Darshan UNIVERSITY

Python for Data Science - 2305CS303

Lab - 11

Roll No. : 135

Name : Nikhil Rathod

GroupBy

In [2]:

```
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]
}
students
```

Out[2]:

```
{'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 [12]:

```
import pandas as pd

df = pd.DataFrame(students)
df

df1 = df.groupby('Dept')
df1[['Math','Science','English']].mean()
```

Out[12]:

	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 [13]:

```python
df1['Math'].max()
```

Out[13]:

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

3. Count how many students belong to each department.

In [14]:

```python
df1[['Math','Science','English']].count()
```

Out[14]:

	Math	Science	English
Dept			
CSE	3	2	3
ECE	1	2	1
ME	1	1	1

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

In [15]:

```python
df1['Science'].min()
df1['Science'].max()
df1['Science'].mean()
```

Out[15]:

```
Dept
CSE    90.0
ECE    81.0
ME     76.0
Name: Science, dtype: float64
```

5. For each department, apply multiple aggregations:

Math: mean, max

Science: min, count

In [20]:

```python
df1[['Math','Science','English']].min()
df1[['Math','Science','English']].max()
```

Out[20]:

	Math	Science	English
Dept			
CSE	92.0	91.0	90.0
ECE	74.0	84.0	80.0
ME	69.0	76.0	74.0

Merge

In [21]:

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

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

In [23]:

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

pd.merge(df,newdf,on='RollNo',how='inner')
```

Out[23]:

	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

In [24]:

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

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

In [26]:

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

pd.merge(df,studf,on='RollNo',how='outer')
```

Out[26]:

	RollNo	Name	Dept	Math	Science	English	Sport
0	101	Aarav	CSE	88.0	91.0	85.0	Cricket
1	102	Diya	CSE	92.0	NaN	87.0	NaN
2	103	Ishaan	ECE	NaN	78.0	80.0	Football
3	104	Meera	ECE	74.0	84.0	NaN	NaN
4	105	Kabir	ME	69.0	76.0	74.0	Badminton
5	106	Anaya	CSE	85.0	89.0	90.0	NaN
6	107	NaN	NaN	NaN	NaN	NaN	Hockey

join

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

In [30]:

```
pd.merge(df,newdf,on='RollNo',how='left').set_index('RollNo')
```

Out[30]:

	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 [31]:

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

In [32]:

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

Out[32]:

	RollNo	Name	Dept	Math	Science	English
0	109	Rohan	ECE	81	79	83
1	110	Sara	CSE	95	88	91

10. Concatenate this DataFrame with the original students.

In [33]:

```
pd.concat([df,newstudf])
```

Out[33]:

	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 [ ]:

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

In [36]:

```
pd.concat([df[['RollNo', 'Name']], studf[['Sport']]], axis=1)
```

Out[36]:

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

Handle missing value

12. Read one csv file of your choice

Use different techniques to deal with missing values in the file

In [42]:

```
s = pd.read_csv('students.csv')
s
```

Out[42]:

	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

In [41]:

```
s.dropna()
```

Out[41]:

	RollNo	Name	Dept	Math	Science	English
0	101	Aarav	CSE	88.0	91.0	85.0
4	105	Kabir	ME	69.0	76.0	74.0
5	106	Anaya	CSE	85.0	89.0	90.0

In [44]:

```
s.dropna(axis=1)
```

Out[44]:

	RollNo	Name	Dept
0	101	Aarav	CSE
1	102	Diya	CSE
2	103	Ishaan	ECE
3	104	Meera	ECE
4	105	Kabir	ME
5	106	Anaya	CSE

In [45]:

```
s.dropna(subset=['Math'])
```

Out[45]:

	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
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

In [46]:

```
s.fillna(0)
```

Out[46]:

	RollNo	Name	Dept	Math	Science	English
0	101	Aarav	CSE	88.0	91.0	85.0
1	102	Diya	CSE	92.0	0.0	87.0
2	103	Ishaan	ECE	0.0	78.0	80.0
3	104	Meera	ECE	74.0	84.0	0.0
4	105	Kabir	ME	69.0	76.0	74.0
5	106	Anaya	CSE	85.0	89.0	90.0

In [47]:

```
s.fillna(method='ffill')
```

```
C:\Users\Nikhil\AppData\Local\Temp\ipykernel_10532\1029756637.py:1: FutureWarnin
g: DataFrame.fillna with 'method' is deprecated and will raise in a future versio
n. Use obj.ffill() or obj.bfill() instead.
  s.fillna(method='ffill')
```

Out[47]:

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

In [48]:

```
s.fillna(method='bfill')
```

```
C:\Users\Nikhil\AppData\Local\Temp\ipykernel_10532\1641295691.py:1: FutureWarnin
g: DataFrame.fillna with 'method' is deprecated and will raise in a future versio
n. Use obj.ffill() or obj.bfill() instead.
  s.fillna(method='bfill')
```

Out[48]:

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

In [51]:

```
s['Science'].fillna(s['Science'].mean())
```

Out[51]:

```
0    91.0
1    83.6
2    78.0
3    84.0
4    76.0
5    89.0
Name: Science, dtype: float64
```