

## 1. Creation and Basic Viewing

```
In [5]: import pandas as pd
data = {
    'Name': ['Alice', 'Bob', 'Charlie', 'David'],
    'Age': [24, 27, 22, 32],
    'City': ['New York', 'Los Angeles', 'Chicago', 'Houston']
}
df = pd.DataFrame(data)
print(df.head(2))
print(df.columns.tolist())
print(df.shape)
print(df.info())
```

```
   Name  Age    City
0  Alice   24  New York
1   Bob   27 Los Angeles
['Name', 'Age', 'City']
(4, 3)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4 entries, 0 to 3
Data columns (total 3 columns):
 #   Column  Non-Null Count  Dtype
---  ---
 0   Name    4 non-null         object
 1   Age     4 non-null         int64
 2   City    4 non-null         object
dtypes: int64(1), object(2)
memory usage: 224.0+ bytes
None
```

## 2. Selection & Indexing

```
In [7]: print(df['Name'])
print(df[['Name', 'City']])
print(df.iloc[1])
print(df.loc[df['Name'] == 'Charlie'])
```

```
0      Alice
1       Bob
2    Charlie
3     David
Name: Name, dtype: object
      Name      City
0    Alice  New York
1     Bob  Los Angeles
2  Charlie   Chicago
3    David   Houston
Name      Bob
Age       27
City  Los Angeles
Name: 1, dtype: object
      Name  Age      City
2  Charlie  22  Chicago
```

### 3.Filtering / Conditions

```
In [21]: print(df[df['Age']>25],"\n")
print(df[(df['City'] == 'Chicago') | (df['City'] == 'Houston')],"\n")
print(df[(df['Age'] >= 23) & (df['Age'] <= 30)],"\n")
```

```
      Name  Age      City
1     Bob   27  Los Angeles
3  David   32    Houston
```

```
      Name  Age      City
2  Charlie  22  Chicago
3    David  32  Houston
```

```
      Name  Age      City
0  Alice   24    New York
1     Bob   27  Los Angeles
```

### 4.Modify the DataFrame:

```
In [33]: import pandas as pd
data = {
    'Name': ['Alice', 'Bob', 'Charlie', 'David'],
    'Age': [24, 27, 22, 32],
    'City': ['New York', 'Los Angeles', 'Chicago', 'Houston']
}

df = pd.DataFrame(data)
print(df, "\n")
df['Score'] = [85, 90, 88, 95]
print(df, "\n")
df.loc[df['Name'] == 'Bob', 'Age'] = 28
print(df, "\n")
df.drop('City', axis=1, inplace=True)
print(df, "\n")
df.drop(df[df['Name'] == 'David'].index, inplace=True)
print(df)
```

	Name	Age	City
0	Alice	24	New York
1	Bob	27	Los Angeles
2	Charlie	22	Chicago
3	David	32	Houston

	Name	Age	City	Score
0	Alice	24	New York	85
1	Bob	27	Los Angeles	90
2	Charlie	22	Chicago	88
3	David	32	Houston	95

	Name	Age	City	Score
0	Alice	24	New York	85
1	Bob	28	Los Angeles	90
2	Charlie	22	Chicago	88
3	David	32	Houston	95

	Name	Age	Score
0	Alice	24	85
1	Bob	28	90
2	Charlie	22	88
3	David	32	95

	Name	Age	Score
0	Alice	24	85
1	Bob	28	90
2	Charlie	22	88

5. Create a new DataFrame:

```
In [34]: data2 = {
    'Department': ['HR', 'IT', 'HR', 'IT'],
    'Salary': [30000, 50000, 35000, 55000],
    'Experience': [2, 5, 3, 6]
}

df2 = pd.DataFrame(data2)

print(df2.groupby('Department')['Salary'].mean())
print(df2.groupby('Department')['Experience'].max())
print(df2['Salary'].sum())
```

```
Department
HR      32500.0
IT      52500.0
Name: Salary, dtype: float64
Department
HR       3
IT       6
Name: Experience, dtype: int64
170000
```

6. Given a DataFrame with missing values:

```
In [36]: data3 = {
    'Student': ['John', 'Emma', 'Sam', 'Olivia'],
    'Marks': [80, None, 75, 90]
}

df3 = pd.DataFrame(data3)

print(df3.fillna(0))
print(df3.dropna())
print(df3.sort_values(by='Marks', ascending=False))
```

```
Student Marks
0  John   80.0
1  Emma    0.0
2   Sam   75.0
3 Olivia  90.0
Student Marks
0  John   80.0
2   Sam   75.0
3 Olivia  90.0
Student Marks
3 Olivia  90.0
0  John   80.0
2   Sam   75.0
1  Emma   NaN
```

## 7. File I/O and Indexing

```
In [39]: import pandas as pd
df_employees = pd.DataFrame({
    'Name': ['Ravi', 'Priya', 'Karan'],
    'Age': [29, 35, 42],
    'Department': ['IT', 'HR', 'Finance']
})
df_employees.to_csv('employees.csv', index=False)
df_employees = pd.read_csv('employees.csv')
df_employees.set_index('Name', inplace=True)
df_employees.reset_index(inplace=True)
print(df_employees)
```

	Name	Age	Department
0	Ravi	29	IT
1	Priya	35	HR
2	Karan	42	Finance

8. Write a line of code using `.loc[]` to display the details of the first and third products in the DataFrame.

```
In [40]: data4 = {
    'Product': ['Laptop', 'Tablet', 'Smartphone', 'Monitor', 'Keyboard'],
    'Price': [70000, 30000, 25000, 15000, 2000],
    'Stock': [10, 25, 50, 15, 100]
}
df4 = pd.DataFrame(data4)
print(df4.loc[[0, 2]])
```

	Product	Price	Stock
0	Laptop	70000	10
2	Smartphone	25000	50

9. Create a DataFrame from the above data and assign custom row labels: 'Student1', 'Student2', and 'Student3' using the index argument. Then, using `.loc[]`, print the marks obtained by 'Student2'.

```
In [41]: data5 = {
    'Subject': ['Math', 'Science', 'English'],
    'Marks': [88, 92, 85]
}
df5 = pd.DataFrame(data5, index=['Student1', 'Student2', 'Student3'])
print(df5.loc['Student2'])
```

	Subject	Science
Marks		92

Name: Student2, dtype: object

