

<b>Semester: 4</b>				
<b>Programme :</b> B.Sc. Computer Science (Hons)				
<b>Course :</b> PYTHON PROGRAMMING				
<b>Paper code:</b> S2CS230411P				<b>Credits: 3</b>
<b>Hours/week :</b> Practical: 4				
<b>Category:</b> Core/MDC/SEC/VAC : SEC				
<b>Theory / Practical / Composite :</b> Practical				
<b>No of Modules :</b> 1				
<p><b>Course Overview:</b> This comprehensive course is designed to take learners from the basics of Python programming to intermediate-level application development and data analysis. Starting with the fundamental syntax and logic, the curriculum progresses through essential data structures, modular programming, and file management. Uniquely, this course bridges the gap between general-purpose programming and data science by incorporating industry-standard libraries such as NumPy, Pandas, and Matplotlib. By the end of the course, students will possess the skills to write efficient Python scripts, manage complex data, and visualize insights.</p>				
<b>Course Outcome:</b>				
1. <b>Explain</b> the fundamental architecture of Python, including data types, operators, and expression evaluation.				
2. <b>Construct</b> logical programs using decision-making statements and iterative loops to solve algorithmic problems.				
3. <b>Implement</b> appropriate built-in data structures (Lists, Tuples, Sets, Dictionaries) to organize and manipulate data efficiently.				
4. <b>Analyze</b> numerical and tabular data by applying array operations and data manipulation techniques using NumPy and Pandas.				
5. <b>Design</b> modular and reusable code by defining parameterized functions, managing scope, and importing external modules.				
6. <b>Generate</b> data visualizations using Matplotlib and manage data persistence through file handling operations.				
<b>Prerequisites:</b>				
<ul style="list-style-type: none"> <li>• Basic programming proficiency</li> <li>• High school algebra and logical reasoning</li> <li>• Recommended: Familiarity with basic data structures (arrays, linked lists, stacks, queues)</li> </ul>				
<b>SYLLABUS</b>				
UNIT/Module	CONTENT	HOURS or NUMBER OF CLASSES	CO Mapping	COGNITIVE LEVEL
I.	Introduction to Python Programming	2	CO1	K2 (Understand)
II.	Operators, Expressions and Python Statements	2	CO1	K2 (Understand)
III.	Control statements (decision making and iterative)	4	CO2	K3 (Apply)
IV.	Set, Tuple, List, Dictionary	4	CO3	K3 (Apply)
V.	Data Types	2	CO1	K2 (Understand)

VI.	Arrays in Python using NumPy and Pandas Libraries	20	CO4	K4 (Analyze)
VII.	Functions and parameterized functions	6	CO5	K6 (Create)
VIII.	Handling graphs using matplotlib	6	CO6	K6 (Create)
IX.	File Handling	4	CO6	K6 (Create)
X.	Scope and Modules	2	CO6	K6 (Create)
<b>Text Books</b>				
1. Python: The Complete Reference by Martin C. Brown, McGraw Hill Education				
2. Head First Python by Paul Barry, O'Reilly				
3. Python Programming: Using problem solving approach by Reema Thareja, Oxford.				
4. Core Python Programming by R. Nageswara Rao, DreamTech.				
<b>Web Resource</b>				
1. <a href="https://onlinecourses.swayam2.ac.in/cec22_cs20/preview">https://onlinecourses.swayam2.ac.in/cec22_cs20/preview</a>				
<b>Evaluation</b>				
<b>Paper Structure for Practical Semester Exam Module:</b>				
Hands-On Assignment based evaluation				

### Course outcomes (COs) and Cognitive Level Mapping

COs	CO Description	Cognitive levels
CO1	<b>Explain</b> the fundamental architecture of Python, including data types, operators, and expression evaluation.	K2 (Understand)
CO2	<b>Construct</b> logical programs using decision-making statements and iterative loops to solve algorithmic problems.	K3 (Apply)
CO3	<b>Implement</b> appropriate built-in data structures (Lists, Tuples, Sets, Dictionaries) to organize and manipulate data efficiently.	K3 (Apply)
CO4	<b>Analyze</b> numerical and tabular data by applying array operations and data manipulation techniques using NumPy and Pandas.	K4 (Analyze)
CO5	<b>Design</b> modular and reusable code by defining parameterized functions, managing scope, and importing external modules.	K6 (Create)
CO6	<b>Generate</b> data visualizations using Matplotlib and manage data persistence through file handling operations.	K6 (Create)