In [8]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

%matplotlib inline

import warnings
warnings.filterwarnings('ignore')
```

In [33]:

```
df = pd.read_csv(r'C:\Users\LENOVO\Desktop\Monty Datascien\dataset_1_202305151046.csv')
```

In [34]:

```
# SELECT*FROM dataset_1
df
```

Out[34]:

	destination	passanger	weather	temperature	time	coupon	expiration	gender	age	mar
0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	1d	Female	21	U
1	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	2h	Female	21	U
2	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	2h	Female	21	U
3	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	2h	Female	21	U
4	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	1d	Female	21	U
...	...	...	...	...	...	...	...	...	...	
12679	Home	Partner	Rainy	55	6PM	Carry out & Take away	1d	Male	26	

In [31]:

```
df.columns
```

Out[31]:

```
Index(['Destination', 'passanger', 'weather', 'temperature', 'time', 'coup
on',
       '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 [4]:

```
# SELECT weather,temperature FROM dataset_1
df[['weather','temperature']]
```

Out[4]:

	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	80

12684 rows × 2 columns

In [5]:

```
# SELECT*FROM dataset_1 LIMIT 10
df.head(10)
```

Out[5]:

	destination	passanger	weather	temperature	time	coupon	expiration	gender
0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	1d	Female
1	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	2h	Female
2	No Urgent Place	Friend(s)	Sunny	80	10AM	Carry out & Take away	2h	Female
3	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	2h	Female
4	No Urgent Place	Friend(s)	Sunny	80	2PM	Coffee House	1d	Female
5	No Urgent Place	Friend(s)	Sunny	80	6PM	Restaurant(<20)	2h	Female
6	No Urgent Place	Friend(s)	Sunny	55	2PM	Carry out & Take away	1d	Female
7	No Urgent Place	Kid(s)	Sunny	80	10AM	Restaurant(<20)	2h	Female
8	No Urgent Place	Kid(s)	Sunny	80	10AM	Carry out & Take away	2h	Female
9	No Urgent Place	Kid(s)	Sunny	80	10AM	Bar	1d	Female

10 rows × 27 columns

In [7]:

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

Out[7]:

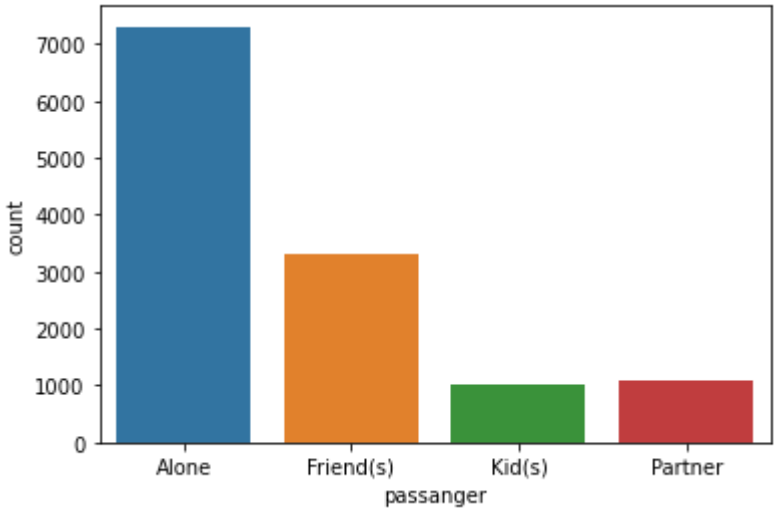
```
array(['Alone', 'Friend(s)', 'Kid(s)', 'Partner'], dtype=object)
```

In [11]:

```
sns.countplot('passanger',data=df)
```

Out[11]:

```
<AxesSubplot:xlabel='passanger', ylabel='count'>
```



From this we understood that passengers traveling by their own are more comapred with rest.

In [14]:

```
# SELECT * FROM dataset_1 WHERE destination = 'Home'
df[df['destination']=='Home']
```

Out[14]:

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

3237 rows × 27 columns

In [18]:

```
# SELECT *FROM dataset_1 ORDER BY coupn
l=df.groupby('coupon')
```

In [24]:

```
l.first()
```

Out[24]:

coupon	destination	passanger	weather	temperature	time	expiration	gender	ag
Bar	No Urgent Place	Kid(s)	Sunny	80	10AM	1d	Female	2
Carry out & Take away	No Urgent Place	Friend(s)	Sunny	80	10AM	2h	Female	2
Coffee House	No Urgent Place	Friend(s)	Sunny	80	10AM	2h	Female	2
Restaurant(20-50)	Home	Alone	Sunny	55	6PM	1d	Female	2
Restaurant(<20)	No Urgent Place	Alone	Sunny	55	2PM	1d	Female	2

5 rows × 26 columns

In [29]:

```
# SELECT destination as Destination FROM dataset_1
df.rename(columns={'destination':'Destination'},inplace=True)
```

In [30]:

```
df
```

Out[30]:

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

12684 rows × 27 columns

In [38]:

```
# SELECT occupation FROM dataset_1 GROUP BY occupation
df.groupby('occupation').size().to_frame('Count').reset_index()
```

Out[38]:

	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 [42]:

```
# 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[42]:

	weather	avg_temp
0	Rainy	55.000000
1	Snowy	30.000000
2	Sunny	68.946271

In [43]:

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

Out[43]:

	weather	Count_temp
0	Rainy	1210
1	Snowy	1405
2	Sunny	10069

In [47]:

```
# SELECT weather ,COUNT(DISTINCT temperature) AS count_distinct_temp FROM dataset_1 GROU
df.groupby('weather')['temperature'].unique().to_frame('count_distinct_temp').reset_inde
```

Out[47]:

	weather	count_distinct_temp
0	Rainy	[55]
1	Snowy	[30]
2	Sunny	[55, 80, 30]

In [48]:

```
# 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[48]:

	weather	sum_temp
0	Rainy	66550
1	Snowy	42150
2	Sunny	694220

In [49]:

```
#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[49]:

	weather	min_temp
0	Rainy	55
1	Snowy	30
2	Sunny	30

In [50]:

```
#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[50]:

	weather	max_temp
0	Rainy	55
1	Snowy	30
2	Sunny	80

In [55]:

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

Out[55]:

```
occupation
Student    1584
dtype: int64
```

In [56]:

```
df1 = pd.read_csv(r'C:\Users\LENOVO\Desktop\Monty Datascien\table_to_union_202305151333.
```

In [57]:

```
df1
```

Out[57]:

	destination	passanger	weather	temperature	time	coupon	expiration	gender	a
0	UNION	UNION	UNION	55	2PM	Restaurant(<20)	1d	Female	

1 rows × 27 columns

In [59]:

```
#SELECT*FROM dataset_1 UNION SELECT *FROM table_to_union or SELECT DISTINCT destination
pd.concat([df, df1])['destination'].drop_duplicates()
```

Out[59]:

```
0     No Urgent Place
13               Home
16               Work
0               UNION
Name: destination, dtype: object
```

In [60]:

```
df2 = pd.read_csv(r'C:\Users\LENOVO\Desktop\Monty Datascien\table_to_join_202305151352.c
```

In [61]:

```
df2
```

Out[61]:

	time	part_of_day
0	2PM	Afternoon
1	10AM	Morning
2	6PM	Evening
3	7AM	Morning
4	10PM	Night

In [62]:

```
# SELECT a.destination,a.time,b.part_of_day FROM dataset_1 a INNER JOIN table_to_join b (
pd.merge(df, df2[['time', 'part_of_day']], on='time', how='inner')[['destination', 'time
```

Out[62]:

	destination	time	part_of_day
0	No Urgent Place	2PM	Afternoon
1	No Urgent Place	2PM	Afternoon
2	No Urgent Place	2PM	Afternoon
3	No Urgent Place	2PM	Afternoon
4	No Urgent Place	2PM	Afternoon
...	...	...	...
12679	No Urgent Place	10PM	Night
12680	No Urgent Place	10PM	Night
12681	Home	10PM	Night
12682	Home	10PM	Night
12683	Home	10PM	Night

12684 rows × 3 columns

In [64]:

```
#SELECT destination ,passenger FROM(SELECT*FROM dataset_1 WHERE passenger = 'Alone')
df[df['passanger'] == 'Alone'][['destination', 'passanger']]
```

Out[64]:

	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

7305 rows × 2 columns

In [65]:

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

Out[65]:

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

10069 rows × 27 columns

In [70]:

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

Out[70]:

```
array([55, 30], dtype=int64)
```

In [68]:

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

Out[68]:

	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 [71]:

```
df.head(2)
```

Out[71]:

	destination	passanger	weather	temperature	time	coupon	expiration	gender
0	No Urgent Place	Alone	Sunny	55	2PM	Restaurant(<20)	1d	Female
1	No Urgent Place	Friend(s)	Sunny	80	10AM	Coffee House	2h	Female

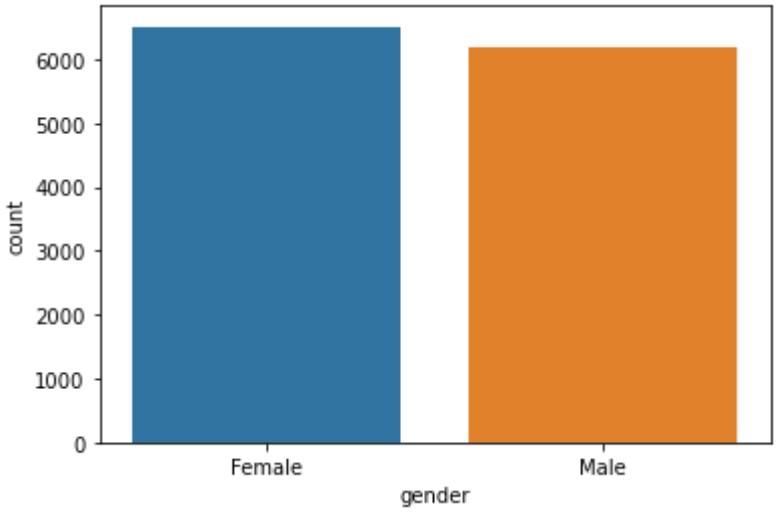
2 rows × 27 columns

In [74]:

```
sns.countplot('gender',data=df)
```

Out[74]:

```
<AxesSubplot:xlabel='gender', ylabel='count'>
```



In [77]:

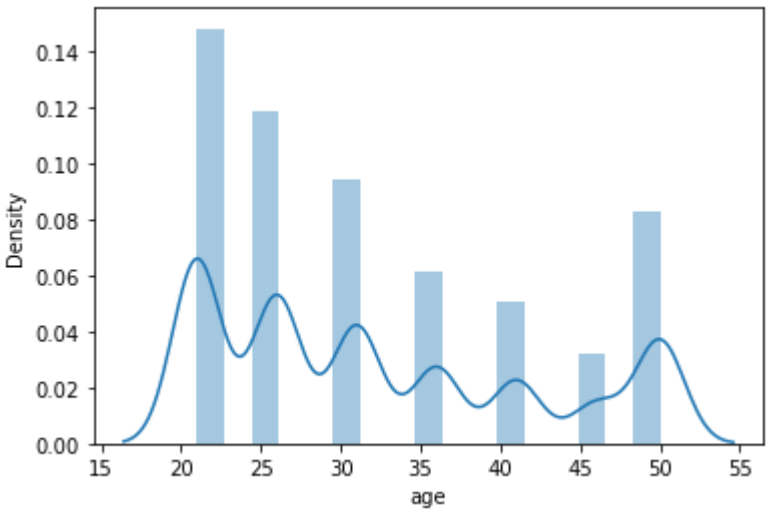
```
 # Using regular expression
df['age'] = df['age'].str.extract('(\d+)')
```

In [78]:

```
sns.distplot(df['age'])
```

Out[78]:

```
<AxesSubplot:xlabel='age', ylabel='Density'>
```



In [ ]:

```

```