



SEMESTER – V

BRANCH	Subject	Subject Code
B.Sc. (Computer Science)	Data Analysis and Visualization with Python (Theory)	CSDSE1-151

Course learning outcomes (CLO):-

1. After studying this subject, students shall be able to- Interpret the fundamental Python syntax and semantics and be fluent in the use of Python control flow statements.
2. Express proficiency in the handling of strings, functions and file handling.
3. Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples and sets.
4. Develop proficiency in using NumPy for data manipulation.
5. Create a variety of data visualizations using Matplotlib.
6. Apply NumPy and Matplotlib to analyze and visualize real-world datasets
7. Complete a hands-on project involving data manipulation and visualization,

Unit – I

Python Basics: Python interpreter, Python idle, dynamically typed and strongly typed features, basic data types, variables, expressions, statements, operators, flow of execution. Input and Output statements, Conditionals: Boolean values and operators, conditional (if), alternative (else), chained conditional (if-elif-else). Iteration: while, for, break, continue, pass, implementing 'for' through range(), 'in' and 'not in' operators for sequence traversal. Creating and executing.py scripts.

Unit – II

Data Structures: Lists append, extend, insert, index, remove, pop, count, sort, reverse, slicing, list comprehension, Copying a list: deep copy, shallow copy. Tuples- index, count, usage, use of tuples as a swap function. Dictionaries-keys, values, tuples, nested dictionaries, dictionary comprehension, Strings-Single line and multi-line strings, formatter, isdigit, isalpha, isalnum, islower, istitle, ispace, utle, lower, upper, strip, split, splalines, join etc. Sets-union, intersection, subset, superset, difference, symmetric difference, copy, add, remove, discard etc. Functions & File Handling Inbuilt Functions id, len, chr, ord etc, defining and calling a function, arguments, global versus local variables, defining and using lambda functions, the map(), filter(), reduce() functions.

Unit – III

NumPy: Introduction to Numpy, NumPy array in Python, Basics of NumPy Arrays, comparison of Python Lists with Numpy Arrays.Array: Array Creation, The Arrange Method, The Zero Method, Numpy array filled with all ones, The linspace Method, The eye Method, Numpy Meshgrid function, empty and full NumPy array, Numpy array filled with all zeros, Numpy array filled with all ones, 2-D Gaussian array,



Creating vector in Python using NumPy. Array Indexing. Array Slicing, Data Types, Copy vs View, Array Shape, Array Reshape, Array Iterating, Array Join, Array Split, Array Search, Array Sort, Array Filter, Concatenation of two arrays, Splitting and Comparison of Arrays. Binary Operations, Mathematical Function, String Operations

Unit – IV

Matrix in NumPy: Matrix manipulation in Python, empty() function, zeros() function, ones() function, eye() function, identity() function, Adding and Subtracting Matrices in Python. Vector Multiplication, Dot product of two arrays. Operations on NumPy Array: Broadcasting with NumPy Arrays, Sorting, Searching and Counting of NumPy arrays. Variations in different Sorting techniques in Python. Universal Functions: Creation of ufuncs, Simple Arithmetic, Rounding. Trigonometric, Hyperbolic, Set functions

Unit – V

Data Visualization with Matplotlib: Overview of Matplotlib and its capabilities, Creating line plots and scatter plots, Customizing: labels, titles, colors, legends, Creating bar plots and histograms, Adding annotations and text to plots, Creating subplots and multiple plots, Saving and exporting plots.

Advanced Data Visualization Creating pie charts and box plots, Visualizing 3D data with Matplotlib, Interactive visualization using widgets.

Practical's

1. Find all numbers which are multiple of 17, but not the multiple of 5, between 2000 and 2500.
2. Print the first 2 and last 3 characters in a given string. Use the string slicing.
3. Write a program that eliminates duplicates in a list.
4. Implement shallow copy and deep copy of a list.
5. Find the largest of n numbers, using a user defined function largest()
6. Write a function that capitalizes all vowels in a string.
7. Read a line containing digits and letters. Write a program to give the count of digits and letters.
8. Write a function my Reverse() which receives a string as an input and returns the reverse of the string.
9. Use the list comprehension methodology in python, to generate the squares of all odd numbers in a given list.
10. Create a NumPy array with values from 1 to 20.
11. Create a 3x3 identity matrix using NumPy,
12. Generate an array of 10 random integers between 0 and 100.
13. Calculate the mean, median, and standard deviation of an array
14. Reshape a 1D array into a 2D array.
15. Filter even numbers from an array using boolean indexing.
16. Calculate the dot product of two matrices.
17. Normalize an array to have values between 0 and 1.
18. Calculate the sum along both rows and columns of a 2D array.
19. Perform element-wise multiplication and division between arrays.
20. Generate a dataset of x and y values and plots it.
21. Analyze and visualize a simple data set (e.g., student grades) using both libraries.
22. Create a bar plot showing comparison of data from two different sources.
23. Visualize data from a CSV file using NumPy and Matplotlib.
24. Generate a contour plot of a 2D function.
25. Analyze and visualize trends in a dataset over time.
26. Create an interactive plot using Matplotlib's interactive mode.