EXERCISE-5

import pandas as pd

Exercise 5: Handling Missing Values

1. Create a DataFrame with missing values:

Program:

```
data = {
    "Name": ["Amit", "Neha", "Raj", "Priya"],
    "Age": [28, None, 35, 29],
    "City": ["Delhi", "Mumbai", None, "Chennai"]
   }
df = pd.DataFrame(data)
```

2. Fill missing values in the `"Age"` column with the average age.

Program:

```
df['Age'] = df['Age'].fillna(df['Age'].mean())
print(df)
```

3. Drop rows where any column has missing data.

Program:

```
df = df.dropna()
print(df)
```

Exercise 6: Adding and Removing Columns

1. Add a new column `"Salary"` with the following values: `[50000, 60000, 70000, 65000]`.

Program:

df['Salary'] = [50000, 60000, 70000, 65000]

2. Remove the `"City"` column from the DataFrame.

Program:

df_dropped = df.drop(columns=["City"])

print(df_dropped)

Exercise 7: Sorting Data

1. Sort the DataFrame by `"Age"` in ascending order.

Program:

sorted_df = df.sort_values(by = "Age", ascending = True)

print(sorted_df)

2. Sort the DataFrame first by `"City"` and then by `"Age"` in descending order.

Program:

df = df.sort_values(by=["City", "Age"], ascending=[True, False])

print(sorted_df)

Exercise 8: Grouping and Aggregation

1. Group the DataFrame by `"City"` and calculate the average `"Age"` for each city.

Program:

d = df.groupby("City")["Age"].mean()

print(d)

2. Group the DataFrame by `"City"` and `"Age"`, and count the number of occurrences for each group.

Program:

df= df.groupby(['City', 'Age']).size()

print(df)

Exercise 9: Merging DataFrames

1. Create two DataFrames:A

Program:

```
df1 = pd.DataFrame({
    "Name": ["Amit", "Neha", "Raj"],
    "Department": ["HR", "IT", "Finance"]
})
df2 = pd.DataFrame({
    "Name": ["Neha", "Raj", "Priya"],
    "Salary": [60000, 70000, 65000]
})
```

2. Merge `df1` and `df2` on the `"Name"` column using an inner join.

Program:

```
df_inner = pd.merge(df1, df2, on="Name", how="inner")
print(df_inner)
```

3. Merge the same DataFrames using a left join.

Program:

```
df_left = pd.merge(df1, df2, on="Name", how="left")
print(df_left)
```