Darshan
UNIVERSITY

Python for Data Science - 2305CS303

Lab - 11

Roll No. : 111

Name : Dhara Maru

GroupBy

In [5]:

```python
import pandas as pd
```

In [6]:

```python
students = {
    'RollNo': [101, 102, 103, 104, 105, 106],
    'Name': ['Aarav', 'Diya', 'Ishaan', 'Meera', 'Kabir', 'Anaya'],
    'Dept': ['CSE', 'CSE', 'ECE', 'ECE', 'ME', 'CSE'],
    'Math': [88, 92, None, 74, 69, 85],
    'Science': [91, None, 78, 84, 76, 89],
    'English': [85, 87, 80, None, 74, 90]
}
```

1. Group students by Dept and find the average marks in each subject.

In [9]:

```python
df = pd.DataFrame(students)
```

In [10]:

```python
df_grp = df.groupby('Dept')

df_grp[['Math','Science','English']].mean()
```

Out[10]:

	Math	Science	English
Dept			
CSE	88.333333	90.0	87.333333
ECE	74.000000	81.0	80.000000
ME	69.000000	76.0	74.000000

2. Find the highest Math score in each department.

In [11]:

```
df_grp['Math'].max()
```

Out[11]:

```
Dept
CSE    92.0
ECE    74.0
ME     69.0
Name: Math, dtype: float64
```

3. Count how many students belong to each department.

In [12]:

```
df_grp['RollNo'].count()
```

Out[12]:

```
Dept
CSE    3
ECE    2
ME     1
Name: RollNo, dtype: int64
```

4. Compute the minimum, maximum, and mean of Science marks.

In [17]:

```
df['Science'].min()
```

Out[17]:

```
76.0
```

In [16]:

```
df['Science'].max()
```

Out[16]:

```
91.0
```

In [18]:

```
df['Science'].mean()
```

Out[18]:

```
83.6
```

5. For each department, apply multiple aggregations:

Math: mean, max

Science: min, count

In [19]:

```
df_grp['Math'].mean()
```

Out[19]:

```
Dept
CSE    88.333333
ECE    74.000000
ME     69.000000
Name: Math, dtype: float64
```

In [20]:

```
df_grp['Math'].max()
```

Out[20]:

```
Dept
CSE    92.0
ECE    74.0
ME     69.0
Name: Math, dtype: float64
```

In [21]:

```
df_grp['Science'].min()
```

Out[21]:

```
Dept
CSE    89.0
ECE    78.0
ME     76.0
Name: Science, dtype: float64
```

In [22]:

```
df_grp['Science'].count()
```

Out[22]:

```
Dept
CSE    2
ECE    2
ME     1
Name: Science, dtype: int64
```

Merge

In [23]:

```
attendance = {
    'RollNo': [101, 102, 103, 104, 107],
    'Attendance(%)': [92, 85, 88, 76, 90]
}
```

In [24]:

```
df1 = pd.DataFrame(students)
```

In [25]:

```
df2 = pd.DataFrame(attendance)
```

6. Merge students and attendance on RollNo (inner join).

In [26]:

```
pd.merge(df1, df2, on = 'RollNo', how = "inner")
```

Out[26]:

	RollNo	Name	Dept	Math	Science	English	Attendance(%)
0	101	Aarav	CSE	88.0	91.0	85.0	92
1	102	Diya	CSE	92.0	NaN	87.0	85
2	103	Ishaan	ECE	NaN	78.0	80.0	88
3	104	Meera	ECE	74.0	84.0	NaN	76

7. Merge students and sports (outer join) – identify students without sports info.

In [28]:

```
df3 = pd.DataFrame(sports)
```

In [40]:

```
merge_df = pd.merge(df1, df3, on = 'RollNo', how = "outer")
merge_df = merge_df[merge_df['Sport'].isna()]
merge_df
```

Out[40]:

	RollNo	Name	Dept	Math	Science	English	Sport
1	102	Diya	CSE	92.0	NaN	87.0	NaN
3	104	Meera	ECE	74.0	84.0	NaN	NaN
5	106	Anaya	CSE	85.0	89.0	90.0	NaN

join

8. Convert students and attendance into DataFrames with RollNo as index. Perform a left join on index.

In [59]:

```
df_students = pd.DataFrame(students)
df_students = df_students.set_index('RollNo')
df_students
```

Out[59]:

	Name	Dept	Math	Science	English
RollNo					
101	Aarav	CSE	88.0	91.0	85.0
102	Diya	CSE	92.0	NaN	87.0
103	Ishaan	ECE	NaN	78.0	80.0
104	Meera	ECE	74.0	84.0	NaN
105	Kabir	ME	69.0	76.0	74.0
106	Anaya	CSE	85.0	89.0	90.0

In [58]:

```
df_attendance = pd.DataFrame(attendance)
df_attendance = df_attendance.set_index('RollNo')
df_attendance
```

Out[58]:

	Attendance(%)
RollNo	
101	92
102	85
103	88
104	76
107	90

In [60]:

```
df_students.join(df_attendance, how='left')
```

Out[60]:

	Name	Dept	Math	Science	English	Attendance(%)
RollNo						
101	Aarav	CSE	88.0	91.0	85.0	92.0
102	Diya	CSE	92.0	NaN	87.0	85.0
103	Ishaan	ECE	NaN	78.0	80.0	88.0
104	Meera	ECE	74.0	84.0	NaN	76.0
105	Kabir	ME	69.0	76.0	74.0	NaN
106	Anaya	CSE	85.0	89.0	90.0	NaN

concat

9. Create a new small DataFrame of newly admitted students:

In [62]:

```
new_students = {
    'RollNo': [109, 110],
    'Name': ['Rohan', 'Sara'],
    'Dept': ['ECE', 'CSE'],
    'Math': [81, 95],
    'Science': [79, 88],
    'English': [83, 91]
}
```

In [63]:

```
df_new_students = pd.DataFrame(new_students)
```

10. Concatenate this DataFrame with the original students.

In [64]:

```
pd.concat([df1,df_new_students])
```

Out[64]:

	RollNo	Name	Dept	Math	Science	English
0	101	Aarav	CSE	88.0	91.0	85.0
1	102	Diya	CSE	92.0	NaN	87.0
2	103	Ishaan	ECE	NaN	78.0	80.0
3	104	Meera	ECE	74.0	84.0	NaN
4	105	Kabir	ME	69.0	76.0	74.0
5	106	Anaya	CSE	85.0	89.0	90.0
0	109	Rohan	ECE	81.0	79.0	83.0
1	110	Sara	CSE	95.0	88.0	91.0

11. Concatenate students[['RollNo','Name']] with sports column-wise.

In [65]:

```
sports_df = pd.DataFrame(sports)
```

In [67]:

```
# student_df = pd.DataFrame()
pd.concat([df1[['RollNo','Name']], sports_df], axis= 1)
```

Out[67]:

	RollNo	Name	RollNo	Sport
0	101	Aarav	101.0	Cricket
1	102	Diya	103.0	Football
2	103	Ishaan	105.0	Badminton
3	104	Meera	107.0	Hockey
4	105	Kabir	NaN	NaN
5	106	Anaya	NaN	NaN

In [27]:

```
sports = {
    'RollNo': [101, 103, 105, 107],
    'Sport': ['Cricket', 'Football', 'Badminton', 'Hockey']
}
```

Handle missing value

In [71]:

```
import numpy as np
```

In [78]:

```
di = {'Score1': [100, 90, np.nan, 95],
        'Score2': [30, 45, 56, np.nan],
        'Score3': [np.nan, 40, 80, 98]}
```

In [79]:

```
df = pd.DataFrame(di)
```

In [80]:

```
df.to_csv('Scores.csv')
```

12. Read one csv file of your choice

Use different techniques to deal with missing values in the file

In [87]:

```
df = pd.read_csv('Scores.csv',index_col=0)
```

In [88]:

```
df
```

Out[88]:

	Score1	Score2	Score3
0	100.0	30.0	NaN
1	90.0	45.0	40.0
2	NaN	56.0	80.0
3	95.0	NaN	98.0