EXPLORATORY DATA ANALYTICS - FOOD ORDER AND DELIVERY

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

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

```
df=pd.read_csv("/content/food_order.csv")
df
```

	order_id	customer_id	restaurant_name	cuisine_type	cost_of_the_order	day
0	1477147	337525	Hangawi	Korean	30.75	
1	1477685	358141	Blue Ribbon Sushi Izakaya	Japanese	12.08	
2	1477070	66393	Cafe Habana	Mexican	12.23	
3	1477334	106968	Blue Ribbon Fried Chicken	American	29.20	
4	1478249	76942	Dirty Bird to Go	American	11.59	
...	...	...	...	...	...	
1893	1476701	292602	Chipotle Mexican Grill $1.99 Delivery	Mexican	22.31	
1894	1477421	397537	The Smile	American	12.18	
1895	1477819	35309	Blue Ribbon Sushi	Japanese	25.22	

```
df.isna().sum()
```

```
order_id                0
customer_id             0
restaurant_name         0
cuisine_type            0
cost_of_the_order       0
day_of_the_week         0
rating                  0
food_preparation_time   0
delivery_time           0
dtype: int64
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1898 entries, 0 to 1897
Data columns (total 9 columns):
 #   Column                 Non-Null Count  Dtype  
---  ------                 --------------  -----  
 0   order_id               1898 non-null   int64  
 1   customer_id            1898 non-null   int64  
 2   restaurant_name        1898 non-null   object 
 3   cuisine_type           1898 non-null   object 
 4   cost_of_the_order      1898 non-null   float64
 5   day_of_the_week        1898 non-null   object 
 6   rating                 1898 non-null   object 
 7   food_preparation_time  1898 non-null   int64  
 8   delivery_time          1898 non-null   int64  
dtypes: float64(1), int64(4), object(4)
memory usage: 133.6+ KB
```

```
df.describe()
```

	order_id	customer_id	cost_of_the_order	food_preparation_time	deliv
count	1.898000e+03	1898.000000	1898.000000	1898.000000	18
mean	1.477496e+06	171168.478398	16.498851	27.371970	
std	5.480497e+02	113698.139743	7.483812	4.632481	
min	1.476547e+06	1311.000000	4.470000	20.000000	
25%	1.477021e+06	77787.750000	12.080000	23.000000	
50%	1.477496e+06	128600.000000	14.140000	27.000000	

```
df["rating"].value_counts()
```

```
Not given    736
5            588
4            386
3            188
Name: rating, dtype: int64
```

```
df['rating_new'] = df['rating'].replace('Not given', np.nan).astype(float)
df['rating_new'] = df['rating_new'].fillna(value = df.groupby(['restaurant_name'])['rating_new'
```

```
df.isna().sum()
```

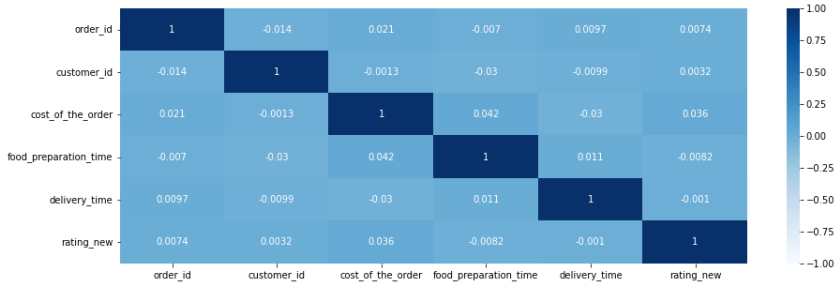
```
order_id                  0
customer_id               0
restaurant_name           0
cuisine_type              0
cost_of_the_order         0
day_of_the_week           0
rating                    0
food_preparation_time     0
delivery_time             0
rating_new               30
dtype: int64
```

```
df.duplicated().sum()
```

```
0
```

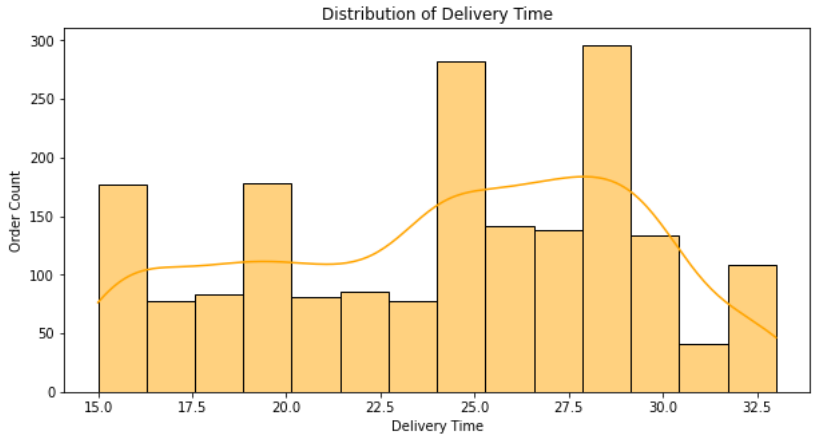
HEAT MAP

```
plt.figure(figsize=(15,5))
sns.heatmap(df.corr(),annot=True,cmap="Blues",vmin=-1,vmax=1)
plt.show()
```



DISTRIBUTION OF DELIVERY TIME

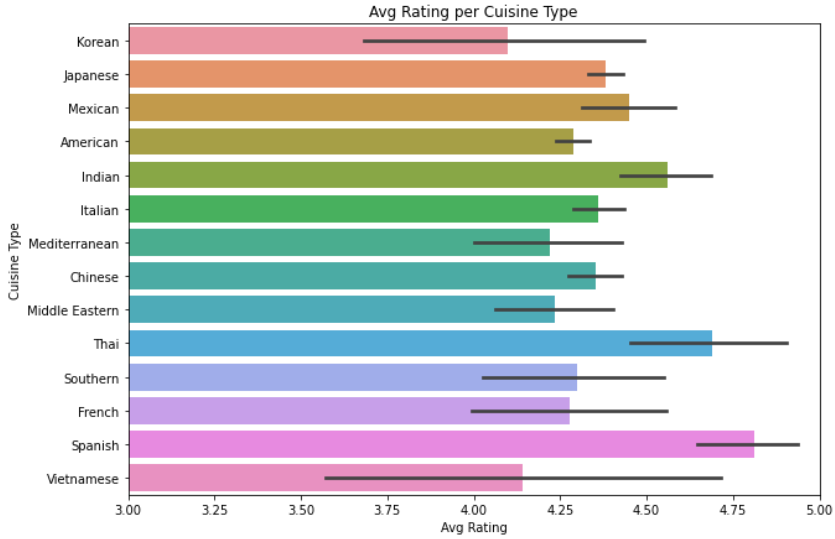
```
plt.figure(figsize=(10,5))
sns.histplot(data=df, x='delivery_time',kde=True, color='orange')
plt.xlabel('Delivery Time')
plt.ylabel('Order Count')
plt.title('Distribution of Delivery Time')
plt.show()
```




```
plt.figure(figsize=(10,5))
sns.boxplot(data=df, x='delivery_time', color='red')
plt.xlabel('Delivery Time')
plt.title('Boxplot of Delivery Time')
plt.show()
```

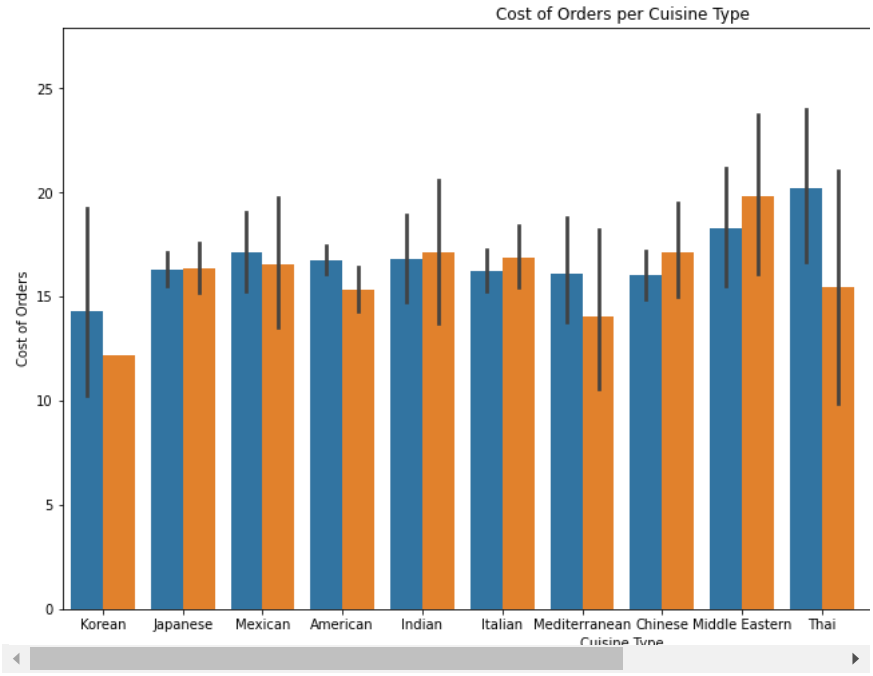
AVERAGE RATING PER CUISINE TYPE

```
plt.figure(figsize=(10,7))
sns.barplot(data=df, y='cuisine_type', x='rating_new')
plt.xlabel('Avg Rating')
plt.ylabel('Cuisine Type')
plt.xlim(3,5)
plt.title('Avg Rating per Cuisine Type')
plt.show()
```



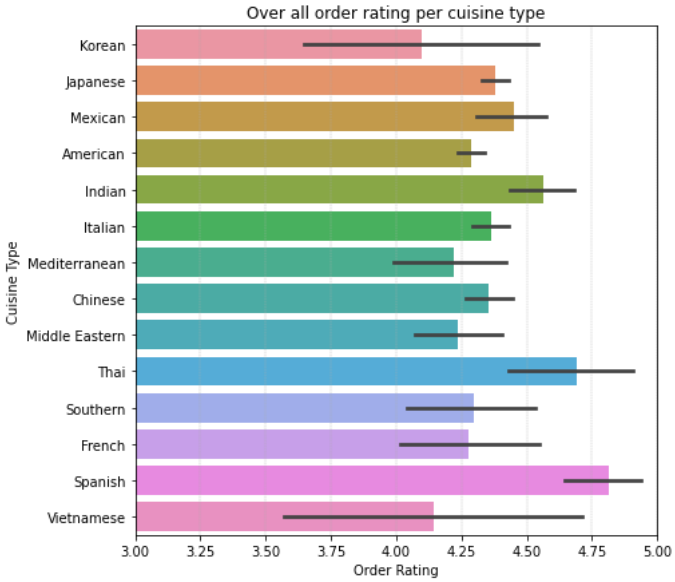
AVERAGE COST PER CUISINE TYPE

```
plt.figure(figsize=(15,8))
sns.barplot(data=df, y='cost_of_the_order', x='cuisine_type', hue='day_of_the_week')
plt.ylabel('Cost of Orders')
plt.xlabel('Cuisine Type')
plt.title('Cost of Orders per Cuisine Type')
plt.show()
```



OVER ALL ORDER RATING PER CUISINE TYPE

```
plt.figure(figsize=(7,7))
sns.barplot(data=df,x="rating_new",y="cuisine_type")
plt.xlabel("Order Rating")
plt.ylabel("Cuisine Type")
plt.xlim(3,5)
plt.title("Over all order rating per cuisine type")
plt.grid(axis="x",linestyle="--",linewidth=0.4)
plt.show()
```



✓ 1s completed at 12:00 PM