Data Analysis Techniques: Filtering, Sorting, and Aggregating Data (1.5 hours)

Filtering Data

- **Basic Filtering**: To filter data based on conditions, you can use boolean indexing.
- ```
  # Get rows where Age is greater than 30
  df_filtered = df[df['Age'] > 30]
  ```

- **Multiple Conditions**: You can combine multiple conditions using logical operators:
- ```
  # Get rows where Age is greater than 30 and City is 'New York'
  df_filtered = df[(df['Age'] > 30) & (df['City'] == 'New York')]
  ```

- **Sorting Data**
  - **Sorting by Columns**: You can sort the DataFrame by a column in ascending or descending order:

    ```
    df_sorted = df.sort_values(by='Age', ascending=False)  # Sort by Age in descending order
    ```

    Sorting by Multiple Columns:

    ```
    df_sorted = df.sort_values(by=['Age', 'City'], ascending=[True, False])
    ```

Aggregating Data

- **Groupby**: pandas `groupby` method is essential for aggregating data based on one or more columns. For example:

```
grouped = df.groupby('City').agg({'Age': 'mean', 'Name': 'count'})
print(grouped)
```

- **Summary Statistics**: pandas provide several built-in functions to calculate summary statistics:

```
df['Age'].mean()  # Mean of 'Age' column
df['Age'].sum()   # Sum of 'Age' column
df['Age'].max()   # Maximum of 'Age' column
```