

<b>Programme</b>	B.Sc. IT Honours (Data Science)			<b>Branch</b>	Computer Applications				
<b>Semester</b>	VI			<b>Version</b>	1.0.0.0				
<b>Effective from Academic Year</b>			2026-27		<b>Effective for the batch Admitted in</b>		June 2024		
<b>Subject code</b>		U76A1PY2		<b>Subject Name</b>		PYTHON PROGRAMMING - II			
<b>Teaching scheme</b>					<b>Examination scheme(Marks)</b>				
<b>(Per week)</b>	<b>Lecture (DT)</b>		<b>Practical (Lab.)</b>		<b>Total</b>	<b>CCE</b>	<b>SEE</b>	<b>Total</b>	
	L	TU	P	TW					
Credit	2	-	2	-	4	Theory	50	50	100
Hours	2	-	4	-	6				

**Objective:**

To develop proficiency in object-oriented programming, data handling, and visualization techniques using Python

**Pre-requisites:**

Basic understanding of programming concepts and logical reasoning

**Learning Outcome:**

Name of CO	Description
CO1	Apply basic object-oriented programming concepts for structured Python development
CO2	Implement core OOP features and handle exceptions effectively in Python programs
CO3	Utilize NumPy for efficient numerical computing and data manipulation in Python
CO4	Apply Pandas for effective data handling, transformation, and analysis in Python
CO5	Create and customize data visualizations for effective data analysis and presentation

**Mapping of CO and PO:**

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	2	1	2	2	2	1	1	0	0	1
CO2	3	3	2	1	2	3	2	1	1	0	0	1
CO3	3	3	2	2	3	1	2	1	1	0	0	1
CO4	3	3	3	2	3	1	2	1	1	0	0	1
CO5	3	2	3	2	3	1	2	1	2	1	0	1

**Content:**

Unit	Content	Hrs.
1	<b>Introduction to Object-Oriented Programming (OOP Basics):</b> Need for Object-Oriented Programming, Comparison between Procedural and Object-Oriented Paradigms, Defining Classes and Creating Objects, The <code>_init_()</code> Constructor, Instance Variables and Methods, Class Variables and Static Methods, Method Overloading using Default Arguments, Access Modifiers, Understanding the self Keyword and Object Reference	06
2	<b>OOP Features and Exception Handling:</b> Inheritance Types, Method Overriding and use of <code>super()</code> Keyword, Basics of Polymorphism, Concept of Encapsulation and Data Hiding, Abstraction using Classes and Methods, Exception Handling Concepts, Difference between Syntax and Runtime Errors, Creating User-defined Exception Classes	06
3	<b>NumPy – Numerical Computing:</b> Introduction to NumPy Arrays – Creation and Types, Array Indexing, Slicing, and Reshaping, Arithmetic and Statistical Operations, Broadcasting and Aggregation Techniques, Performance Comparison between NumPy Arrays and Python Lists	06

4	<b>Pandas – Data Handling:</b> Introduction to Pandas Series and DataFrame, Data Import and Export using CSV and Excel Files, Data Cleaning and Handling Missing Values, Data Sorting, Filtering, and Grouping, Merging and Joining DataFrames	06
5	<b>Data Visualization with Matplotlib and Seaborn:</b> Introduction to Data Visualization and Its Importance, Plotting with Matplotlib – Line Chart, Bar Chart, Pie Chart, Histogram, and Subplots, Customizing Plots Data Visualization with Seaborn – Distribution Plots (distplot, histplot, kdeplot), Relationship Plots (scatterplot, lineplot), Categorical Plots (boxplot, barplot), Heatmaps and Pairplots for Multivariate Analysis, Saving and Exporting Charts in Various Formats	06
<b>Practical Content:</b>		
List of programs specify by subject teacher based on above mention topics.		
<b>Reference Books:</b>		
1	Introduction to Computation and Programming Using Python by John V. Guttag (3rd Edition) MIT Press	
2	Python for Data Analysis by Wes McKinney (2nd Edition), O’Reilly Media	
3	Data Structures and Algorithmic Thinking with Python by Narasimha Karumanchi (1st Edition) CareerMonk Publications	
<b>Web Reference:</b>		
1	<a href="https://www.w3schools.com/python/">https://www.w3schools.com/python/</a>	
2	<a href="https://www.tpointtech.com/python-tutorial">https://www.tpointtech.com/python-tutorial</a>	
3	<a href="https://www.geeksforgeeks.org/python/python-programming-language-tutorial/">https://www.geeksforgeeks.org/python/python-programming-language-tutorial/</a>	
<b>MOOC/Certificate Course:</b>		
1	<a href="https://www.coursera.org/learn/object-oriented-python">https://www.coursera.org/learn/object-oriented-python</a>	
2	<a href="https://nptel.ac.in/courses/106106145">https://nptel.ac.in/courses/106106145</a>	
3	<a href="https://www.udemy.com/course/object-oriented-python-programming/?srsltid=AfmBOorYT_c9GMnLjhHhHmDIWDqs_Y5SCIOXeWV8XggzvbYgdjAP10">https://www.udemy.com/course/object-oriented-python-programming/?srsltid=AfmBOorYT_c9GMnLjhHhHmDIWDqs_Y5SCIOXeWV8XggzvbYgdjAP10</a>	
<b>Question Paper Scheme:</b>		
<p><b>End Semester Examination Duration:</b> (2 Hours Theory Examination)</p> <p><b>Note for Examiner: -</b>  Q-1 Any Five out of Seven (25 Marks)  Q-2 Any Two out of Three (06 Marks)  Q-3 Mandatory question (05 Marks)  Q-4 Any Two out of Three (08 Marks)  Q-5 Any Two out of Three(06 Marks)</p> <p>*The question paper must comprehensively address all Course Outcomes (COs), align with Bloom's Taxonomy levels, and ensure complete syllabus coverage.</p>		