

```

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("DataFrame created from dictionary:\n", df)
print("\nFirst 5 rows of the DataFrame:\n", df.head())
print("\nLast 5 rows of the DataFrame:\n", df.tail())
sample_size = min(5, len(df))
print(f"\nRandom {sample_size} rows of the DataFrame:\n",
df.sample(sample_size))
print("\nSummary statistics for numerical columns:\n", df.describe())
print("\nInformation about the DataFrame:\n", df.info())
print("\nNumber of missing values in each column:\n",
df.isnull().sum())
if df['Age'].isnull().sum() > 0:
    df['Age'] = df['Age'].fillna(df['Age'].mean())
print("\nNumber of missing values after handling in 'Age' column:\n",
df.isnull().sum())
print("\nNumber of duplicate rows before removal:",
df.duplicated().sum())
df = df.drop_duplicates()
print("Number of duplicate rows after removal:",
df.duplicated().sum())
filtered_df = df[df['Age'] > 25]
print("\nFiltered DataFrame (Age > 25):\n", filtered_df)
sorted_df = df.sort_values(by='Age')
print("\nSorted DataFrame by 'Age':\n", sorted_df)
grouped_df = df.groupby('City')['Age'].mean()
print("\nGrouped DataFrame by 'City' with mean 'Age':\n", grouped_df)

```

DataFrame created from dictionary:

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

First 5 rows of the DataFrame:

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

Last 5 rows of the DataFrame:

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

Random 4 rows of the DataFrame:

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

Summary statistics for numerical columns:

Age

count	4.000000
mean	26.250000
std	4.349329
min	22.000000
25%	23.500000
50%	25.500000
75%	28.250000
max	32.000000

<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: 228.0+ bytes

Information about the DataFrame:

None

Number of missing values in each column:

Name	0
Age	0
City	0

dtype: int64

Number of missing values after handling in 'Age' column:

Name	0
Age	0
City	0

dtype: int64

Number of duplicate rows before removal: 0

Number of duplicate rows after removal: 0

Filtered DataFrame (Age > 25):

	Name	Age	City
1	Bob	27	Los Angeles

```
3 David 32 Houston
```

```
Sorted DataFrame by 'Age':
```

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

```
Grouped DataFrame by 'City' with mean 'Age':
```

City	
Chicago	22.0
Houston	32.0
Los Angeles	27.0
New York	24.0

Name: Age, dtype: float64

```
import pandas as pd
file_path = r'C:\Users\Aishu\OneDrive\Desktop\people-100.csv'
try:
    df = pd.read_csv(file_path)
    print("First 5 rows:\n", df.head())
    print("Summary info:\n", df.info())
    selected_columns = df[['column_name1', 'column_name2']]
    print("Selected columns:\n", selected_columns.head())
    value = 25 # Replace with the actual value you want to filter on
    filtered_df = df[df['column_name'] > value]
    print("Filtered rows:\n", filtered_df.head())
    sorted_df = df.sort_values(by='column_name')
    print("Sorted DataFrame:\n", sorted_df.head())
    print("Missing values:\n", df.isnull().sum())
    df['column_name'] =
df['column_name'].fillna(df['column_name'].mean())
    df = df.drop_duplicates()
    df.to_csv('cleaned_data.csv', index=False)
except FileNotFoundError:
    print(f"The file '{file_path}' was not found. Please check the
file path and try again.")
except Exception as e:
    print(f"An error occurred: {e}")
```

```
First 5 rows:
```

	Index	User Id	First Name	Last Name	Sex	\
0	1	88F7B33d2bcf9f5	Shelby	Terrell	Male	
1	2	f90cD3E76f1A9b9	Phillip	Summers	Female	
2	3	DbeAb8CcdfeFC2c	Kristine	Travis	Male	
3	4	A31Bee3c201ef58	Yesenia	Martinez	Male	
4	5	1bA7A3dc874da3c	Lori	Todd	Male	

Email

Phone Date of birth \

```

0      elijah57@example.net    001-084-906-7849x73518    1945-10-26
1      bethany14@example.com    214.112.6044x4913    1910-03-24
2      bthompson@example.com    277.609.7938    1992-07-02
3      kaitlinkaiser@example.com    584.094.6111    2017-08-03
4      buchananmanuel@example.net    689-207-3558x7233    1938-12-01

```

```

      Job Title
0      Games developer
1      Phytotherapist
2      Homeopath
3      Market researcher
4      Veterinary surgeon

```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 100 entries, 0 to 99
```

```
Data columns (total 9 columns):
```

#	Column	Non-Null Count	Dtype
0	Index	100 non-null	int64
1	User Id	100 non-null	object
2	First Name	100 non-null	object
3	Last Name	100 non-null	object
4	Sex	100 non-null	object
5	Email	100 non-null	object
6	Phone	100 non-null	object
7	Date of birth	100 non-null	object
8	Job Title	100 non-null	object

```
dtypes: int64(1), object(8)
```

```
memory usage: 7.2+ KB
```

```
Summary info:
```

```
None
```

```
An error occurred: "None of [Index(['column_name1', 'column_name2'],
dtype='object')] are in the [columns]"
```

```
import pandas as pd
```

```
file_path = r'C:\Users\Aishu\OneDrive\Desktop\people-100.csv'
```

```
try:
```

```
    df = pd.read_csv(file_path)
```

```
    print("First 5 rows:\n", df.head())
```

```
    value = 25 # Replace with the actual value you want to filter on
```

```
    filtered_df = df[df['column_name'] > value]
```

```
    print("Filtered rows (column_name > value):\n",
```

```
filtered_df.head())
```

```
    sorted_df = df.sort_values(by='column_name')
```

```
    print("Sorted DataFrame by 'column_name':\n", sorted_df.head())
```

```
    grouped_df = df.groupby('group_column_name').mean()
```

```
    print("Grouped DataFrame by 'group_column_name' with mean values:\n", grouped_df)
```

```
    filtered_df.to_csv('filtered_data.csv', index=False)
```

```
    sorted_df.to_csv('sorted_data.csv', index=False)
```

```
    grouped_df.to_csv('grouped_data.csv', index=True)
```

```

except FileNotFoundError:
    print(f"The file '{file_path}' was not found. Please check the
file path and try again.")
except Exception as e:
    print(f"An error occurred: {e}")

```

First 5 rows:

	Index	User Id	First Name	Last Name	Sex	\
0	1	88F7B33d2bcf9f5	Shelby	Terrell	Male	
1	2	f90cD3E76f1A9b9	Phillip	Summers	Female	
2	3	DbeAb8CcdfcFC2c	Kristine	Travis	Male	
3	4	A31Bee3c201ef58	Yesenia	Martinez	Male	
4	5	1bA7A3dc874da3c	Lori	Todd	Male	

		Email	Phone	Date of birth	\
0		elijah57@example.net	001-084-906-7849x73518	1945-10-26	
1		bethany14@example.com	214.112.6044x4913	1910-03-24	
2		bthompson@example.com	277.609.7938	1992-07-02	
3		kaitlinkaiser@example.com	584.094.6111	2017-08-03	
4		buchananmanuel@example.net	689-207-3558x7233	1938-12-01	

	Job Title
0	Games developer
1	Phytotherapist
2	Homeopath
3	Market researcher
4	Veterinary surgeon

An error occurred: 'column\_name'