“Python Syllabus”

An Introduction To Python:-

- What is Python and history of Python?
- Features of Python.
- Installation and Working with Python.
- Understanding Python variables.
- Python basic Operators.
- Python Identifiers, Keywords and Indentation.
- Understanding python blocks.
- Command line arguments.
- Getting User Input.
- Python Data Types.
- What are variables?
- Python Core objects and Functions.

Program Flow Control :-

- Conditional blocks using if, else and else if.
- For loops in python.
- For loop using ranges, string, list and dictionaries.
- Use of while loops in python.
- Loop manipulation using pass, continue, break and else.
- Programming using Python conditional and loops block.

Logic Building :-

- Condition Based Problems.
- Looping Related Problems.
- Numeric Logical Problems.

- String Logical Problems.
- Sorting Problems.
- Design Patterns.

List, Ranges, Dictionaries, Tuples and Sets in Python:-

- Introduction.
- Lists in Python
- Understanding Iterators
- Generators ,Comprehensions and Lambda Expressions
- Generators and Yield
- Next and Ranges
- Understanding and using Ranges
- Python Dictionaries
- Dictionary manipulation.
- Ordered Sets with tuples
- Sets
- Python Sets Examples

File Input and Output in Python:-

- Reading and writing text files.
- Reading config files in python.
- Writing log files in python.
- Understanding read functions, read (), readline (), readlines (), write () and writelines ().
- Writing Binary Files Manually.
- Using Pickle to Write Binary Files.
- Manipulating file pointer using seek.

Object Oriented Programming in Python:-

- OOPs Concepts.
- Concept of class, object and instances.
- Constructor, class attributes and destructors.
- Accessing attributes, Built-In Class Attributes.
- Inheritance
- Polymorphism (overlapping and overloading operators).
- Achieving Abstraction
- Encapsulation

Exception Handling in Python:-

- Exceptions Handling Introduction.
- Avoiding code break using exception handling.
- Handling various exceptions using try....except...else.
- Try-finally clause.
- Try-except-finally with return keyword.
- Argument of an Exception and create self-exception class.
- Exception Classes Hierarchy
- Raising an exceptions
- Custom(User-Defined) Exceptions.

Decorators and Debugging Python Programs:-

- Iterables
- Generators
- Yielding from the generators
- Inner Functions

- Decorators
- Debug Python programs using pdb debugger.
- Assert for debugging.

Structured Query Language:-

- MYSQL Introduction
- Data Types
- DDL, DML, TCL
- Constraints
- DISTINCT Clause
- WHERE Clause
- MYSQL Conditions (AND, OR, BOOLEAN, LIKE, IN)
- MYSQL Functions (MIN, MAX, AVG, SUM, COUNT)
- ORDER BY Clause
- GROUP BY Clause
- Relationships in SQL
- Joins in SQL

Python Database Connectivity (PDBC):-

- SQL Database connection using python.
- Install the MySQL dB and other Packages
- DML and DDL Operations with Databases.
- Performing Transactions.
- Handling Database Errors.
- Disconnecting Database.
- CRUD Operation Project using PDBC.

SQLAlchemy - Object Relational Mapper :-

- ORM Introduction
- SQLAlchemy Overview.
- SQLAlchemy over PDBC.
- Advantages of SQLAlchemy.
- Classical Way of Mapping
- Declarative Way of Mapping
- DML and DDL Operations with Database.
- Queries in SQLAlchemy.
- Applying Filters.

Multithreading in Python Programs:-

- What is multithreading?
- Single v/s Multithreaded Apps
- Starting a New Thread.
- Forking threads.
- The Threading Module.
- Class level & Object level Locks
- Synchronizing Threads.

HTML5:-

- Introduction
- Basic Formatting Tags
- Grouping Using Div Span
- Lists
- Images
- Hyperlink

- Table
- Iframe
- Form
- Headers
- Miscellaneous

CSS2:-

- Introduction
- Syntax
- Selectors
- Color Background Cursor
- Text Fonts
- Lists Tables
- Box Model
- Display Positioning
- Floats

JavaScript:-

- Introduction
- JS in HTML
- Function Calling
- JS Events
- Event Handling
- Table Creation

Angular 10:-

- Angular installation
- Types of Data Bindings in Angular
- Decorators used in Angular

- Pipes
- Directives
- Services
- Dependency Injection
- Singleton & Prototype Service
- Angular CRUD
- JSON Server
- Angular Material
- Angular Forms
- Reactive Forms with Validation
- Routing
- Lazy Loading/Feature Module
- Angular Project

Flask Framework:-

- What is Framework & Web-Framework?
- Overview of Flask Web App with python.
- Installation of Flask and Demo Application.
- Routing in flask.
- Templates
- Static Files
- Cookies & Sessions.
- Databases Connectivity in Flask.
- Flask-SQLAlchemy ORM
- Authentication(Login, Logout, SignUp)

Django Framework:-

- Basic of Django Framework & its uses.
- Installation and setting up Django.
- Django with PyCharm CE.
- Virtual Environments.
- Templates in Django & Template Inheritance
- Context in Django
- Static Files in Django.
- Syntax and URL.
- Routing in Django.
- Request/Response Architecture in Django.
- Models
- Relationships in Models
- Handling various Databases in Django
- Django-ORM
- Queries of Django-ORM
- Function-based Views.
- Class-based Views.
- Forms – HTML,Model & Django Forms.
- Crispy Forms
- CRUD Operations using Model Forms
- Form Validation
- Custom user models.
- Cookies & Session in Django.
- User Authentication(Login, Logout, SignUp)

API(Application Programming Interfaces):-

- Introduction
- Serialization & Deserialization
- Python JSON Module
- API
- Web API / Web Services
- REST & RESTful API
- Basics of SOAP API, REST API

Django REST Framework:-

- Introduction & Installation
- Requests
- Response
- API using api_view decorator and APIView class
- Serializer
- ModelSerializer
- CRUD using DRF
- API using GenericApiView and mixins
- ViewSets
- Routers
- Authentication and Permissions
- Throttling
- Pagination
- Testing API using Postman

Version Control System (GIT & GITHUB):-

- What is Git and GitHub?
- Account Setup
- Installing SourceTree Git GUI
- Git File Management
- Branching
- Pull Requests
- Common Workflows
- Git Branching
- Inside a Local Repository

Live Project

- Handle various phases of SDLC.
- Enhance your debugging skills.
- Be technically strong in particular domain.
- Flex your muscles on the cutting edge technology.
- Get help from passion driven industry experts who has diverse experience.
- Develop end-to-end project by yourself.

****