In [1]: #To read the file
In [2]: import pandas as pd
In [3]: df=pd.read\_csv(r'C:\Users\arati\OneDrive\Desktop\SQL\data.csv')
In [4]: df
Out[4]: destination passanger weather temperature time coupon expiration

		destination	passanger	weather	temperature	time	coupon	expiration
	0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	10
	1	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	2ł
	2	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	21
	3	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	21
	4	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	1c
	•••				•••		•••	
	12679	Home	Partner	Rainy	55	6PM	Carry out & Take away	1c
	12680	Work	Alone	Rainy	55	7AM	Carry out & Take away	1c
	12681	Work	Alone	Snowy	30	7AM	Coffee House	1c
	12682	Work	Alone	Snowy	30	7AM	Bar	1c
	12683	Work	Alone	Sunny	80	7AM	Restaurant(20- 50)	21

12684 rows × 27 columns



In [5]: df[['weather','temperature']] #Select wether, temperature from dataset\_1

Out[5]:		weather	temperature
	0	Sunny	55
	1	Sunny	80
	2	Sunny	80
	3	Sunny	80
	4	Sunny	80
	•••		
	12679	Rainy	55
	12680	Rainy	55
	12681	Snowy	30
	12682	Snowy	30

**12683** Sunny

In [6]: df.head(10) #To see first 10 rows

80

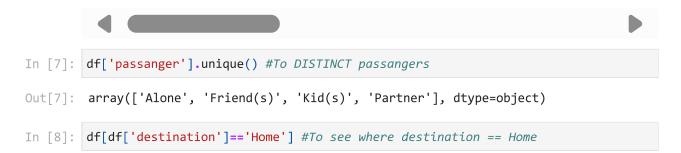
Out[6]:		destination	passanger	weather	temperature	time	coupon	expiration	ge
	0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	1d	Fe
	1	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	2h	F€
	2	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	2h	F€
	3	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	2h	F€
	4	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	1d	F€
	5	No Urgent Place	Friend(s)	Sunny	80	6PM	Restaurant(<20)	2h	F€
	6	No Urgent Place	Friend(s)	Sunny	55	2PM	Carry out & Take away	1d	F€
	7	No Urgent Place	Kid(s)	Sunny	80	10AM	Restaurant(<20)	2h	F€
	8	No Urgent Place	Kid(s)	Sunny	80	10AM	Carry out & Take away	2h	Fe

No Urgent

Place

Kid(s)

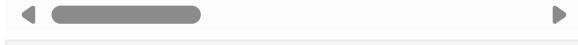
Sunny



80 10AM

1d F€

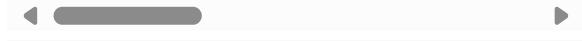
Out[8]:		destination	passanger	weather	temperature	time	coupon	expiratior
	13	Home	Alone	Sunny	55	6PM	Bar	1c
	14	Home	Alone	Sunny	55	6PM	Restaurant(20- 50)	1c
	15	Home	Alone	Sunny	80	6PM	Coffee House	21
	35	Home	Alone	Sunny	55	6PM	Bar	1c
	36	Home	Alone	Sunny	55	6PM	Restaurant(20- 50)	1c
	•••	•••	•••		•••		***	••
	12675	Home	Alone	Snowy	30	10PM	Coffee House	2h
	12676	Home	Alone	Sunny	80	6PM	Restaurant(20- 50)	1c
	12677	Home	Partner	Sunny	30	6PM	Restaurant(<20)	1c
	12678	Home	Partner	Sunny	30	10PM	Restaurant(<20)	2h
	12679	Home	Partner	Rainy	55	6PM	Carry out & Take away	1c



In [9]: df.sort\_values('coupon') #Shorting the COUPON value in ascending order

			,			
$\cap$	1.1	+	ш	a	-	,
$\cup$	и	L.	ш	フ	-	9

	destination	passanger	weather	temperature	time	coupon	expiratior
11702	Home	Partner	Sunny	30	10PM	Bar	2ŀ
9930	No Urgent Place	Alone	Snowy	30	2PM	Bar	1c
10632	Home	Alone	Rainy	55	6PM	Bar	1c
7997	No Urgent Place	Friend(s)	Rainy	55	10PM	Bar	2h
11166	Work	Alone	Snowy	30	7AM	Bar	1c
•••							
10476	Home	Alone	Sunny	80	6PM	Restaurant(<20)	1c
5447	Home	Alone	Sunny	80	10PM	Restaurant(<20)	2ŀ
10478	Home	Alone	Snowy	30	10PM	Restaurant(<20)	2h
5440	No Urgent Place	Alone	Sunny	80	2PM	Restaurant(<20)	2ŀ
0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	1c



In [10]: df.rename(columns={'destination':'Destination'},inplace=True)

In [11]: df

			_	
Out	- Г	11	7	0
Out	-	44		0

		Destination	passanger	weather	temperature	time	coupon	expiratio
	0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	10
	1	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	21
	2	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	21
	3	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	21
	4	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	10
	•••		•••		•••		•••	
	12679	Home	Partner	Rainy	55	6PM	Carry out & Take away	10
	12680	Work	Alone	Rainy	55	7AM	Carry out & Take away	10
	12681	Work	Alone	Snowy	30	7AM	Coffee House	10
	12682	Work	Alone	Snowy	30	7AM	Bar	10
	12683	Work	Alone	Sunny	80	7AM	Restaurant(20- 50)	21



In [12]: df.groupby('occupation').size().to\_frame('Count').reset\_index() # Select occupat

	occupation	Count
0	Architecture & Engineering	175
1	Arts Design Entertainment Sports & Media	629
2	Building & Grounds Cleaning & Maintenance	44
3	Business & Financial	544
4	Community & Social Services	241
5	Computer & Mathematical	1408
6	Construction & Extraction	154
7	Education&Training&Library	943
8	Farming Fishing & Forestry	43
9	Food Preparation & Serving Related	298
10	Healthcare Practitioners & Technical	244
11	Healthcare Support	242
12	Installation Maintenance & Repair	133
13	Legal	219
14	Life Physical Social Science	170
15	Management	838
16	Office & Administrative Support	639
17	Personal Care & Service	175
18	Production Occupations	110
19	Protective Service	175
20	Retired	495
21	Sales & Related	1093
22	Student	1584
23	Transportation & Material Moving	218
24	Unemployed	1870

occupation Count

In [13]: df.groupby('weather')['temperature'].mean().to\_frame('avg\_temp').reset\_index() #
Out[13]: weather avg\_temp

0	Rainy	55.000000
1	Snowy	30.000000
2	Sunny	68.946271

```
In [14]: #Select weather count(temperature) as Count_temp from dataset_1 groupby weather
         df.groupby('weather')['temperature'].size().to_frame('Count_temp').reset_index()
Out[14]:
            weather Count_temp
          0
                            1210
               Rainy
                            1405
          1
              Snowy
          2
               Sunny
                           10069
In [15]: #Select weather count( distinct temperature) as count_distinct_temp from dataset
         df.groupby('weather')['temperature'].nunique().to_frame('count_distinct_temp').r
Out[15]:
            weather count_distinct_temp
          0
                                      1
               Rainy
          1
                                      1
              Snowy
                                      3
          2
               Sunny
In [16]: #SELECT weather from SUM(temperature) as sum_temp from database_1 groupby weathe
         df.groupby('weather')['temperature'].sum().to_frame('sum_temp').reset_index()
Out[16]:
            weather sum_temp
          0
               Rainy
                          66550
              Snowy
                         42150
          2
               Sunny
                        694220
In [17]: #SELECT weather from MIN(temperature) as min_temp from database_1 groupby weather
         df.groupby('weather')['temperature'].min().to_frame('min_temp').reset_index()
Out[17]:
            weather min_temp
          0
                            55
               Rainy
                            30
          1
              Snowy
          2
                            30
               Sunny
In [18]: #SELECT weather from MAX(temperature) as max_temp from database_1 groupby weather
         df.groupby('weather')['temperature'].max().to_frame('max_temp').reset_index()
Out[18]:
            weather max_temp
          0
                             55
               Rainy
                             30
              Snowy
          2
                             80
               Sunny
In [19]: #SELECT occupation from dataset_1 groupby OCCUPATION HAVING occupation = student
```

df.groupby('occupation').filter(lambda x: x['occupation'].iloc[0] == 'Student').

Out[19]: occupation

Student 1584 dtype: int64

In [20]: df1=df.copy()

In [21]: df1

Out[21]:

	Destination	passanger	weather	temperature	time	coupon	expiratio
0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	10
1	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	21
2	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	21
3	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	21
4	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	10
•••	•••						
12679	Home	Partner	Rainy	55	6PM	Carry out & Take away	10
12680	Work	Alone	Rainy	55	7AM	Carry out & Take away	10
12681	Work	Alone	Snowy	30	7AM	Coffee House	10
12682	Work	Alone	Snowy	30	7AM	Bar	10
12683	Work	Alone	Sunny	80	7AM	Restaurant(20- 50)	21

12684 rows × 27 columns

**1** 

In [22]: pd.concat([df, df1])['Destination'].drop\_duplicates()

Out[22]: 0 No Urgent Place 13 Home

16 Work

Name: Destination, dtype: object

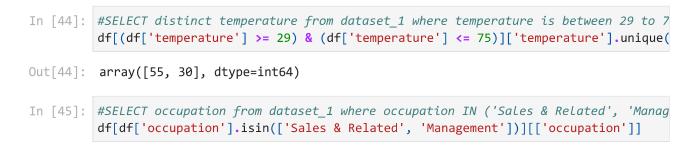
In [42]: #SELECT destination, passangers FROM dataset\_1 where passanger=Alone
df[df['passanger'] == 'Alone'][['Destination', 'passanger']]

	0.1		-
$\cap$	ıtl	1/17	
Vι	<i>I</i> L	144	1 .

	Destination	passanger
0	No Urgent Place	Alone
13	Home	Alone
14	Home	Alone
15	Home	Alone
16	Work	Alone
•••	•••	•••
12676	Home	Alone
12680	Work	Alone
12681	Work	Alone
12682	Work	Alone
12683	Work	Alone

```
In [43]: #SELECT weather from dataset_1 where weather is sunny
df[df['weather'].str.startswith('Sun')]
```

43]:		Destination	passanger	weather	temperature	time	coupon	expiratio
	0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	10
	1	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	21
	2	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	21
	3	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	21
	4	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	10
	•••			•••				
	12673	Home	Alone	Sunny	30	6PM	Carry out & Take away	10
	12676	Home	Alone	Sunny	80	6PM	Restaurant(20- 50)	10
	12677	Home	Partner	Sunny	30	6PM	Restaurant(<20)	10
	12678	Home	Partner	Sunny	30	10PM	Restaurant(<20)	21
	12683	Work	Alone	Sunny	80	7AM	Restaurant(20- 50)	21
10069 rows × 27 columns								
	4							



Out[45]:		occupation
	193	Sales & Related
	194	Sales & Related
	195	Sales & Related
	196	Sales & Related
	197	Sales & Related
	•••	
	12679	Sales & Related
	12680	Sales & Related
	12681	Sales & Related
	12682	Sales & Related
	12683	Sales & Related
	1931 rov	ws × 1 columns

In [ ]: