NITTE EDUCATION TRUST | NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY

Department of Computer Science and Engineering

22CSE554- Data Science using Python
Practice Questions for MSE1
Course Coordinator: Dr.Vani V

Portion: Unit 1, 2 and Unit 3 - first half

1. Define data and explain its significance in data science with a real-world example where data has led to better decision-making.
2. Mention any two basic data types in Python and illustrate each with an example. Also explain how these are different in terms of storage/usage.
3. Explain how variables work in Python with an example. Discuss why variables are important in organizing data during analysis.
4. Write the purpose of an if statement in Python. Provide a simple code example and explain the output.
5. NumPy arrays are considered faster than Python lists. Explain why, with reference to memory allocation and vectorization concepts.
6. State the purpose of the Pandas library in Python and explain with an example how it simplifies data manipulation compared to plain Python lists/dictionaries.
7. Differentiate between Pandas DataFrame and a Python dictionary in terms of data organization and operations. Give suitable examples.
8. What is the purpose of a NumPy array? Create a 1D array and show how basic arithmetic operations (like addition and multiplication) can be applied directly.
9. Define data processing. Why is it considered an essential step in any data science project? Support your answer with a practical scenario.
10. List and explain two techniques to handle missing values in Pandas DataFrames. Justify why handling missing values is crucial for reliable analysis.
11. Differentiate between slicing and indexing in Pandas with examples. Which situations are suitable for each?
12. Explain the use of dropna() in Pandas. Demonstrate with a small DataFrame before and after applying the function.
13. What is the purpose of .loc[] and .iloc[] in Pandas? Explain with examples showing selection by label and by position.
14. Explain how fillna() helps deal with missing values. Provide an example of replacing missing values with column mean.
15. Discuss the role of the to_csv() method in Pandas. Provide an example of saving a DataFrame to CSV and explain how index handling works.
16. What is Matplotlib? Explain its role in Python data science projects with an example visualization.
17. State two key differences between a line chart and a bar chart in Matplotlib. In which scenarios would you prefer each?
18. Explain how a scatter plot helps in understanding relationships between two variables. Illustrate with a real-world example.
19. Discuss how a box plot can be used to identify outliers in a dataset. Provide a sketch/example.
20. Explain the purpose of a heatmap for visualizing correlation between variables. Why is it useful in data analysis?

21. Write a Pandas program to load an employee dataset (Name, Department, Salary) from a CSV file. Filter employees who earn more than 50,000 and belong to the HR department. Display only their names. Discuss how conditional filtering works in Pandas.
22. Generate a NumPy array of 15 random integers between 1 and 100. Extract elements greater than 50 using slicing/boolean indexing and calculate their mean. Show code and output.
23. Create a Pandas DataFrame with Region and Sales columns for 5 rows. Group sales data by region and calculate the total sales for each region. Explain how groupby() works internally.

Department of Computer Science and Engineering

22CSE554- Data Science using Python
Practice Questions for MSE1
Course Coordinator: Dr.Vani V

24. Construct a DataFrame with product sales data (Product, Sales_Q1, Sales_Q2, Sales_Q3, Sales_Q4). Compute total sales for each product and add it as a new column. Demonstrate the use of axis in Pandas operations.
25. Create a 1D NumPy array of 20 elements. Use element-wise operations to multiply each by 2. Print both original and modified arrays. Comment on vectorization.
26. Given employee data (Name, Gender, Age, Department) in a Pandas DataFrame, filter employees below 30 years in the CSE department. Display their names and ages. Explain how multiple conditions are applied in Pandas filtering.
27. Create a Pandas DataFrame with store sales data (Product, Price, Quantity). Add a column Total_Revenue = Price * Quantity. Explain column-wise operations in Pandas.
28. Using weather data (City, Temperature, Humidity, Wind Speed), compute average temperature and humidity for each city using Pandas. Show both groupby() and pivot_table() methods.
29. Given student marks data (Student, Subject, Marks), write a Pandas program to find the highest marks scored in each subject. Display the subject with maximum marks.
30. Create a COVID-19 dataset (Country, Cases, Deaths, Recovered). Calculate mortality rate (Deaths/Cases) and add as a new column. Display countries with mortality >5%.
31. Create a NumPy 2D array with values 10–30, reshape into 4x5, and compute column sums using np.sum(axis=0). Compare results with axis=1.
32. Generate a 3x3 NumPy array of random integers (5–20). Find minimum values in each row. Display both the array and results.
33. Create an array of 10 evenly spaced values between 0 and 5. Compute and display the mean and standard deviation.
34. Create a 3x3 identity matrix and multiply it element-wise with a random 3x3 matrix. Display original and resultant matrices.
35. Generate a 1D array of random values. Filter elements greater than 0.5 and count them. Display both the filtered array and count.
36. Create a 5x5 array of normally distributed values. Calculate variance column-wise using NumPy.
37. Using np.array(), create a 2D array from a list of lists. Find the sum of elements greater than 10 in each row.
38. Generate an array with 15 elements, reshape into 3x5, and compute median of each row. Display results.
39. Create a 4x4 matrix with integers 1–20. Verify if all elements are less than 25 using a comparison operator.
40. Create a 1D array of 20 random integers (10–100). Find its 75th percentile using NumPy.
41. Create a Pandas Series with indices (a, b, c, d) and values [0.25, 0.5, 0.75, 1.0]. Extract and display the value of index c.
42. Create a Pandas DataFrame with MultiIndex (State1, State2) and years (2000, 2010). Assign population data and unstack it into columns.
43. Using a MultiIndex DataFrame with subjects (Bob, Guido, Sue) and hierarchical columns (year, visit), generate mock health data. Display only Guido’s records.
44. Create a DataFrame with (state, year, population) and reset the index to flatten the structure.
45. Convert the flattened DataFrame back into a MultiIndex of states and years. Display the result.
46. Using a MultiIndex DataFrame with health data, compute mean values across rows and columns. Discuss difference between mean(axis=0) and mean(axis=1).
47. Concatenate lists [1,2,3], [4,5,6], [7,8,9] using pd.concat() into a Series. Display result.
48. Concatenate two Series with different indices using pd.concat(axis=1). Explain alignment of indices.
49. Create a DataFrame with duplicate indices. Resolve duplicates by grouping and summing values.
50. Load a JSON file into Pandas. Display first 10 rows with .head(). Show .shape and .info() outputs.
51. Write a Python program to read data from SQLite database into Pandas. Use SQL query to select rows from table diabetes and display them.
52. Create a hierarchical DataFrame and use .unstack() to convert one index level into columns. Display transformed result.

Department of Computer Science and Engineering

22CSE554- Data Science using Python
Practice Questions for MSE1
Course Coordinator: Dr.Vani V

53. Create a DataFrame with missing values. Replace them with column means using fillna(). Show before/after.
54. Load a CSV file into Pandas. Remove rows with missing values and display first 5 rows of the cleaned DataFrame.
55. Using a DataFrame, slice rows where Age > 30. Display only Name and Age.
56. Load JSON data into Pandas. Extract rows where status == "active" using .loc[].
57. Write a program to read SQLite DB into Pandas and filter rows where Salary > 50,000.
58. Using .iloc[], select first 3 rows and 2 columns of a sales DataFrame.
59. Load a CSV into Pandas. Group data by Department and calculate average salary.
60. Load JSON into Pandas. Replace missing values in Score with median. Display result.
61. Create a DataFrame with customer data (CustomerID, Name, Purchase Amount). Select customers with purchases above 1000.
62. Using Matplotlib, plot monthly sales trend (12 months) with line chart. Add labels, title, markers.
63. Create a bar plot comparing sales across 4 regions with colors and gridlines.
64. Create a box plot of test scores for 30 students. Add axis labels and title.
65. Plot a scatter chart for hours studied vs. scores. Add trend line.
66. Generate heatmap of correlation matrix (Temperature, Humidity, Wind Speed) with annotations.
67. Create a 3D scatter plot of (Age, Height, Weight) for 50 samples. Label axes, add legend.
68. Create a styled bar plot of revenue for 5 products. Add custom colors and value labels.
69. Plot daily temperatures for a week. Use dashed line for weekdays, solid line for weekends. Add legend.
70. Create box plot comparing scores across three subjects (AIML, OOP, DSP).
71. Generate a heatmap for a 4-class confusion matrix with color bar and annotations.