	for i in range(5):print(i)
While/else	General loop	while x>y: print('hello')
def	Function and method	def fun(a,b): pass
class	Building objects	class student: pass
Import	Module access	import numpy
from	Attribute access	from module import fun