

| Subject Code | Subject Name       | Credits |
|--------------|--------------------|---------|
| 26CS907      | PYTHON PROGRAMMING | 4       |

**Course Objectives:**

1. To read and write simple Python programs.
2. To develop Python programs with conditionals and loops.
3. Understanding and applying the concepts of functions, arrays
4. Introduction to Object oriented paradigm.
5. To do input/output with files in Python and how to handle exceptions in Python.

**Learning Outcomes:**

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

1. Read, write, execute by hand simple Python programs.
2. Structure simple Python programs for solving problems.
3. Decompose a Python program into functions.
4. Structure python programs using OOP's design for problem solving.
5. Read and write data from/to files in Python Programs.
6. Handle types of exceptions that occur during execution of programs.

**Unit 1 – Introduction (12 hrs.)**

Introduction to Python and installation: data types: Int, float, Boolean, string, and list; variables, expressions, statements, precedence of operators, comments; modules, functions: function and its use, flow of execution, parameters and arguments.

**Unit 2: Control flow and loops (12 hrs.)**

Control statement and Loops: Conditionals: Boolean values and operators, conditional (if), alternative (if-else), chained conditional (if-elif-else); Iteration: while, for, break, continue.

**Unit 3 - Functions and arrays (12 hrs.)**

Functions and arrays: Fruitful functions: return values, parameters, local and global scope, function composition, recursion; Strings: string slices, immutability, string functions and methods, string module; Python arrays, Access the Elements of an Array, array methods.

**Unit 4 – Object Oriented Programming (12 hrs.)**

Concept of OOPs in Python, Classes and Objects, Class properties, Class methods, Inheritance, Polymorphism, Encapsulation, Nested Classes

**Unit 5 – Files and Exceptions (12 hrs.)**

Exceptions: Catching Exceptions, Handling Exceptions, raising your own exceptions, writing our first file, reading a file line-at-a-time, turning a file into a list of lines, Reading the whole file at once. Working with binary files, Directories.

**Reference Books:**

1. Dusty Phillips, Python 3 Object-Oriented Programming, Packt Publishing.
2. Steven F. Lott, Mastering Object-Oriented Python, Packt Publishing.
3. Luciano Ramalho, Fluent Python, O'Reilly Media.
4. Mark Lutz, Learning Python, O'Reilly Media.
5. Allen B. Downey, Think Python: How to Think Like a Computer Scientist, O'Reilly Media