CORE PYTHON

- What is Python ?
- Uses of Python
- Features of Python

PYTHON ENVIRONMENT

- Installation of Python
- Path Settings of Python
- Python Shell, Editors , IDLE's
- Basic Syntax
- Running Python Scripting
- Getting Started
- Language fundamentals
- Keywords
- Data types
- Static Data types Vs Dynamic Data types
- Collection Types
- Mutable Objects vs Immutable Objects
- Naming Conventions
- Print( ) , type( ), id( ) functions
- Input( ) and Raw_input( )
- Type conversion Function
- Del keyword

STRING HANDLING

- What is String ?
- String-quoted String literals
- Triple-quoted String Literals
- String Indexing
- String Slicing
- Working with String Functions
- Working With String methods
- String multiplication

OPERATORS

- Arithmetic Operators
- Relation Operations
- Logical Operations
- Assignment Operators
- Short hand Assignment Operators
- Membership Operators
- Bitwise Operators
- Identity Operators
- Evaluating expressions

FLOW CONTROLS

- Conditional Statements
- Simple if
- If..else
- If..elif
- Looping Statements

- While Loop
- While..else
- For loop
- Using Range( ) in for loop
- Nested Loops
- Break Statement
- Continue Statement
- Pass Statement

COLLECTIONS

- Introduction Of Collections
- List
- Tuple
- Set
- Dict
- Collection indexing and slicing
- Functions for all Collections
- Working with methods of Collections
- Nested Collections
- Implementation of SET operations
- Getting dictionary values
- Reading file data into a dictionary
- Counting with dictionary
- Difference between list , set and dictionary
- Sorting Collections

FUNCTIONS

- Defining a function
- Calling function
- Types of arguments
- Default arguments
- Non default arguments
- Keyword arguments
- Non keyword arguments
- Arbitrary arguments
- Return statement in function
- Handling returning values
- Global vs local variables
- Scope of global variables and local variables
- Call by values
- Call by reference
- Passing function to functions
- Lambda functions / anonymous functions

MODULES

- What is modules ?
- Types of Modules
- The import Statement
- Module Aliases / renaming a module

- ➢ From .. Input
- ➢ Reloading a module
- ➢ Dir( ) function
- ➢ Creating user defined module
- ➢ Command line arguments
- ➢ Working with pre defined standard modules(Math, Random, Datetime,Os,Sys,String....)
- ➢

PACKAGES
- ➢ Introduction of packages
- ➢ __init__.py file
- ➢ Defining packages
- ➢ Importing from packages
- ➢ Importing from sub packages

EXCEPTION HANDLING
- ➢ Syntax Errors
- ➢ Runtime Errors
- ➢ What is an Exception ?
- ➢ Need of Exception Handling
- ➢ Predefined Exceptions
- ➢ Try, expect, and finally block
- ➢ Handling Multiple Exceptions
- ➢ Nested try, expect and finally blocks
- ➢ Named Exceptions
- ➢ User defined Exceptions
- ➢ Raise Statements

FILE HANDLING
- ➢ What is a file ?
- ➢ Opening a file
- ➢ Reading data from a file
- ➢ Writing Data into a file
- ➢ Closing a file
- ➢ Working with File Methods
- ➢ Working with directories

ADVANCED PYTHON
- ➢ OOPS Concepts
- ➢ Why oops ?
- ➢ What is Difference Between POP and OOPS
- ➢ Encapsulations
- ➢ Defining Classes
- ➢ Creating an Objects
- ➢ Class variables and instance variables

- ➢ Constructors
- ➢ Defining methods
- ➢ Deference between function and methods
- ➢ Destructors
- ➢ Object reference count
- ➢ Object Garbage Collections

INHERITANCE
- ➢ Types of inheritances
- ➢ Polymorphism (Overloading and Overriding)
- ➢ Super( ) statement
- ➢ Data hiding
- ➢ Built in properties of Class
- ➢ Inner Classes

REGULAR EXPRESSIONS
- ➢ What is Regular Expressions
- ➢ Simple character matches
- ➢ Special character matches
- ➢ Quantifiers
- ➢ Grouping and Match Objects
- ➢ Match( ), search( ) and sub( )
- ➢ Splitting a String
- ➢ Replacing Text

DATABASE ACCESS
- ➢ What is Database
- ➢ Difference Between Files and Databases
- ➢ Difference Between DBMS and RDBMS
- ➢ Creating users and Grating Privileges
- ➢ SQL Commands
- ➢ Working with Stored Procedures
- ➢ Transaction Management

MULTI THREADING
- ➢ Intro
- ➢ About multi Tasking
- ➢ Difference between Thread based multi tasking and Process Based Multi Tasking
- ➢ Defining Thread
- ➢ Starting Thread
- ➢ Thread safety using synchronization
- ➢ Joining of Threads

- ➢ Python iterators
- ➢ Python Generators
- ➢ PIP
- ➢ GUI Programming
- ➢ Data Analytics
- ➢ Pandas
- ➢ Numpy
- ➢ Matplotlib .....