

Python Programming with NumPy and Pandas

1. Define a 1D, 2D, and 3D array with examples.
2. Explain the purpose of the reshape method in NumPy. Write a sample code to reshape a 1D array of size 12 into a 2D array with 3 rows and 4 columns.
3. Create a NumPy array with values from 1 to 20. Reshape it into a 4x5 matrix and find its mean, median, and standard deviation.
4. Using the Pandas DataFrame below, answer the following:

Col1	Col2	Col3
10	20	30
40	50	60
70	80	90

- a. Select the first two rows and last two columns.
 - b. Replace all values greater than 50 with NaN.
 - c. Find and print the sum of each column. (10 Marks)
5. Create a Python program to perform the following:
 - Generate a NumPy array with values ranging from 1 to 90, stepping by 3.
 - Reshape it into a DataFrame with 10 rows and 3 columns.
 - Select and print the rows 2, 4, and 6 along with columns 2 and 3 using iloc.
 6. Handling Missing Values
 - a. Create the following DataFrame:

```
data = {
    'names': ['Steve', 'John', 'Richard', 'Sarah', 'Randy', 'Michael', 'Julie'],
    'age': [20, 22, 20, 21, 24, 23, 22],
    'gender': ['Male', 'Male', 'Male', 'Female', 'Male', 'Male', 'Female'],
    'rank': [2, 1, 4, 5, 3, 7, 6]
}
```
 - b. Introduce missing values (NaN) in the age column for rows 2 to 5.
 - c. Demonstrate how to fill these missing values using:
 - A constant value (e.g., 0)
 - The mean of the column
 - Print the modified DataFrame in each case.
 7. Interpolation of Missing Data:
 - ✓ Explain the purpose of the interpolate() method in Pandas. Write Python code to apply interpolation to fill missing values in a DataFrame.
 8. Dropping Rows with Missing Values:

Create a DataFrame with missing values. Write Python code to:

a. Drop rows where any value is missing.

9. Handling Duplicate Rows:

Using the following DataFrame:

```
DF_obj = pd.DataFrame({
    'column 1': [1, 1, 2, 2, 3, 3, 3],
    'column 2': ['a', 'a', 'b', 'b', 'c', 'c', 'c'],
    'column 3': ['A', 'A', 'B', 'B', 'C', 'C', 'C']
})
```

a. Check for duplicate rows using the duplicated() method.

b. Remove rows with duplicate values in column 3 using the drop_duplicates() method.

10. Mean Imputation for Missing Values:

- ✓ Create a DataFrame with missing values in the age column. Calculate the mean of the column and replace the missing values with this mean. Print the updated DataFrame.

11. Concatenation of DataFrames

a. Create two DataFrames:

- ✓ DF_1 with values ranging from 0 to 35, reshaped into a 6x6 matrix.
- ✓ DF_2 with values ranging from 0 to 14, reshaped into a 5x3 matrix.

b. Concatenate the DataFrames:

- i. Along columns.
- ii. Along rows.
- iii. Along rows without repeating index

12. Drop Operations

Using the DF_1 DataFrame:

a. Drop rows with indices 0 and 2.

b. Drop columns with indices 0 and 2.

13. GroupBy with Cars Dataset

a. Load the cars.csv dataset with columns:

Cars_name, mpg, cyl, disp, hp, drat, wt, qsec, vs, am, gear, carb.

b. Group the data by the column cyl.

- i. Calculate the mean of all numeric columns for each group.

14. Working with Series

a. Create a Series with values ranging from 0 to 5 and name it "Series column".

b. Add this Series as a new column to DF_1.