IMPORTING LIB

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

IMPORTING DATASET

```
df = pd.read_excel(r"C:\Users\anush\Downloads\DA\Food Delivery Complaints.xlsx")
```

EXPLORING DATA

```
df.head()
```

	order_id	customer_name	age	gender	city	order_date	food_category	delivery_partner	delivery_time_mins	is_dela
0	1.0	Alasteir Sporrij	23.0	Female	Chennai	10/23/2025	Burger	Zippy	83.0	
1	2.0	Lettie Cleare	30.0	Male	Mumbai	4/19/2025	South Indian	QuickKart	108.0	
2	3.0	Danika Tryme	17.0	Other	NaN	6/7/2025	NaN	QuickKart	81.0	N
3	4.0	NaN	46.0	Male	Delhi	7/5/2025	Chinese	SpeedEats	106.0	
4	5.0	Shaun Dodshon	22.0	Male	Hyderabad	4/5/2025	Biriyani	QuickKart	39.0	

```
df.tail()
```

	order_id	customer_name	age	gender	city	order_date	food_category	delivery_partner	delivery_time_mins	is_d
998	999.0	NaN	54.0	Male	NaN	9/14/2025	Chinese	QuickKart	90.0	
999	1000.0	Demott Reeken	49.0	Female	Hyderabad	9/11/2025	Desserts	DashX	105.0	
1000	13.0	Tabor Corbet	31.0	Male	Mumbai	3/31/2025	Biryani	Zippy	53.0	
1001	13.0	Tabor Corbet	31.0	Male	Mumbai	3/31/2025	Biryani	Zippy	53.0	
1002	13.0	Tabor Corbet	31.0	Male	Mumbai	3/31/2025	Biryani	Zippy	53.0	

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1003 entries, 0 to 1002
Data columns (total 14 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   order_id            883 non-null    float64
 1   customer_name       898 non-null    object 
 2   age                 893 non-null    float64
 3   gender              899 non-null    object 
 4   city                874 non-null    object 
 5   order_date          1003 non-null   object 
 6   food_category       906 non-null    object 
 7   delivery_partner    884 non-null    object 
 8   delivery_time_mins  891 non-null    float64
 9   is_delayed          891 non-null    float64
 10  rating              1003 non-null   int64  
 11  complaint           888 non-null    object 
 12  refund_amount       1003 non-null   float64
 13  duplicate_flag      896 non-null    object 
dtypes: float64(5), int64(1), object(8)
memory usage: 109.8+ KB
```

```
df.describe()
```

	order_id	age	delivery_time_mins	is_delayed	rating	refund_amount
count	883.000000	893.000000	891.000000	891.000000	1003.000000	1003.000000
mean	502.369196	37.398656	65.720539	0.601571	3.013958	249.494437
std	293.589367	13.074300	32.159389	0.489850	1.428888	147.172037
min	1.000000	16.000000	10.000000	0.000000	1.000000	0.310000
25%	241.500000	26.000000	38.000000	0.000000	2.000000	119.680000
50%	506.000000	37.000000	66.000000	1.000000	3.000000	251.260000
75%	757.500000	49.000000	94.000000	1.000000	4.000000	381.195000
max	1000.000000	60.000000	120.000000	1.000000	5.000000	499.950000

CHECKING FOR NULL VALUES

```
df.isnull().sum()
```

```
order_id             120
customer_name        105
age                  110
gender               104
city                 129
order_date             0
food_category         97
delivery_partner     119
delivery_time_mins   112
is_delayed           112
rating                 0
complaint            115
refund_amount          0
duplicate_flag       107
dtype: int64
```

DROPING NULL VALUES

```
df = df.dropna(subset =['order_id','customer_name','age','gender','city','food_category','delivery_partner',
'delivery_time_mins','is_delayed','complaint','duplicate_flag' ])
```

```
df.isnull().sum()
```

```
order_id            0
customer_name       0
age                 0
gender              0
city                0
order_date          0
food_category       0
delivery_partner    0
delivery_time_mins  0
is_delayed          0
rating              0
complaint           0
refund_amount       0
duplicate_flag      0
dtype: int64
```

CHECKING FOR DUPLICATES

```
df.duplicated()
```

```
0      False
1      False
4      False
19     False
23     False
       ...
977    False
980    False
983    False
992    False
999    False
Length: 269, dtype: bool
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 269 entries, 0 to 999
Data columns (total 14 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   order_id           269 non-null    float64
 1   customer_name      269 non-null    object 
 2   age                269 non-null    float64
 3   gender             269 non-null    object 
 4   city               269 non-null    object 
 5   order_date         269 non-null    object 
 6   food_category      269 non-null    object 
 7   delivery_partner   269 non-null    object 
 8   delivery_time_mins 269 non-null    float64
 9   is_delayed         269 non-null    float64
 10  rating             269 non-null    int64  
 11  complaint          269 non-null    object 
 12  refund_amount      269 non-null    float64
 13  duplicate_flag     269 non-null    object 
dtypes: float64(5), int64(1), object(8)
memory usage: 31.5+ KB
```

CHANGING DATATYPES

```
df['age'] = df['age'].astype('int')
```

```
df['order_date'] = pd.to_datetime(df['order_date'])
```

```
df['is_delayed'].unique()
```

```
array([1., 0.])
```

```
df['is_delayed'] =df['is_delayed'].astype(str).str.replace('1.0','1')
```

```
df['is_delayed'] =df['is_delayed'].astype(str).str.replace('0.0','0')
```

```
df['is_delayed'] = df['is_delayed'].astype('int')
```

```
df['delivery_time_mins'] = df['delivery_time_mins'].astype('int')
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 269 entries, 0 to 999
Data columns (total 14 columns):
 #   Column             Non-Null Count  Dtype         
---  ------             --------------  -----         
 0   order_id           269 non-null    float64       
 1   customer_name      269 non-null    object        
 2   age                269 non-null    int64         
 3   gender             269 non-null    object        
 4   city               269 non-null    object        
 5   order_date         269 non-null    datetime64[ns]
 6   food_category      269 non-null    object        
 7   delivery_partner   269 non-null    object        
 8   delivery_time_mins 269 non-null    int64         
 9   is_delayed         269 non-null    int64         
 10  rating             269 non-null    int64         
 11  complaint          269 non-null    object        
 12  refund_amount      269 non-null    float64       
 13  duplicate_flag     269 non-null    object        
dtypes: datetime64[ns](1), float64(2), int64(4), object(7)
memory usage: 31.5+ KB
```

```
df['food_category'] = df['food_category'].replace(['Biryani', 'Birynai'],'Biriyani')
```

```
df['food_category'].unique()
```

```
array(['Burger', 'South Indian', 'Biriyani', 'Pizza', 'Chinese',
       'Desserts'], dtype=object)
```

DATA VISUALIZATION

1️⃣ What is the distribution of delivery times?

2️⃣ Which food categories receive the highest number of orders?

3️⃣ How does customer age relate to delivery ratings?

4️⃣ Which delivery partner has the longest average delivery time?

5️⃣ Are delayed orders (`is_delayed`) associated with lower customer ratings?

6️⃣ What are the top complaint types by frequency?

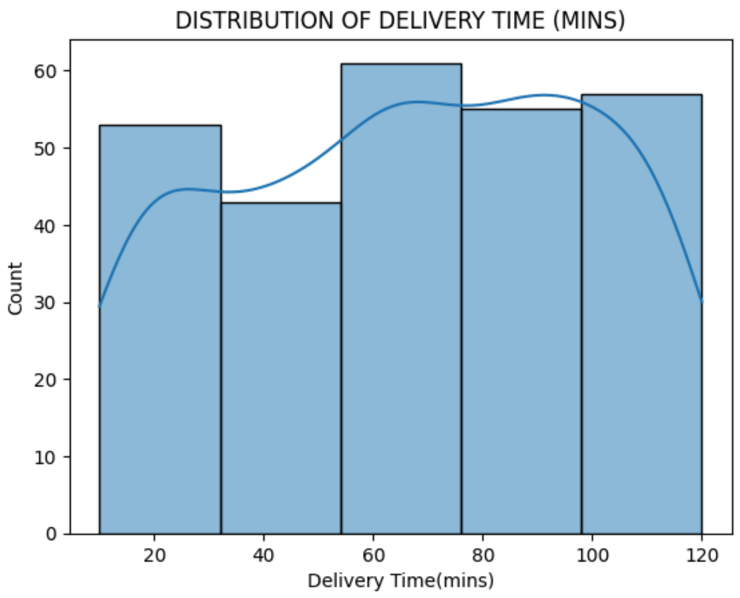
7️⃣ Which cities have the highest number of refunds issued?

8️⃣ How many duplicate orders exist across cities or partners?

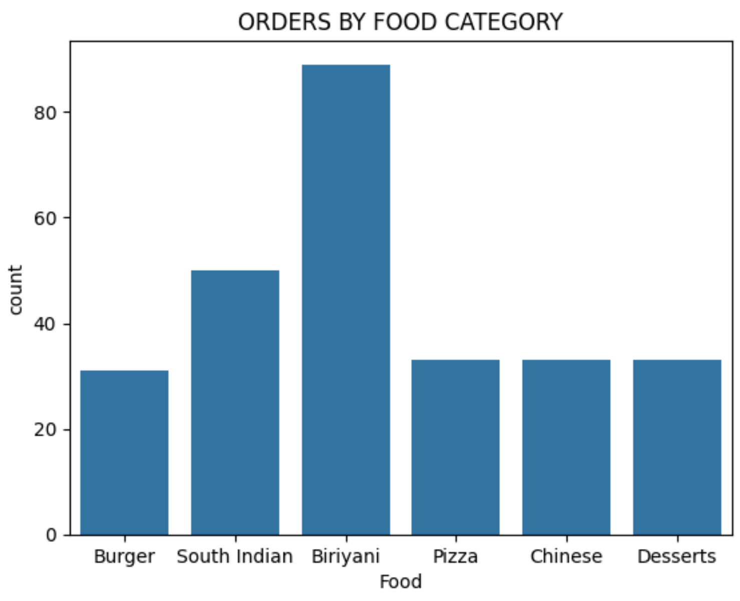
9️⃣ What is the relationship between delivery time and refund amount?

🔟 How do customer ratings vary across different food categories?

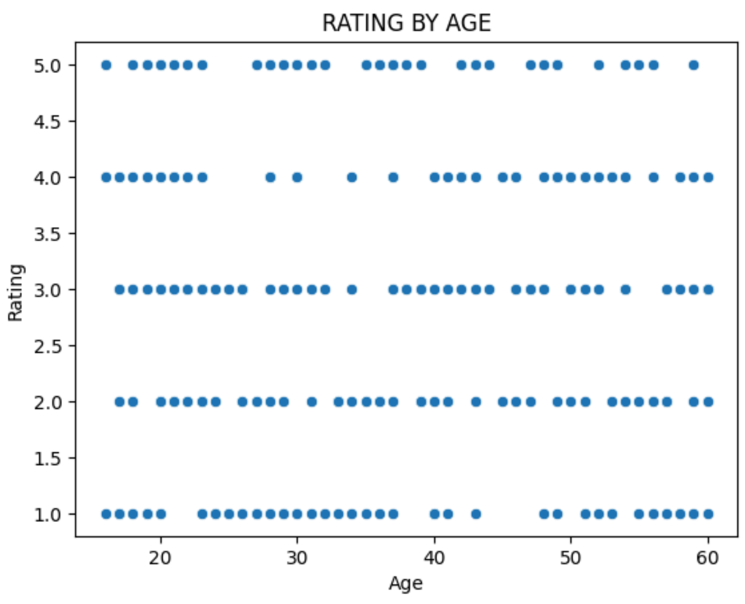
```
import seaborn as sns
import matplotlib.pyplot as plt

sns.histplot(df['delivery_time_mins'],bins=5,kde=True)
plt.title('DISTRIBUTION OF DELIVERY TIME (MINS)')
plt.xlabel('Delivery Time(mins)')
plt.ylabel('Count')
plt.show()
```



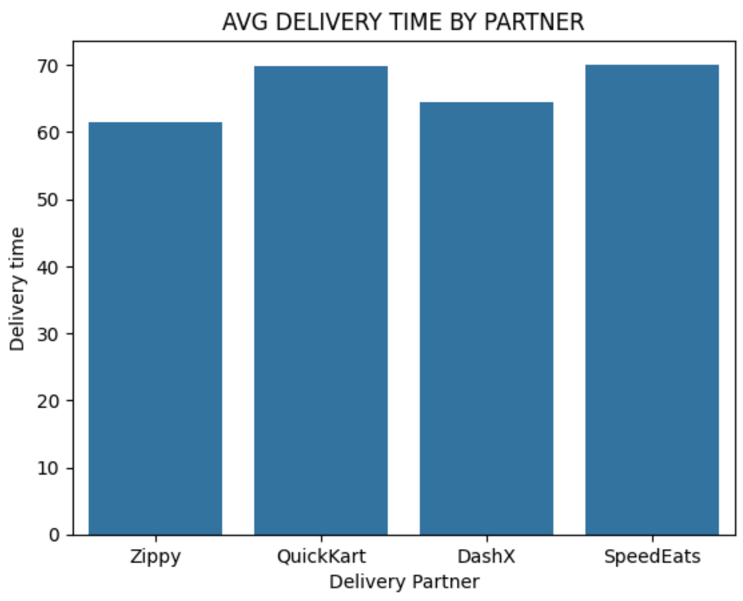
```
sns.countplot(data=df, x='food_category')
plt.xlabel('Food')
plt.title('ORDERS BY FOOD CATEGORY')
plt.show()
```



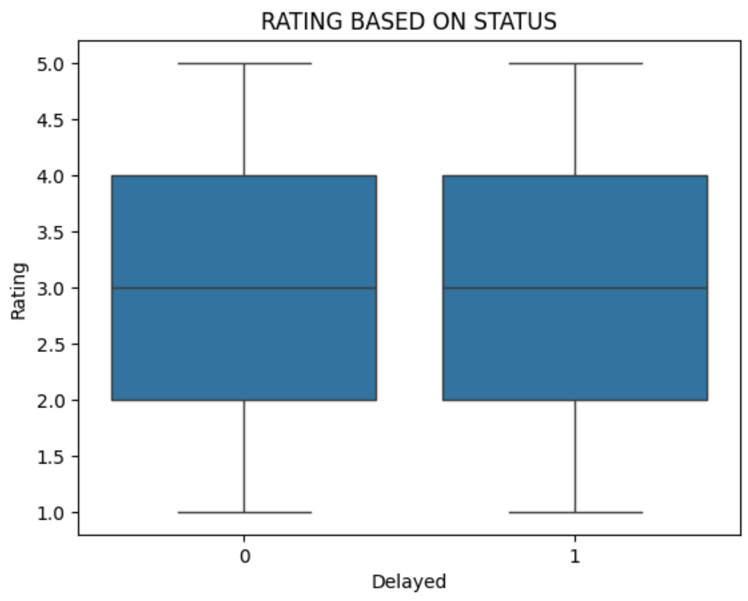
```
sns.scatterplot(data=df, x='age', y='rating')
plt.title('RATING BY AGE')
plt.xlabel('Age')
plt.ylabel('Rating')
plt.show()
```



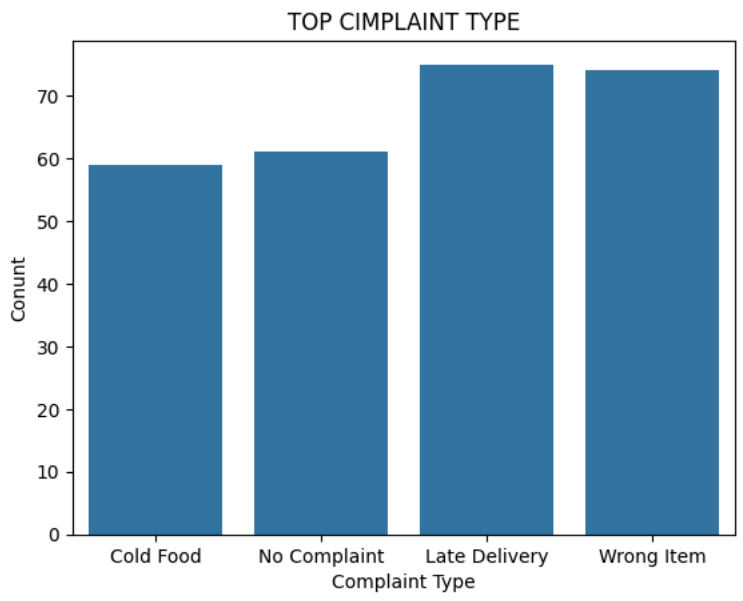
```
sns.barplot(data=df, x='delivery_partner', y='delivery_time_mins',estimator='mean',errorbar=None)
plt.title('AVG DELIVERY TIME BY PARTNER')
plt.xlabel('Delivery Partner')
plt.ylabel('Delivery time')
plt.show()
```



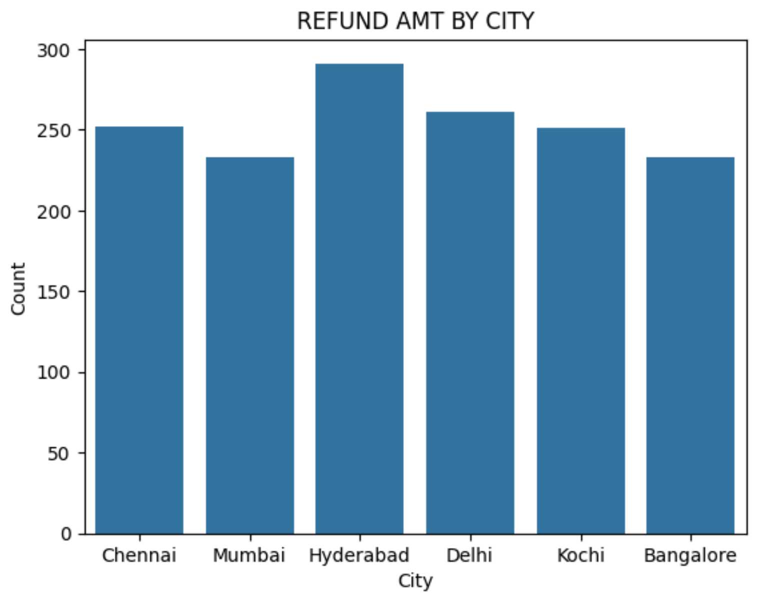
```
sns.boxplot(data=df, x='is_delayed', y='rating')
plt.title('RATING BASED ON STATUS')
plt.xlabel('Delayed')
plt.ylabel('Rating')
plt.show()
```



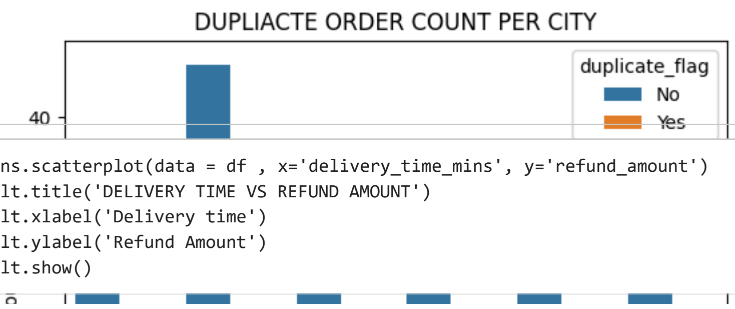
```
sns.countplot(data=df, x='complaint')
plt.title('TOP CIMPLAINT TYPE')
plt.xlabel('Complaint Type')
plt.ylabel('Conunt')
plt.show()
```



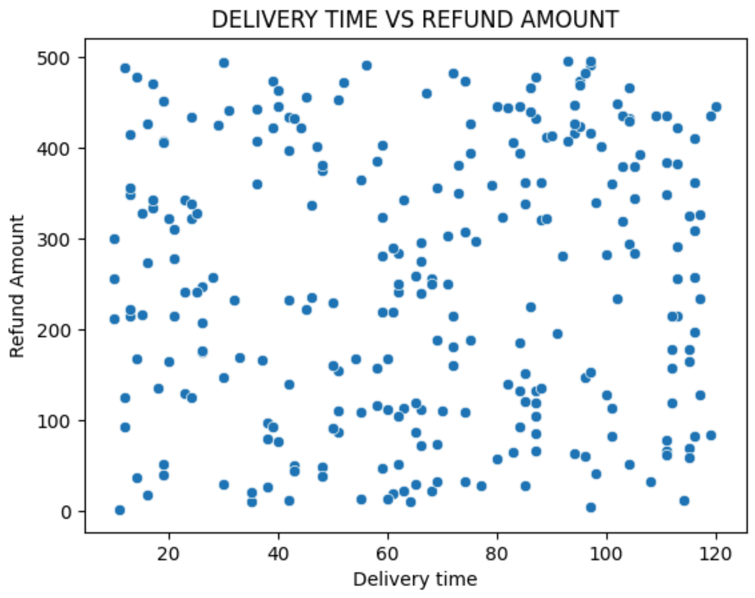
```
sns.barplot(data=df, x='city', y='refund_amount',errorbar=None)
plt.title('REFUND AMT BY CITY')
plt.xlabel('City')
plt.ylabel('Count')
plt.show()
```



```
sns.countplot(data=df, x='city', hue='duplicate_flag')
plt.title('DUPLIACTE ORDER COUNT PER CITY')

plt.show()
```



```
sns.scatterplot(data = df , x='delivery_time_mins', y='refund_amount')
plt.title('DELIVERY TIME VS REFUND AMOUNT')
plt.xlabel('Delivery time')
plt.ylabel('Refund Amount')
plt.show()
```



```
sns.boxplot(data=df, x='food_category', y='rating')
plt.title("RATING DISTRIBUTION ACROSS FOOD CATEGORIES")
plt.xlabel("Food Category")
plt.ylabel("Customer Rating")
plt.show()
```