

```
In [1]: import pandas as pd
```

```
In [12]: #SELECT * FROM dataset_1;
df = pd.read_csv(r'C:\Users\Ravi\Desktop\sql\dataset_1_202602121649.csv')
df
```

Out[12]:

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

12684 rows × 27 columns



```
In [4]: df.shape
```

```
Out[4]: (12684, 27)
```

```
In [5]: df.head()
```

Out[5]:

	destination	passanger	weather	temperature	time	coupon	expiration	ge
<b>0</b>	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	1d	Fe
<b>1</b>	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	2h	Fe
<b>2</b>	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	2h	Fe
<b>3</b>	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	2h	Fe
<b>4</b>	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	1d	Fe

5 rows × 27 columns

In [6]: `df.columns`

Out[6]: Index(['destination', 'passanger', 'weather', 'temperature', 'time', 'coupon', 'expiration', 'gender', 'age', 'maritalStatus', 'has\_children', 'education', 'occupation', 'income', 'car', 'Bar', 'CoffeeHouse', 'CarryAway', 'RestaurantLessThan20', 'Restaurant20To50', 'toCoupon\_GEQ5min', 'toCoupon\_GEQ15min', 'toCoupon\_GEQ25min', 'direction\_same', 'direction\_opp', 'Y', 'row\_count'], dtype='object')

In [13]: `#SELECT weather,temperature FROM dataset_1;`  
`df[['weather', 'temperature']]`

Out[13]:

	weather	temperature
<b>0</b>	Sunny	55
<b>1</b>	Sunny	80
<b>2</b>	Sunny	80
<b>3</b>	Sunny	80
<b>4</b>	Sunny	80
...	...	...
<b>12679</b>	Rainy	55
<b>12680</b>	Rainy	55
<b>12681</b>	Snowy	30
<b>12682</b>	Snowy	30
<b>12683</b>	Sunny	80

12684 rows × 2 columns

```
In [14]: #SELECT * FROM dataset_1 LIMIT 10;
df[0:10]
```

Out[14]:

	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	Fe
2	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	2h	Fe
3	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	2h	Fe
4	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	1d	Fe
5	No Urgent Place	Friend(s)	Sunny	80	6PM	Restaurant(<20)	2h	Fe
6	No Urgent Place	Friend(s)	Sunny	55	2PM	Carry out & Take away	1d	Fe
7	No Urgent Place	Kid(s)	Sunny	80	10AM	Restaurant(<20)	2h	Fe
8	No Urgent Place	Kid(s)	Sunny	80	10AM	Carry out & Take away	2h	Fe
9	No Urgent Place	Kid(s)	Sunny	80	10AM	Bar	1d	Fe

10 rows × 27 columns



```
In [10]: df['passanger'].nunique()
```

Out[10]: 4

```
In [15]: #SELECT DISTINCT passenger FROM dataset_1;
df['passanger'].unique()
```

Out[15]: array(['Alone', 'Friend(s)', 'Kid(s)', 'Partner'], dtype=object)

```
In [16]: #SELECT * FROM dataset_1 WHERE destination = 'Home';
df[df['destination']=='Home']
```

Out[16]:

	<b>destination</b>	<b>passanger</b>	<b>weather</b>	<b>temperature</b>	<b>time</b>	<b>coupon</b>	<b>expiration</b>
<b>13</b>	Home	Alone	Sunny	55	6PM	Bar	1c
<b>14</b>	Home	Alone	Sunny	55	6PM	Restaurant(20-50)	1c
<b>15</b>	Home	Alone	Sunny	80	6PM	Coffee House	2h
<b>35</b>	Home	Alone	Sunny	55	6PM	Bar	1c
<b>36</b>	Home	Alone	Sunny	55	6PM	Restaurant(20-50)	1c
...	...	...	...	...	...	...	..
<b>12675</b>	Home	Alone	Snowy	30	10PM	Coffee House	2h
<b>12676</b>	Home	Alone	Sunny	80	6PM	Restaurant(20-50)	1c
<b>12677</b>	Home	Partner	Sunny	30	6PM	Restaurant(<20)	1c
<b>12678</b>	Home	Partner	Sunny	30	10PM	Restaurant(<20)	2h
<b>12679</b>	Home	Partner	Rainy	55	6PM	Carry out & Take away	1c

3237 rows × 27 columns


In [17]: `#SELECT *FROM dataset_1 ORDER BY coupon;  
df.sort_values('coupon')`

Out[17]:

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

12684 rows × 27 columns



In [18]: `#SELECT destination as Destination FROM dataset_1;  
df.rename({'destination':'Destination'},inplace = True)`

In [19]: `df`

Out[19]:

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

12684 rows × 27 columns



In [21]: #SELECT occupation FROM dataset\_1 GROUP BY occupation

df.groupby('occupation').size().to\_frame('Count').reset\_index()

Out[21]:

	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

In [22]:

```
# SELECT weather ,AVG(temperature) as avg_temp FROM dataset_1 GROUP BY weather;
df.groupby('weather')[ 'temperature'].mean().to_frame('avg_temp').reset_index()
```

Out[22]:

	weather	avg_temp
0	Rainy	55.000000
1	Snowy	30.000000
2	Sunny	68.946271

```
In [23]: #SELECT weather ,COUNT( temperature) AS count_temp FROM dataset_1 GROUP BY weather
df.groupby('weather')[ 'temperature'].size().to_frame('Count_temp').reset_index()
```

Out[23]:

	weather	Count_temp
0	Rainy	1210
1	Snowy	1405
2	Sunny	10069

```
In [24]: # SELECT weather ,COUNT(DISTINCT temperature) AS count_distinct_temp FROM dataset_1
df.groupby('weather')[ 'temperature'].nunique().to_frame('count_distinct_temp').reset_index()
```

Out[24]:

	weather	count_distinct_temp
0	Rainy	1
1	Snowy	1
2	Sunny	3

```
In [25]: #SELECT weather ,SUM(temperature) AS sum_temp FROM dataset_1 GROUP BY weather;
df.groupby('weather')[ 'temperature'].sum().to_frame('sum_temp').reset_index()
```

Out[25]:

	weather	sum_temp
0	Rainy	66550
1	Snowy	42150
2	Sunny	694220

```
In [26]: #SELECT weather ,MIN(temperature) AS min_temp FROM dataset_1 GROUP BY weather;
df.groupby('weather')[ 'temperature'].min().to_frame('min_temp').reset_index()
```

Out[26]:

	weather	min_temp
0	Rainy	55
1	Snowy	30
2	Sunny	30

```
In [27]: #SELECT weather ,MAX(temperature) AS max_temp FROM dataset_1 GROUP BY weather;
df.groupby('weather')[ 'temperature'].max().to_frame('max_temp').reset_index()
```

Out[27]:

	weather	max_temp
0	Rainy	55
1	Snowy	30
2	Sunny	80

```
In [28]: #SELECT occupation FROM dataset_1 GROUP BY occupation HAVING occupation='Student'
df.groupby('occupation').filter(lambda x: x['occupation'].iloc[0] ==
```

```
'Student').groupby('occupation').size()
```

Out[28]: occupation  
Student 1584  
dtype: int64

In [31]: # SELECT DISTINCT destination FROM(SELECT \* FROM dataset\_1 UNION SELECT \* FROM t  
pd.concat([df, df])['destination'].drop\_duplicates()

Out[31]: 0 No Urgent Place  
13 Home  
16 Work  
Name: destination, dtype: object

In [43]: # SELECT a.destination,a.time,b.part\_of\_day FROM dataset\_1 a INNER JOIN table\_to  
df2 = pd.read\_csv(r'C:\Users\Ravi\Desktop\sql\table\_to\_join\_202602131046.csv')  
pd.merge(df, df2[['time', 'part\_of\_day']], on='time', how='inner')[['destination', 'time', 'part\_of\_day']]

Out[43]:

	destination	time	part_of_day
0	No Urgent Place	2PM	Afternoon
1	No Urgent Place	10AM	Morning
2	No Urgent Place	10AM	Morning
3	No Urgent Place	2PM	Afternoon
4	No Urgent Place	2PM	Afternoon
...	...	...	...
12679	Home	6PM	Evening
12680	Work	7AM	Morning
12681	Work	7AM	Morning
12682	Work	7AM	Morning
12683	Work	7AM	Morning

12684 rows × 3 columns

In [37]: #SELECT destination ,passenger FROM(SELECT\*FROM dataset\_1 WHERE passenger = 'Alone'  
df[df['passenger'] == 'Alone'][['destination', 'passenger']]

Out[37]:

	destination	passanger
<b>0</b>	No Urgent Place	Alone
<b>13</b>	Home	Alone
<b>14</b>	Home	Alone
<b>15</b>	Home	Alone
<b>16</b>	Work	Alone
...	...	...
<b>12676</b>	Home	Alone
<b>12680</b>	Work	Alone
<b>12681</b>	Work	Alone
<b>12682</b>	Work	Alone
<b>12683</b>	Work	Alone

7305 rows × 2 columns

In [38]:  

```
#SELECT * FROM dataset_1 WHERE weather LIKE 'Sun%';
df[df['weather'].str.startswith('Sun')]
```

Out[38]:

	<b>destination</b>	<b>passanger</b>	<b>weather</b>	<b>temperature</b>	<b>time</b>	<b>coupon</b>	<b>expiration</b>
<b>0</b>	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	1c
<b>1</b>	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	2h
<b>2</b>	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	2h
<b>3</b>	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	2h
<b>4</b>	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	1c
...	...	...	...	...	...	...	..
<b>12673</b>	Home	Alone	Sunny	30	6PM	Carry out & Take away	1c
<b>12676</b>	Home	Alone	Sunny	80	6PM	Restaurant(20-50)	1c
<b>12677</b>	Home	Partner	Sunny	30	6PM	Restaurant(<20)	1c
<b>12678</b>	Home	Partner	Sunny	30	10PM	Restaurant(<20)	2h
<b>12683</b>	Work	Alone	Sunny	80	7AM	Restaurant(20-50)	2h

10069 rows × 27 columns



In [39]:

```
#SELECT DISTINCT temperature FROM dataset_1 WHERE temperature BETWEEN 29 AND 75;
df[(df['temperature'] >= 29) & (df['temperature'] <= 75)]['temperature'].unique()
```

Out[39]: array([55, 30])

In [40]:

```
# SELECT occupation FROM dataset_1 WHERE occupation IN('Sales & Related', 'Management')
df[df['occupation'].isin(['Sales & Related', 'Management'])][['occupation']]
```

Out[40]:

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 rows × 1 columns

In [ ]:

In [ ]: