Python Syllabus

A) Core Python

1] 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.

2] 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.

3] Logic Building :-

- Condition Based Problems.
- Looping Related Problems.
- Numeric Logical Problems.
- String Logical Problems.
- Sorting Problems.
- Design Patterns.

4] 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
- Performance Assessment - 1

B) Advance Python

1] 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
- Mini Project-1
- Performance Assessment – 2

2] Exception Handling in Python:-

- Avoiding code break using exception handling.
- Exceptions Handling Introduction.
- Handling various exceptions using try....except...else.
- Try-finally clause.

- Try-except-finally with return keyword.
- Argument of an Exception and create self-exception
- Exception Classes Hierarchy
- Raising an exceptions
- Custom (User-Defined) Exceptions.

3] File Input and Output in Python:-

- Reading and writing text files.
- Reading config files in python.
- Writing log files in python.
- Understanding built-in functions.
- Writing Binary Files Manually.
- Using Pickle to Write Binary Files.
- Manipulating file pointer using seek.

4] Decorators, Iterators and Comprehensions:-

- Iterables
- Generators
- Yielding from the generators
- Inner Functions
- Decorators
- Comprehensions - List, Set & Dict.

5] 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. Performance Assessment -3

C) Backend

1] 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. Mini Project - 2

2] 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. Mini Project – 3

3] 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

4] 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.
- 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
- Mini Project - 6
- Form Validation
- Custom user models.
- Cookies & Session in Django.
- User Authentication (Login, Logout, SignUp)
- Mini Project - 7
- Performance Assessment -6

Project

- Core Project
- Live Project

Certificate