Introduction to Pandas

Fundamentals of Data Handling with Pandas

Muhammad Jamil

What is Pandas?

- Pandas is a powerful data analysis and manipulation library for Python.
- Used for data cleaning, transformation, and analysis.
- **Why use Pandas?**
  - Easy to handle structured data (like tables).
  - Provides data manipulation tools (filtering, grouping, merging).

Installing Pandas

- Activate Virtual Environment (Best Practice).
- Installation Command:

```
pip install pandas
```

- The terminal will start the installation.
- Once the installation is finished, use the testing code to test the functionality of the library.

Pandas Data Structures

- 1. Series
  - One-dimensional labeled array.
  - Similar to a list but with labels (index).

```
1  import pandas as pd
2  series = pd.Series([10, 20, 30], index=['a', 'b', 'c'])
3  print(series)
4
```

Pandas Data Structures

- 2. DataFrame
  - Two-dimensional, like a table with rows and columns.

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

```

Reading Data

- Reading CSV Files

```
1  df = pd.read_csv('data.csv')
2  print(df.head())  # Shows first 5 rows
```

- Reading Excel Files

```
1  df = pd.read_excel('data.xlsx')
```

Data Exploration

- Basic Info and Summary

```
print(df.info())  # Summary of data
print(df.describe())  # Statistical summary
```

- Viewing Data

```
print(df.head())  # First 5 rows
print(df.tail())  # Last 5 rows
```

Selecting Data

- Selecting Columns
- Selecting Rows
- Filtering Data

```
1  print(df['Name'])
```

```
1  print(df.iloc[0])  # First row
```

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

Data Cleaning

- Handling Missing Data

```
df = df.fillna(0)  # Replace missing values with 0
df = df.dropna()  # Remove rows with missing values
```

- Renaming Columns

```
df = df.rename(columns={'Name': 'Full Name'})
```

Data Manipulation

- Sorting Data

```
df = df.sort_values('Age', ascending=False)
```

- Adding a New Column

```
df['Salary'] = [50000, 60000, 70000]
```

Grouping and Aggregating Data

- Grouping by a Column

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

- Aggregating Functions

```
print(df['Age'].sum())  # Sum of ages
print(df['Age'].mean())  # Average age
```

Merging DataFrames

- Combining Two DataFrames

```
1  df1 = pd.DataFrame({'ID': [1, 2], 'Name': ['Alice', 'Bob']})
2  df2 = pd.DataFrame({'ID': [1, 2], 'Score': [85, 90]})
3
4  merged_df = pd.merge(df1, df2, on='ID')
5  print(merged_df)
6
```

Exporting Data

- Saving to CSV

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

- Saving to Excel

```
1  df.to_excel('output.xlsx', index=False)
```

Practice Task

- **Task**
  - Read a CSV file of student data (names, grades).
  - Filter out students with grades below 50.
  - Add a new column called Passed (True/False based on grade).
  - Export the updated data to a new CSV file.

Thank You!

jamil@cuisahiwal.edu.pk