

Pandas is a powerful Python library primarily used for data analysis and data manipulation. It's built on top of **NumPy** and provides two primary data structures:

- **Series** – 1D labeled array
 - **DataFrame** – 2D labeled table of data (like an Excel sheet or SQL table)
-

Common Uses of Pandas in Data Analysis

1. Loading Data

import pandas as pd

```
df = pd.read_csv('data.csv')    # Read CSV
df = pd.read_excel('data.xlsx')  # Read Excel
df = pd.read_json('data.json')  # Read JSON
```

2. Exploring Data

```
df.head()      # First 5 rows
df.tail()      # Last 5 rows
df.shape       # (rows, columns)
df.info()      # Summary of DataFrame
df.describe()  # Stats summary for numeric columns
```

3. Selecting Data

```
df['column_name']    # Select a column
df[['col1', 'col2']] # Select multiple columns
df.iloc[0]           # Select first row by index
df.loc[0, 'col_name'] # Select specific cell
```

4. Filtering / Querying

```
df[df['age'] > 30]          # Rows where age > 30
df[(df['age'] > 30) & (df['city'] == 'NY')] # Multiple conditions
```

5. Data Cleaning

```
df.isnull().sum()    # Count missing values
df.dropna()          # Drop rows with NaN
```

```
df.fillna(0) # Replace NaN with 0
df['col'] = df['col'].astype(int) # Convert data type
```

6. Aggregation & Grouping

```
df.groupby('city')['salary'].mean() # Avg salary per city
df['salary'].sum() # Total salary
```

7. Sorting

```
df.sort_values(by='age') # Sort by age
df.sort_values(by='age', ascending=False) # Descending
```

8. Merging & Joining

```
pd.merge(df1, df2, on='id', how='inner') # SQL-style join
df1.append(df2) # Stack vertically
pd.concat([df1, df2], axis=1) # Stack horizontally
```

9. Saving Data

```
df.to_csv('output.csv', index=False) # Save to CSV
df.to_excel('output.xlsx') # Save to Excel
```

Example Use Case

```
# Analyze average sales by region
df = pd.read_csv('sales_data.csv')
summary = df.groupby('region')['sales'].mean()
print(summary)
```
