**Pandas Questions**

1. How do you read a CSV file and display the first 10 rows?

import pandas as pd

df = pd.read\_csv("data n1.csv")

print(df)

print(df.head(10))

o/p:

Name Age Gender Salary Department emp\_id Sales Date

0 Abhi 21 Female NaN HR 1 5000 1-5-25

1 Ananya 25 Female 50000.0 Sales 2 6000 20-7-25

2 Vibha 30 Female 40000.0 Sales 3 7000 4-6-25

3 Vibha 35 Female 30000.0 IT 4 8000 5-7-25

4 bhumika 32 Female 35000.0 IT 5 4000 20-5-25

5 esha 23 Female 30000.0 IT 6 3000 10-5-25

6 karthik 28 Male NaN Sales 7 6000 15-7-25

7 Bhargav 27 Male 70000.0 HR 8 5000 12-6-25

8 Harini 26 Female NaN HR 9 8000 11-6-25

9 Hemanth 25 Male 45000.0 Sales 10 9000 12-6-25

10 Disha 22 Female 55000.0 HR 11 4500 25-5-25

11 rahul 31 Male 65000.0 Trainee 12 3500 20-6-25

12 Rohit 27 Male 20000.0 Trainee 13 5000 30-5-25

13 Kavya 25 Female 40000.0 Sales 14 6000 1-5-25

14 Likitha 24 Female 90000.0 IT 15 7000 4-5-25

first 10 rows:

Name Age Gender Salary Department emp\_id Sales Date

0 Abhi 21 Female NaN HR 1 5000 1-5-25

1 Ananya 25 Female 50000.0 Sales 2 6000 20-7-25

2 Vibha 30 Female 40000.0 Sales 3 7000 4-6-25

3 Vibha 35 Female 30000.0 IT 4 8000 5-7-25

4 bhumika 32 Female 35000.0 IT 5 4000 20-5-25

5 esha 23 Female 30000.0 IT 6 3000 10-5-25

6 karthik 28 Male NaN Sales 7 6000 15-7-25

7 Bhargav 27 Male 70000.0 HR 8 5000 12-6-25

8 Harini 26 Female NaN HR 9 8000 11-6-25

9 Hemanth 25 Male 45000.0 Sales 10 9000 12-6-25

1. How do you find the number of rows and columns in a DataFrame?

import pandas as pd

df = pd.read\_csv("data n1.csv")

print(df.shape)

o/p:

(15, 8)

1. Given a DataFrame df, how do you display only the rows where age > 25?

import pandas as pd

df = pd.read\_csv("data n1.csv")

print(df[df['Age'] > 25])

o/p:

Name Age Gender Salary Department emp\_id Sales Date

2 Vibha 30 Female 40000.0 Sales 3 7000 4-6-25

3 Vibha 35 Female 30000.0 IT 4 8000 5-7-25

4 bhumika 32 Female 35000.0 IT 5 4000 20-5-25

6 karthik 28 Male NaN Sales 7 6000 15-7-25

7 Bhargav 27 Male 70000.0 HR 8 5000 12-6-25

8 Harini 26 Female NaN HR 9 8000 11-6-25

11 rahul 31 Male 65000.0 Trainee 12 3500 20-6-25

12 Rohit 27 Male 20000.0 Trainee 13 5000 30-5-25

1. Replace all missing values in column salary with the **mean salary**.

import pandas as pd

df = pd.read\_csv("data n1.csv")

df['Salary'].fillna(df['Salary'].mean(), inplace=True)

print(df['Salary'])

o/p:

0 47500.0

1 50000.0

2 40000.0

3 30000.0

4 35000.0

5 30000.0

6 47500.0

7 70000.0

8 47500.0

9 45000.0

10 55000.0

11 65000.0

12 20000.0

13 40000.0

14 90000.0

Name: Salary, dtype: float64

1. How do you rename the column emp\_id to employee\_id?

import pandas as pd

df = pd.read\_csv("data n1.csv")

df.rename(columns={'emp\_id': 'employee\_id'}, inplace=True)

print(df.columns)

o/p:

Index(['Name', 'Age', 'Gender', 'Salary', 'Department', 'employee\_id', 'Sales',

'Date'],

dtype='object')

1. How to drop duplicate rows in a DataFrame?

import pandas as pd

df = pd.read\_csv("data n1.csv")

df.drop\_duplicates(inplace=True)

print(df)

o/p:

Name Age Gender Salary Department employee\_id Sales Date

0 Abhi 21 Female 47500.0 HR 1 5000 1-5-25

1 Ananya 25 Female 50000.0 Sales 2 6000 20-7-25

2 Vibha 30 Female 40000.0 Sales 3 7000 4-6-25

3 bhumika 32 Female 35000.0 IT 5 4000 20-5-25

4 esha 23 Female 30000.0 IT 6 3000 10-5-25

5 karthik 28 Male 47500.0 Sales 7 6000 15-7-25

6 Bhargav 27 Male 70000.0 HR 8 5000 12-6-25

7 Harini 26 Female 47500.0 HR 9 8000 11-6-25

8 Hemanth 25 Male 45000.0 Sales 10 9000 12-6-25

9 Disha 22 Female 55000.0 HR 11 4500 25-5-25

10 rahul 31 Male 65000.0 Trainee 12 3500 20-6-25

11 Rohit 27 Male 20000.0 Trainee 13 5000 30-5-25

12 Kavya 25 Female 40000.0 Sales 14 6000 1-5-25

13 Likitha 24 Female 90000.0 IT 15 7000 4-5-25

1. How do you convert a column of string dates "2023-05-01" into datetime format?

import pandas as pd

df = pd.read\_csv("data n1.csv")

df['Date'] = pd.to\_datetime(df['Date'])

print(df['Date'].dtype)

1. How to group a DataFrame by department and find the total salary for each?

import pandas as pd

df = pd.read\_csv("data n1.csv")

total\_salary=df.groupby('Department')['Salary'].sum()

print(total\_salary)

o/p:

Department

HR 220000.0

IT 185000.0

Sales 222500.0

Trainee 85000.0

Name: Salary, dtype: float64

1. How do you sort a DataFrame by date in descending order?

import pandas as pd

df = pd.read\_csv("data n1.csv")

sort=df.sort\_values(by='Date', ascending=False)

print(sort)

o/p:

Name Age Gender Salary Department employee\_id Sales Date

7 Bhargav 27 Male 70000.0 HR 8 5000 2025-12-06

9 Hemanth 25 Male 45000.0 Sales 10 9000 2025-12-06

8 Harini 26 Female 47500.0 HR 9 8000 2025-11-06

5 esha 23 Female 30000.0 IT 6 3000 2025-10-05

1 Ananya 25 Female 50000.0 Sales 2 6000 2025-07-20

6 karthik 28 Male 47500.0 Sales 7 6000 2025-07-15

11 rahul 31 Male 65000.0 Trainee 12 3500 2025-06-20

12 Rohit 27 Male 20000.0 Trainee 13 5000 2025-05-30

10 Disha 22 Female 55000.0 HR 11 4500 2025-05-25

4 bhumika 32 Female 35000.0 IT 5 4000 2025-05-20

3 Vibha 35 Female 30000.0 IT 4 8000 2025-05-07

2 Vibha 30 Female 40000.0 Sales 3 7000 2025-04-06

14 Likitha 24 Female 90000.0 IT 15 7000 2025-04-05

0 Abhi 21 Female 47500.0 HR 1 5000 2025-01-05

13 Kavya 25 Female 40000.0 Sales 14 6000 2025-01-05

1. Create a new column total\_marks by adding three columns: math, science, and english.

import pandas as pd

df = pd.read\_csv("data n2.csv")

print(df)

df['total\_marks'] = df['Maths'] + df['Science'] + df['English']

print(df[['Name', 'total\_marks']])

o/p:

Name Score Maths Science English

0 Abhi 80 70 78 92

1 Ananya 90 75 90 93

2 Vibha 77 80 58 77

3 Vibha 65 90 97 66

4 bhumika 72 67 69 84

5 esha 74 50 62 95

6 karthik 70 96 60 72

7 Bhargav 80 72 83 74

8 Harini 94 65 80 79

9 Hemanth 91 82 88 61

10 Disha 83 94 77 94

11 rahul 79 85 92 91

12 Rohit 60 70 76 88

13 Kavya 65 72 70 84

14 Likitha 69 79 91 81

Name total\_marks

0 Abhi 240

1 Ananya 258

2 Vibha 215

3 Vibha 253

4 bhumika 220

5 esha 207

6 karthik 228

7 Bhargav 229

8 Harini 224

9 Hemanth 231

10 Disha 265

11 rahul 268

12 Rohit 234

13 Kavya 226

14 Likitha 251

1. How do you filter rows where column score is between 40 and 80?

import pandas as pd

df = pd.read\_csv("data n2.csv")

filter=df[(df['Score'] >= 40) & (df['Score'] <= 80)]

print(filter)

o/p:

Name Score Maths Science English

0 Abhi 80 70 78 92

2 Vibha 77 80 58 77

3 Vibha 65 90 97 66

4 bhumika 72 67 69 84

5 esha 74 50 62 95

6 karthik 70 96 60 72

7 Bhargav 80 72 83 74

11 rahul 79 85 92 91

12 Rohit 60 70 76 88

13 Kavya 65 72 70 84

14 Likitha 69 79 91 81

1. Display the top 3 rows with the highest values in column sales.

import pandas as pd

df = pd.read\_csv("data n1.csv")

Highest=df.nlargest(3, 'Sales')[['Name', 'Sales']]

print(Highest)

o/p:

Name Sales

Date

2025-12-06 Hemanth 9000

2025-05-07 Vibha 8000

2025-11-06 Harini 8000

1. Find the average, min, and max of the price column using a single function.

import pandas as pd

df = pd.read\_csv("data n1.csv")

describe=df[['Age', 'Salary']].describe()

print(describe)

mean\_values = describe.loc['mean']

min\_values = describe.loc['min']

max\_values = describe.loc['max']

print("Mean:\n", mean\_values)

print("\nMin:\n", min\_values)

print("\nMax:\n", max\_values)

o/p:

Age Salary

count 15.000000 15.000000

mean 26.733333 47500.000000

std 3.899939 17576.363999

min 21.000000 20000.000000

25% 24.500000 37500.000000

50% 26.000000 47500.000000

75% 29.000000 52500.000000

max 35.000000 90000.000000

Mean:

Age 26.733333

Salary 47500.000000

Name: mean, dtype: float64

Min:

Age 21.0

Salary 20000.0

Name: min, dtype: float64

Max:

Age 35.0

Salary 90000.0

Name: max, dtype: float64

1. How to set a column (e.g., date) as the index of a DataFrame?

import pandas as pd

df = pd.read\_csv("data n1.csv")

df.set\_index('Date', inplace=True)

print(df.head(2))

o/p:

Name Age Gender Salary Department employee\_id Sales

Date

2025-01-05 Abhi 21 0 47500.0 HR 1 5000

2025-07-20 Ananya 25 0 50000.0 Sales 2 6000

2025-04-06 Vibha 30 0 40000.0 Sales 3 7000

2025-05-07 Vibha 35 0 30000.0 IT 4 8000

2025-05-20 bhumika 32 0 35000.0 IT 5 4000

2025-10-05 esha 23 0 30000.0 IT 6 3000

2025-07-15 karthik 28 1 47500.0 Sales 7 6000

2025-12-06 Bhargav 27 1 70000.0 HR 8 5000

2025-11-06 Harini 26 0 47500.0 HR 9 8000

2025-12-06 Hemanth 25 1 45000.0 Sales 10 9000

2025-05-25 Disha 22 0 55000.0 HR 11 4500

2025-06-20 rahul 31 1 65000.0 Trainee 12 3500

2025-05-30 Rohit 27 1 20000.0 Trainee 13 5000

2025-01-05 Kavya 25 0 40000.0 Sales 14 6000

2025-04-05 Likitha 24 0 90000.0 IT 15 7000

1. Convert a categorical column gender (with values M/F) into numeric using mapping (M → 1, F → 0).

import pandas as pd

df = pd.read\_csv("data n1.csv")

df['Gender'] = df['Gender'].map({'Male': 1, 'Female': 0})

print(df[['Name', 'Gender']])

o/p:

Name Gender

Date

2025-01-05 Abhi 0

2025-07-20 Ananya 0

2025-04-06 Vibha 0

2025-05-07 Vibha 0

2025-05-20 bhumika 0

2025-10-05 esha 0

2025-07-15 karthik 1

2025-12-06 Bhargav 1

2025-11-06 Harini 0

2025-12-06 Hemanth 1

2025-05-25 Disha 0

2025-06-20 rahul 1

2025-05-30 Rohit 1

2025-01-05 Kavya 0

2025-04-05 Likitha 0

**NumPy Questions**

1. Create a 1D NumPy array from 0 to 9.

import numpy as np

arr = np.arange(10)

print("Array :\n", arr)

o/p:

Array :

[0 1 2 3 4 5 6 7 8 9]

1. Create a 3x3 NumPy array filled with random integers between 1 and 100.

random\_arr=np.random.randint(1,101,size=(3,3))

print("Random Array :\n", random\_arr)

o/p:

Random Array :

[[71 27 55]

[29 84 92]

[29 21 19]]

1. How do you find the mean and standard deviation of a NumPy array?

arr = np.array([10, 20, 30, 40, 50])

mean = np.mean(arr)

std = np.std(arr)

print("Mean:", mean)

print("Standard Deviation:", std)

o/p:

Mean: 30.0

Standard Deviation: 14.142135623730951

1. Given an array, replace all values greater than 50 with 50.

arr=np.array([10,53,79,40,60,85,35])

arr[arr>50]=50

print("Array :\n", arr)

o/p:

Array :

[10 50 50 40 50 50 35]

1. Create a NumPy array and reshape it from 1D to 2D (e.g., 12 elements → 3x4).

arr=np.arange(12)

reshape=arr.reshape(3,4)

print("reshaped array :\n", reshape)

o/p:

reshaped array :

[[ 0 1 2 3]

[ 4 5 6 7]

[ 8 9 10 11]]

1. Find the number of even numbers in a NumPy array.

arr=np.array([8,23,12,55,48,80,9])

even=np.sum(arr%2==0)

print("no of even no.s in the array :\n", even)

o/p:

no of even no.s in the array :

4

1. How do you flatten a 2D NumPy array to 1D?

arr = np.array([[1, 2, 3], [4, 5, 6]])

flattened = arr.flatten()# converts to 1d array

print("\nFlattened array:\n", flattened)

o/p:Flattened array:

[1 2 3 4 5 6]

1. Create an array of 10 random floats between 0 and 1.

rand\_floats=np.random.rand(10)

print("Random Array :\n", rand\_floats)

o/p:

Random Array :

[0.31910597 0.60428839 0.78108632 0.68851577 0.27746419 0.80993041

0.10923391 0.16566654 0.92441503 0.34218882]

1. Multiply two NumPy arrays element-wise.

arr1=np.array([1,2,3])

arr2=np.array([7,8,9])

product = arr1\*arr2

print("product :\n", product)

o/p:

product :

[ 7 16 27]

1. Given a NumPy array, how do you find the index of the maximum value?

arr=np.array([10,25,60,55,35,70])

max=np.argmax(arr)

print("index of max array :\n", max)

o/p: index of max array :

5