Python Programming

Duration: 4 Days

Introduction	Python is an interpreted general-purpose high-level programming language. Python aims to combine "remarkable power with very clear syntax", and its standard library is large and comprehensive. Its use of indentation for block delimiters is unique among popular programming languages.
Learning Objectives	At the end of the session, participants will be able to provide useful help in Python projects for programming.
Prerequisites	Any Programming Language
Training Methodology	Heavily hands on Material and example python scripts will be given to the participants at the end of the training.

Day wise Course overview	
Day 1	Introduction and Python (loops, data structures, functions)
Day 2	File handling, OOPS, Exception Handling
Day 3	Advanced oops, Machine Learning Libraries numpy, pandas, matplotlib
Day 4	Functional programming, AWS automation using boto, Creating python modules and packages.

Introduction to Python

* Working with the python interpreter
* Numbers and expressions
* Variables and statements
* Conditional statements and loop
* Handling user input
* Python syntax, style and coding conventions

Lists and Tuples

* Common sequence operations
* Manipulation of Lists
* Manipulation of Tuples

Working with Strings

* An overview of strings in python
* String operators
* Built-in string manipulation functions
* Built-in string methods
* Special string features in python

Working with dictionaries

* Introduction to dictionaries
* Creating, assigning, updating dictionaries
* Dictionary operators, functions and built-in methods

Functions and Modules

* Creating user-defined functions
* Passing functions
* Formal arguments
* Variable-length arguments
* Variable scope
* Creating modules
* Understanding namespaces
* Importing modules and module attributes
* Packages

File and Directory handling

* File I/O operations
* Built-in file and directory handling libraries
* fileinput
* stat
* filecmp and dircmp
* glob, zipfile and tarfile

Classes and Objects

* Introduction to OOP using python
* Classes and class attributes
* Instances and instance attributes
* Binding and method invocation
* Inheritance

Errors and exception handling

* Introduction to exceptions
* Detecting and handling exceptions
* Exceptions as Strings and Classes
* Raising exceptions
* Creating exceptions
* Standard exceptions

Advanced OOPS

* Understanding Duck typing
* Operator Emulation
* Understanding str,len,bool methods
* How to make an object iterable
* Creating a generator

Iterators and Generators

* Understanding “yield”
* Creating generators using comprehension syntax

Data Analytics - Libraries

* numpy
* pandas
* matplotlib

Functional Programming

*Understanding Lambdas
* Map
* Reduce
* Filter

AWS automation using boto 3

Amazon S3

- Creating the Connection
- Creating a Bucket
- Storing Data
- Accessing a Bucket
- Deleting a Bucket
- Iteration of Buckets and Keys

Amazon EC2

- Creating the Connection
- Launching New Instances
- Stopping & Terminating Instances
- Checking What Instances Are Running
- Checking Health Status Of Instances