Introduction to Python Programming
01CI0624

Objective of the Course:

- To acquaint students with the basics of python programming and its functions.
- To understand the reading from and writing to I/O files in Python.
- To Explore common data structures, decision making and loops in Python.
- To develop various application and programs for solutions of problems related to civil engineering.

Credit Earned: 04

Prerequisite: Introduction to Programming

Students learning outcomes:

After successful completion of the course, it is expected that students will be able to,

1. Understand Python syntax fundamentals, Python variable types and Python control flow statements.
2. Demonstrate proficiency in effectively managing strings, functions, and lists within the Python programming context.
3. Apply commonly used operations related to file systems, showcasing competence in file handling.
4. To Compare Pandas and NumPy libraries for performing the scientific computing.

Teaching and Examination Scheme

Teaching Scheme (Hours)			Credits	Theory Marks			Tutorial/ Practical Marks		Total Marks
Theory	Tutorial	Practical		ESE (E)	IA (M)	CSE (I)	Viva (V)	Term Work (TW)	
03	00	02	04	50	30	20	25	25	150

Detailed Syllabus

Sr No.	Title of the unit	Number of hours
1	**Basics of Python**	**14**
	1.1 **Introduction to Python:** Getting Started with Python Programming, Overview of Python's significance in programming and its application in civil engineering.	04

	1.2 **Variables and Data Types:** Understanding Python variables and their role, exploring fundamental data types: int, float, str, bool, Utilizing Python's operators for various tasks, Performing computations with Arithmetic, Assignment, Comparison, Logical, Membership, Identity, and Bitwise operators, Practical applications of operators in list, tuple, and string operations.	04
	1.3 **Decision Making and Loops:** Logical decisions with if, if...else, and elif statements, incorporating Boolean expressions for effective decision-making, while loops for iterative tasks, Employing For loops for efficient sequence traversal, Use of Nested loops, Infinite loops, and loop control statements (break, continue, pass), Integrating built-in functions for list, set, and dictionary manipulation, Visualizing data & plotting techniques.	06
2	**Python Functions, Modules and File Operations**	**08**
	2.1 **Defining Custom Functions:** The role of functions in code organization, Creating and implementing custom functions.	01
	2.2 **Advanced Function Techniques:** Variable scope and referencing, Skills for working with lists, tuples, dates, times, and dictionaries.	02
	2.3 **Importing and Using Modules:** Organizing code using external modules, Importing and leveraging external packages, applications using functions, module external packages.	03
	2.4 **File Operations:** Understanding file input/output operations, Handling Different File Types, Utilizing text, CSV, and binary files, Practical Exercises and Illustrative Programs.	02
3	**Scientific Computing Using Python & Data Visualization**	**20**
	3.1 **Basics of NumPy and SciPy**: Introduction to Numerical Computing, Installation and setup of NumPy and SciPy, Creating arrays and basic operations in NumPy.	03
	3.2 **Array Manipulation and Operations**: Advanced array manipulation in NumPy, Slicing and indexing arrays for data extraction, Matrix operations and linear algebra using NumPy, Exploring special functions and random numbers generation	03
	3.3 **Linear Algebra and Nonlinear Equations**: Utilizing SciPy for linear algebra operations, solving systems of linear equations, numerical techniques for solving nonlinear equations	03
	3.4 **Numerical Integration and ODE Solving**: Understanding and solving numerical integration with SciPy.	03
	3.5 **Data Manipulation and Analysis with Pandas:** Introduction to Pandas, Reading data from files into Pandas Data Frames, Understanding and exploring Pandas Series and Data Frame structures, Extracting information from Pandas Data Frames, Grouping and aggregation operations in Pandas.	04
	3.6 **Plotting:** An Interactive Session with PyPlot, Basic Plotting, Logarithmic Plots, Plots with multiple axes, Mathematics and Greek symbols, Structure of matplotlib, Contour and Vector Field Plots.	04
	Total	**42**

List of Experiments

Sr. No	Topic name	No. of Hours
1	Introduction to Jupyter Notebook and python script in CMD.	2
2	Create programs to understand how loops and conditions work in Python.	4
3	Develop programs to grasp the concepts of functions and working with variables.	4
4	Explore handling text and creating lists in Python through practical exercises.	2
5	Learn how to read from and write to files in Python, understanding file operations.	4
6	Get an introduction to basic data analysis concepts using Python.	4
7	Learn to create simple visualizations, like charts and graphs, to understand data better.	4
8	Understand the use of popular Python libraries like NumPy and Pandas for data manipulation.	4

Suggested Theory Distribution

The suggested theory distribution as per Bloom's taxonomy is as per follows. This distribution serves as guidelines for teachers and students to achieve an effective teaching-learning process

Distribution of Theory for course delivery and evaluation					
Remember	Understand	Apply	Analyze	Evaluate	Create
5%	20%	30%	25%	10%	10%

Instructional Method and Pedagogy:

1. At the start of course, the course delivery pattern, and prerequisite of the subject will be discussed.
2. Lectures will be taken in class room with the use of multi-media presentations, white board– mix of both.
3. Attendance is compulsory in lectures and laboratory which carries a 5% component of the overall evaluation.
4. Minimum two internal exams will be conducted and average of two will be considered as a part of continuous evaluation
5. Assignments based on course content will be given to the students at the end of each unit/topic and will be evaluated at regular interval. It carries a weightage of 5%.
6. Surprise tests/Quizzes will be conducted which carries 5% component of the overall evaluation.
7. The course includes a laboratory, where students have an opportunity to build an appreciation for the concepts being taught in lectures.

Recommended Study Material

1. Gowrishankar S, Veena A, "Introduction to Python Programming", 1st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372
2. Mark Summerfield, "Programming in Python 3: A Complete Introduction to the Python Language", Pearson Education
3. Zed A. Shaw, "Learn Python the Hard Way: A Very Simple Introduction to the Terrifyingly Beautiful World of Computers and Code, 3rd edition", Adisson-Wesley
4. Erik Westra, "Modular Programming with Python: introducing modular techniques for building sophisticated programs using Python", Packt Publishing
5. David Love, "Tkinter GUI Programming by Example", Packt Publishing Ltd.
6. John Hunt "Advanced Guide to Python 3 Programming", Springer.

Web Links

1. https://docs.spyder-ide.org/
2. https://wiki.python.org/moin/BeginnersGuide
3. https://www.programiz.com/python-programming