

SEMESTER-IV

COURSE 10: DATA SCIENCE USING PYTHON

Theory

Credits: 3

3 hrs/week

Course Objectives:

1. To introduce students to the foundational concepts and significance of data science in modern business and technology contexts.
2. To develop proficiency in Python programming, focusing on data types, control structures, and core operations.
3. To equip students with skills in handling and manipulating structured and unstructured data using Python libraries.
4. To enable students to perform exploratory data analysis and data cleaning using tools like NumPy and Pandas.
5. To foster the ability to visualize data effectively using Python libraries such as Matplotlib and Seaborn.

Course Outcomes:

1. Understand the role of data science, its components, and its relationship with AI and machine learning.
2. Apply Python programming constructs such as strings, lists, tuples, sets, and dictionaries to solve basic data problems.
3. Demonstrate the ability to perform data manipulation and cleaning using NumPy and Pandas.
4. Conduct exploratory data analysis and extract insights from datasets using Python-based tools.
5. Create and customize data visualizations using Matplotlib and Seaborn to communicate analytical findings.

Unit-I: Introduction to Data Science

Importance of Data Science: Need for Data Science, What is Data Science?, Data Science process, Business Intelligence and Data Science, Components of Data Science, Relationship between Artificial Intelligence, Machine Learning and Data Science.

Categories of Data: Unstructured data, Structured data, Semi Structured data, Time series data, Sources of data: APIs, Web pages, Databases, Files, Taxonomy of types of data.

Data Scientist: Responsibilities of a data scientist, Skills required for a data scientist, Programming and Analysis tools for Data Science.

Case Study: Mapping Data Categories to Banking Operations.

Unit-II: Basic Python: Part-I

Features of Python, writing and executing python program, Indentation, Constants, Identifiers, Variables, Keywords, Data types, Input/Output, Operators. Conditional statements, Looping Statements.

Strings: Definition, Positive, Negative indexing, String slicing, String Operations: Concatenation, Repetition, Membership(in, not in), Comparison, String Methods: upper(), lower(), strip(), replace(), split(), join(), find().

Lists: Definition, Indexing, Slicing, Negative Indexing, Nested Lists, Modifying Lists:

Updating, Inserting, Deleting elements, List Functions: len(), max(), min(), sorted(), Looping through Lists.

Unit-III : Basic Python: Part-II

Tuples : Definition, Indexing, Slicing, Nested tuples, Tuple Operations: Concatenation, repetition, membership testing, Tuple Methods: count(), index().

Sets: Definition, Accessing Set elements, Modifying Sets, Set Operations: Union, Intersection, difference, Symmetric Difference.

Dictionaries: Definition, Accessing elements, Modifying Dictionaries, Dictionary methods.

Functions: User defined functions, built-in functions, lambda functions, recursive functions.

Modules: math, random.

Unit-IV : Advance Python : Part-I

Exploratory Data Analysis: What is EDA, Steps involved in EDA, Python Libraries used in EDA.

Numpy: Basics, Array Attributes, Slicing (1D, 2D and 3D), Numpy Array Iteration, Operations on Arrays: Concatenating Arrays , Reshaping Arrays , Splitting Arrays , Statistical Operations on Arrays.

Pandas: Basics, Series, DataFrame, Comparison between Series and DataFrame. Reading data from varied data sources into python DataFrame: Read from Excel Data Source, Read data from .csv file, Read data from Dictionaries.

Data Cleaning : Handling missing values, Removing duplicates, Correcting Inconsistencies.

Dataset Repositories for Data Science & Visualization : Kaggle Datasets, UCI ML repository.

Case Study: Retail Sales Analysis using Kaggle Dataset.

Unit-V : Advance Python : Part-II

Data visualization : Introduction to Data Visualization, Benefits of Data Visualization, Popular Python Data Visualization Libraries.

Matplotlib : Introduction, Basic Plots: Line plot, Bar chart, pie chart, Histogram, Scatter plot, Plot Customization: Titles, labels, and legends, Gridlines and ticks, Colors, markers, and line styles, Subplots and Layouts.

Seaborn : Introduction, Comparison with matplotlib, Generating Plots: Heatmaps, Boxplots.

Case Study: Visualizing Financial data.

Text Books:

1. Kooper, Steven. *Data Science from Scratch*. 2nd Edition. O'Reilly Media, 2019.
2. Vasiliev, Yuli. *Python for Data Science: A Hands-On Introduction*. No Starch Press, 2022.
3. Acharya, Seema. *Reimagining Data Visualization Using Python*. Chapman & Hall/CRC, 2022.

SEMESTER-IV

COURSE 10: DATA SCIENCE USING PYTHON

Practical

Credits: 1

2 hrs/week

List of Experiments:

1. Write a Python program to demonstrate basic data types in python.
2. Write a Python program to illustrate arithmetic operators.
3. Write a Python program to find the maximum of three numbers.
3. Write a Python Program to find the sum of digits of a number.
4. Write a Python Program to check whether a given number is palindrome or not.
5. Write a Python program to check whether a given number is prime or not.
6. Write a Python program to calculate Simple Interest and Compound Interest.
6. Write a Python program to illustrate Strings and String methods.
7. Write a Python program to illustrate string slicing.
8. Create a list and perform the following methods.
 - a) append() b) extend() c) insert() d) pop() e) remove() f) clear
9. Create a list and perform the following functions.
 - a) len() b) max() c) min() d) sorted()
10. Write a program to input n numbers from the user. Store these numbers in a tuple.
Print the maximum and minimum number from this tuple.
11. Create a dictionary and apply the following methods.
 - a) Print the dictionary items b) Access items c) Use get() d) Use get() e) Use len()
12. Write a Python program to find the factorial of a number using functions.
13. Write a Python program to find the factorial of a number using recursion.
14. Write a Python program to find the maximum of two numbers using lambda function.
15. Write a Python program to illustrate a *math* module.
16. Write a Python program to illustrate a *random* module.
17. Write a Python program to illustrate numpy array attributes: ndim, shape, size, dtype.
18. Write a Python program to illustrate numpy array statistical operations.
19. Write a python program to demonstrate Reading data from varied data sources into Pandas DataFrame.
 - a) Read Data from Excel Data Source and display.
 - b) Read Data from “iris.csv” into pandas DataFrame and display. [source: UCI ML repository]
 - c) Load a Python Dictionary into a DataFrame.
20. Consider any Dirty Dataset for Data Cleaning [Source: Kaggle Dataset Repository]
And perform the following.
 - a) Remove Rows that contain empty cells.
 - b) Clean Wrong Format.
 - c) Clean Wrong Data.
 - d) Remove Duplicates.
21. By using matplotlib generate Bar chart for revenue by product category.
22. By using matplotlib generate Pie chart for market share distribution of companies.
23. By using matplotlib generate Histogram for customer age distribution.
24. By using seaborn generate correlation Heatmap.
25. By using seaborn illustrate Boxplot.