

Q.1 How do you create a Pandas DataFrame?

Ans1

```
import pandas as pd
```

```
data = {'Name': ['Alice', 'Bob', 'Charlie'],  
        'Age': [25, 30, 35],  
        'City': ['New York', 'Los Angeles', 'Chicago']}  
df = pd.DataFrame(data)  
print(df)
```

Q.2 How do you read a CSV file into a Pandas DataFrame?

Ans.2

```
df = pd.read_csv('data.csv')  
print(df.head())
```

Q.3 How do you check for missing values in a DataFrame?

Ans.3

```
print(df.isnull().sum())
```

Q.4 How do you drop rows with missing values?

Ans.4

```
df_cleaned = df.dropna()  
print(df_cleaned)
```

Q.5 How do you fill missing values with a specific value?

Ans.5

```
df_filled = df.fillna(0)  
print(df_filled)
```

Q.6 How do you filter rows based on a condition?

Ans.6

```
filtered_df = df[df['Age'] > 30]
print(filtered_df)
```

Q.7 How do you select a specific column in a DataFrame?

Ans.7

```
ages = df['Age']
print(ages)
```

Q.8 How do you rename columns in a DataFrame?

Ans.8

```
df.rename(columns={'Name': 'Full Name', 'Age': 'Years'}, inplace=True)
print(df)
```

Q.9 How do you sort a DataFrame by a specific column?

Ans.9

```
sorted_df = df.sort_values(by='Age')
print(sorted_df)
```

Q.10 How do you group data and calculate aggregate statistics?

Ans.10

```
grouped = df.groupby('City').mean()
print(grouped)
```

Q.11 How do you reset the index of a DataFrame?

Ans.11

```
df.reset_index(drop=True, inplace=True)
print(df)
```

Q.12 How do you merge two DataFrames?

Ans.12

```
df1 = pd.DataFrame({'ID': [1, 2], 'Name': ['sam', 'raj']})
df2 = pd.DataFrame({'ID': [1, 2], 'Age': [25, 30]})
merged_df = pd.merge(df1, df2, on='ID')
print(merged_df)
```

Q.13 How do you concatenate two DataFrames?

Ans.13

```
df1 = pd.DataFrame({'A': [1, 2]})
df2 = pd.DataFrame({'A': [3, 4]})
concatenated = pd.concat([df1, df2])
print(concatenated)
```

Q.14 How do you pivot a DataFrame?

Ans.14

```
df = pd.DataFrame({'Name': ['Alice', 'Bob'], 'Year': [2020, 2021], 'Score': [85, 90]})
pivoted = df.pivot(index='Name', columns='Year', values='Score')
print(pivoted)
```

Q.15 How do you melt a DataFrame?

Ans.15

```
melted = pd.melt(df, id_vars=['Name'], value_vars=['Year', 'Score'])
print(melted)
```

Q.16 How do you calculate the rolling mean of a column?

Ans.16

```
df['Rolling Mean'] = df['Age'].rolling(window=2).mean()
print(df)
```

Q.17 How do you find the unique values in a column?

Ans.17

```
unique_cities = df['City'].unique()
```

```
print(unique_cities)
```

Q.18 How do you count the occurrences of unique values in a column?

Ans.18

```
value_counts = df['City'].value_counts()  
print(value_counts)
```

Q.19 How do you apply a custom function to a column?

Ans.19

```
df['Age Squared'] = df['Age'].apply(lambda x: x**2)  
print(df)
```

Q.20 How do you drop duplicate rows in a DataFrame?

Ans.20

```
df = df.drop_duplicates()  
print(df)
```

Q.21 How do you add a new column to a DataFrame?

Ans.21

```
df['Country'] = 'USA'  
print(df)
```

Q.22 How do you remove a column from a DataFrame?

Ans.22

```
df = df.drop(columns=['Country'])  
print(df)
```

Q.23 How do you filter rows based on multiple conditions?

Ans.23

```
filtered = df[(df['Age'] > 25) & (df['City'] == 'New York')]  
print(filtered)
```

Q.24 How do you create a DataFrame from a NumPy array?

Ans.24

```
import numpy as np
arr = np.array([[1, 2], [3, 4]])
df = pd.DataFrame(arr, columns=['A', 'B'])
print(df)
```

Q.25 How do you convert a column's data type?

Ans.25

```
df['Age'] = df['Age'].astype(float)
print(df.dtypes)
```

Q.26 **How do you create a histogram of a column?**

Ans.26

```
df['Age'].plot(kind='hist')
```

Q.27 How do you export a DataFrame to a CSV file?

Ans.27

```
df.to_csv('output.csv', index=False)
```

Q.28 How do you create a DataFrame from a dictionary of lists?

Ans.28

```
data = {'A': [1, 2], 'B': [3, 4]}
df = pd.DataFrame(data)
print(df)
```

Q.29 **How do you find the correlation between columns?**

Ans.29

```
correlation = df.corr()
print(correlation)
```

Q.30 How do you convert a DataFrame to a NumPy array?

Ans.30

```
arr = df.to_numpy()
print(arr)
```

Q.31 Find the average salary for each department in a company.

Ans.31

```
import pandas as pd
```

```
data = {'Department': ['HR', 'IT', 'HR', 'IT', 'Finance'],
        'Salary': [40000, 50000, 45000, 55000, 60000]}
df = pd.DataFrame(data)
```

```
result = df.groupby('Department')['Salary'].mean()
print(result)
```

Q.32 Count the number of employees in each department.

Ans32

```
result = df.groupby('Department')['Salary'].count()
print(result)
```

Q.33 Calculate the total salary paid in each department.

Ans.33

```
result = df.groupby('Department')['Salary'].sum()
print(result)
```

Q.34 Find the mean and maximum salary for each department.

A.s34

```
result = df.groupby('Department')['Salary'].agg(['mean', 'max'])
print(result)
```

Q.35 Find the total salary for each combination of department and job role.

Ans.35

```
data = {'Department': ['HR', 'IT', 'HR', 'IT', 'Finance'],  
       'Role': ['Manager', 'Developer', 'Assistant', 'Developer', 'Analyst'],  
       'Salary': [40000, 50000, 45000, 55000, 60000]}  
df = pd.DataFrame(data)
```

```
result = df.groupby(['Department', 'Role'])['Salary'].sum()  
print(result)
```

Q.36 Retrieve the first employee record from each department.

Ans.36

```
result = df.groupby('Department').first()  
print(result)
```

Q.37 Find departments where the total salary is greater than 90,000.

Ans.37

```
result = df.groupby('Department')['Salary'].sum()  
filtered_result = result[result > 90000]  
print(filtered_result)
```

Q.38 Add a column that shows the cumulative salary in each department.

Ans.38

```
df['Cumulative_Salary'] = df.groupby('Department')['Salary'].cumsum()  
print(df)
```

Q.39 Normalize salaries within each department by subtracting the group mean.

Ans.39

```
df['Normalized_Salary'] = df.groupby('Department')['Salary'].transform(lambda x: x - x.mean())  
print(df)
```

Q.40 Find out how many employees are there in each department.

Ans.40

```
result = df.groupby('Department').size()  
print(result)
```