Python for Everybody

Course Name : Python for Everybody

Duration: 4 days

Course Objectives: How to use python for scripting, automation, development, data processing, data visualization, website, RESTFul web services and testing.

Certification Link(If Any): NA

Pre- Training Udemy Course Link:

1. **https://www.udemy.com/course/python-complete-bootcamp-2019-learn-by-applying-knowledge/**
2. **https://www.udemy.com/course/python-for-beginners-learn-all-the-basics-of-python/**

Post- Training Udemy Course Link:

1. **https://www.udemy.com/course/learn-python-programming-a-step-by-step-course-to-beginners/**
2. **https://www.udemy.com/course/python-rest-apis-with-flask-docker-mongodb-and-aws-devops/**
3. **https://www.udemy.com/course/python-and-flask-bootcamp-create-websites-using-flask/**
4. **https://www.udemy.com/course/elegant-automation-frameworks-with-python-and-pytest/**
5. **https://www.udemy.com/course/learn-selenium-automation-in-easy-python-language/**
6. **https://www.udemy.com/course/complete-course-on-data-visualization-matplotlib-and-python/**
7. **https://www.udemy.com/course/data-analysis-with-pandas/**

Detailed Lab Setup: Latest python to be downloaded and installed from the website Python.org

Prerequisites:

About python.

1. Software installation, introduction to IDLE and PyCharms.
2. Object types
    a. Numbers
    b. Strings
    c. Lists
    d. Dictionary
    e. Tuples
    f. Sets
3. if, if elif, if elif else.
4. while, while else.
5. for, for else.
6. break, continue, pass.

Day – 1:

What we learn:

- How to use python for processing files?
- Code reusability using functions

Topics to be covered:

1. Brief about prerequisites
2. File operations
    a. Open function
    b. Read write functions
    c. Modes

   d. Flush and close
   e. Print function for file operations
3. Functions
   a. About functions
   b. Positional arguments
   c. Default value arguments
   d. Variable arguments
   e. Keyword only arguments
   f. Variable keyword only arguments

Day – 2:

What we learn:

- How to use python for getting data from web
- How to use python for processing unstructured data using regex
- How to use python for communicating with SQL databases

Topics:

1. Modules and Packages.
   a. Creating modules
   b. Environments variables
   c. Different ways of importing the module
   d. From import
2. Pip and pypi
3. Read website data using Urllib library
4. Regular expressions
   a. Identifiers

b. Quantifies

c. Modifiers

5. Connecting to SQL databases

d. Sqlite/oracle/mysql

6. Database to csv/txt

Day – 3:

What we learn:

- How to use python for data analysis
- How to use python for data visualization
- How to use python for web development
- How to use python for RESTFul web services
- How to consume REST API

Topics:

Data Science Library - Pandas

1. About pandas
2. 2D data - DataFrame
3. Columns and rows
4. List/tuple to DataFrame
5. Database to DataFrame
6. DataFrame to csv
7. DataFrame to excel
8. DataFrame to json
9. Example DataFrame methods – describe, count, groupby, valuecounts

10. Accessing each columns using column name
11. Filtering the data
12. Slicing – access portion of the data
13. Creating excel sheet with graph and data

DataVisualization - matplotlib

14. Draw bar graph
15. View the graph
16. Saving the graph

Website Using Flask Framework

17. Creating the app
18. Url mapping
19. Error handling
20. Running the server
21. Connecting to databases
22. Creating html files

REST API / RESTFul Web Services Using Flask Framework

23. Creating the app
24. Creating an end point
25. Creating json responses

Accessing REST-API

26. Requests module

Day – 4:

What we learn:

- How to use python for web application testing

- How to use python for unit testing
- Object oriented programming
- Exceptions handling

Topics:

Selenium – Web Application Testing – 1 Test Case

1. Open the browser using webdriver
2. Access the url
3. Locate the elements
4. Sending keys
5. Verifying the output

Pytest – unit test – 1 Test Case

6. Creating the testcases
7. Run the tests

Custom Classes and Exceptions

1. Creating the classes
2. Class variables
3. Instance variables
4. Class methods
5. Instance methods
6. Static variables
7. Constructors
8. Single and multiple inheritance
9. Polymorphism
10. Method overriding

11. Operator overloading
12. Exception handling