



Python for Development

Document History

Ver. Rel. No.	Release Date	Prepared. By	Reviewed By	Approved By	Remarks/Revision Details
V 1.3	20/03/2024	Ramakant Debata	Priya Ranjan Acharya (MS/EDX5- XC)	Priya Ranjan Acharya (MS/EDX5-XC)	

Course Title:

1. Course Summary:

The course aims at explaining the practical usage of python for development. The course discusses the fundamental syntax and philosophy of python along with its core programming elements.

2. Pre-Requisite

Basic understanding of programming would be preferable.

3. Audience

Engineers who are looking forward to start programming.

4. Hardware & Network Requirements

- Any quad core CPU or above
- Windows or mac
- 8gb RAM or above
- Basic internet connection

5. Software Requirements

- OS: Windows 10/11 Pro or Enterprise
- IDE: Visual Studio Code (<https://code.visualstudio.com/>)
- Git
 - [Git for windows](#) (for Windows platform)
 - git package for Linux/MacOs
- Installed software/modules:
 - Python Python 3.11.3 (<https://www.python.org/ftp/python/3.11.3/python-3.11.3-amd64.exe>)
 - Pip
 - `curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py`
 - `python get-pip.py` # Install pip
 - `pip --version` # Verify if pip installed
 - Python3 Modules (On command line, where python is installed)
 - `py -m pip install --upgrade pip`
 - `pip install virtualenv numpy pandas scipy urllib3 multipledispatch requests`
 - Should be able to run below program (SetupTest.py) from participant login

```
# SetupTest.py

import platform import
sys
import virtualenv
import numpy
```

```
import scipy import
urllib3
import multipledispatch import
requests

print("Python version:", platform.python_version()) print("virtualenv
version:", virtualenv._version_) print("numpy version:", numpy._
version_) print("pandas version:", pandas._version_) print("scipy
version:", scipy._version_) print("urllib3 version:", urllib3._version_)
print("multipledispatch version:",
multipledispatch._version_)
print("requests version:", requests._version_)
```

v. Program should run without error and list out the versions of the modules correctly.

6. Learning Outcomes:

- Understand the basic language structure of python
- Python fundamental system
- Python collection
- Python object-oriented programming

7. Course Content (day wise):

Day 1

Introduction

- What's Python?
- Why do people use Python?
- Python Ecosystem
- Python Versioning
- Installing
- Switching
- Python IDE

Hello World Python

- Python Shell (REPL)
- Writing simple scripts
- Python 2 vs 3 Differences
- Executing Python Scripts

Python – Getting Started

- Python Statements
- Variables

- Operators and Expressions
- Datatype
- Object and Id
- Mutable and Immutable

Python Functions

- Function Basics
- Defining Functions
- Calling Functions
- Scopes

Python's Lists

- Common List Methods
- The range() Function
- List Operations
- String Indexing
- String Slicing
- String Iteration
- Multi-Dimensional Lists (Matrices)

Python String Types

- Generating Strings in Python
- Immutable
- Common String Methods
- Type Conversion in Python
- Formatting String Output
- Format Specifier
- Variable Substitution
- String Indexing
- String Slicing
- String Iteration

Python's Tuples

- Immutable
- Common Tuples Methods
- Tuples Operations
- Tuples Indexing
- Tuples Slicing
- Tuples Iteration
- Multi-Dimensional Tuples (Matrices)

Python Dictionaries

- Python Dictionaries

- Assigning Values to Dictionaries
- Dictionary Methods
- Dictionaries vs Lists & Tuples
- Dictionary Indexing
- Dictionary Iteration

Day 2

More on Python Functions

- Function Basics
- Defining Functions
- Argument Defaults
- Lambdas
- Local Variables
- Understanding `__builtin__`
- Preventing Variable Modifications
- Variable Args
- Keyword Argument Methods

Object Oriented Programming

- Introduction to OOP using python
- Classes and class attributes
- Instances and instance attributes
- Initialization and cleanup
- Binding and method invocation
- scopes
- Composition and Subclasses
- Built-in functions for classes, instances and other objects

Day 3

Exceptions

- About Exceptions
- Python's Default Exception Handler
- Using Try/Except/Else/Finally Exceptions
- Generating User Defined Exceptions
- More on Exceptions
- Exception Examples

Modules & Packages

- Module Basics
- Packages
- Using `__all__` and `_` Variables
- Using `__name__`
- Using third party modules

Standard Python modules

- Using the sys module
 - sys.argv, sys.path, sys.version
- An overview on `__builtin__` and `__future__` modules
- Using the os module
- Filesystem/directory functions

Day 4

Basic Input/Output with Files

- Opening Files
- Working with Files
- Controlling Output Location

XML and JSON

- Working with XML
- DOM and Sax
- Introducing ElementTree
- Parsing XML
- Navigating the document
- Creating a new XML document
- JSON
- Parsing JSON into Python
- Converting Python into JSON

Day 5

Regular expressions Overview

- Introduction to regexps
- Special symbols and characters for RE
- Metacharacters and Metasymbols
- Practical examples

Unit Testing and TDD

- What is Unit Testing
- Why is it important
- Unit Testing Framework
- Unit Testing anatomy
- TDD
- Red-Green-Refactor

Additional Topics

Overview of working with GUI using PyQt

- Creating UI
- Understanding UI Component
- Composing a UI
- Handling an Event

Packaging [Subject to time availability and participants' caliber]

- Packaging structure
- Tools available
- Versioning
- Creating distribution packages
- Publishing to PyPI

'requests' module [Subject to time availability and participants' caliber]

- Intro and installation of 'requests' module
- Basic requests
 - GET
 - POST
 - PUT
 - DELETE
- Handling responses
 - Response object
 - Status codes
 - Headers
 - Content
- Sending data
 - Query parameters
 - Form data
 - JSON data
- Headers
 - Custom headers
 - User-Agent
- Authentication
 - Basic Authentication
 - Bearer Tokens

8. Course Structure:

Activity	Indicative Number of Hours
Pre-Read Hours	n/a
Teaching Hours	32.5
Hands on Sessions Hours	02.5
Assignments & Tutorial Hours	n/a
Mock Project Hours	n/a

9. Course Structure:

Method of Assessment	Yes/No	Weightage
Pre-Assessment	Yes	100%
Mid-Assessment	No	
Post-Assessment	Yes	100%
Project Work	n/a	

10. Course Resources:

- 1) **Code Samples:** Sample code snippets and solutions for better understanding.
- 2) **Assignments:** Practical assignments to reinforce learning and build real-world skills.

11. Recommended Reading Links:

12. Course Owner (s):

Employee Name	Employee Mail ID	Business Unit