Course Name: Python Programming

Course Code:

Course Credit: 3-0-4

Course Objectives:

- To setup the environment to run the python programs
- To understand concepts about Data Types and Looping techniques
- To understand and implement the OOP concepts, Decorators, and Iterators
- To understand and build the Web Applications
- Debugging and Troubleshooting Python Programs

Course Content:

Theory

Module 1: Introduction to Python

Introduction: Introduction to Python, Setting up the environment, Installing Python, Running python program, Python's execution model, Guidelines on how to write good, The Python culture, A note on the IDEs

Built-in Data Types: Numbers, Immutable sequences, Mutable sequences, Set types, Mapping types – dictionaries, The collections module, Final considerations

Iterating and Making Decisions: Conditional programming, Looping, Putting this all together.

Module 2: Advanced Concepts

Functions, the Building Blocks of Code: Use of functions, Scopes and name resolution, Input parameters, Return values, Recursive functions, Anonymous functions, Function attributes, Built-in functions, Importing objects.

Saving Time and Memory: map, zip, and filter, Comprehensions, Generators, Some performance considerations, Name localization, and Generation behavior in built-ins.

Advanced Concepts – OOP, Decorators, and Iterators: Decorators, Class and object namespaces, Attribute shadowing, Initializing an instance, Accessing a base class, Multiple inheritance, Static and class methods, Private methods and name mangling, The property decorator, Operator overloading, Polymorphism

Module 3: Web Development

The Edges – GUIs and Scripts: Scripting-The imports, Parsing Arguments, The business logic, GUI application- The import, The layout logic, The business logic, The tkinter.tix module,The turtle module, wxPython, PyQt, and PyGTK, The principle of least astonishment, Threading considerations.

Web Development Done Right: Django design philosophy, The Django URL dispatcher, Setting up Django, Adding the Entry model, Customizing the admin panel, Creating the form, Writing the views, Tying up URLs and views, Writing the templates, Writing a Flask view, Building a JSON quote server in Falcon.

Module 4: Cloud Native Python

Building Microservices in Python: Modeling microservices, Building microservices, Testing the RESTful API.

Building a Web Application in Python: Getting started with applications, Working with Observables and AJAX, Binding data for the adduser template, Working on Observables with AJAX for the addtweet template, Data binding for the addtweet template, CORS - Cross-Origin Resource Sharing, Session management, Cookies.

Interacting Data Services: MongoDB terminology, Initializing the MongoDB database, Integrating microservices with MongoDB, Working with user resources, Working with the tweets resources.

Module 5: Exception Handling
Testing, Profiling, and Dealing with Exceptions: The anatomy of a test, Testing guidelines, Unit testing, Test-driven development, Exceptions, Profiling Python.
Debugging and Troubleshooting: Debugging with print, Debugging with a custom function, Inspecting the traceback, Using the Python debugger, Inspecting log files, Other techniques, Troubleshooting guidelines.

Course Outcomes:

On successful completion of the course, students will be able to,

- Install and Run Python Program
- Write functions and Loops in the python program
- Implementing OOPs concepts while writing Python Program
- Developing web applications using Django
- Build microservices in Python
- Test, Debug and Troubleshoot Python Programs

Text Books:

1. Learn Python Programming, 2nd Edition by Fabrizio Romano
2. Python Cookbook, 3rd Edition by David Beazley (Author), Brian K. Jones

Reference Books:

1. Python Programming: A Step-by-Step Guide For Absolute Beginners by Brian Jenkins and ATS Coding Academy
2. Python and AWS Cookbook: Managing Your Cloud with Python and Boto by Mitch Garnaat
3. Advanced Python Programming: Build high performance, concurrent, and multi-threaded apps with Python using proven design patterns by Dr. Gabriele Lanaro
4. Programming Google App Engine with Python: Build and Run Scalable Python Apps on Google's Infrastructure by Dan Sanderson