

Aim

- To Analyse the Airline Passenger Satisfaction dataset using Python and to obtain an insight into the factors lead to customer satisfaction for an Airline.

Import required python libraries

In [105...

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

Read the dataset using pandas

In [4]:

```
# Read the file

df=pd.read_csv('D:/project/csv_files/test.csv')
```

Show top 10 records

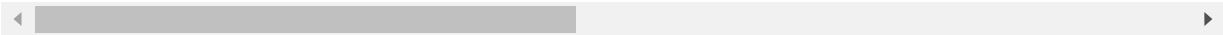
In [5]:

```
df.head(10)
```

Out[5]:

	Unnamed: 0	id	Gender	Customer Type	Age	Type of Travel	Class	Flight Distance	Inflight wifi service	Departure/Arrival time convenient
0	0	19556	Female	Loyal Customer	52	Business travel	Eco	160	5	
1	1	90035	Female	Loyal Customer	36	Business travel	Business	2863	1	
2	2	12360	Male	disloyal Customer	20	Business travel	Eco	192	2	
3	3	77959	Male	Loyal Customer	44	Business travel	Business	3377	0	
4	4	36875	Female	Loyal Customer	49	Business travel	Eco	1182	2	
5	5	39177	Male	Loyal Customer	16	Business travel	Eco	311	3	
6	6	79433	Female	Loyal Customer	77	Business travel	Business	3987	5	
7	7	97286	Female	Loyal Customer	43	Business travel	Business	2556	2	
8	8	27508	Male	Loyal Customer	47	Business travel	Eco	556	5	
9	9	62482	Female	Loyal Customer	46	Business travel	Business	1744	2	

10 rows × 25 columns



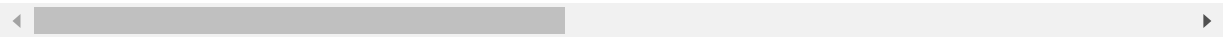
Last 5 records

```
In [7]: df.tail()
```

Out[7]:

	Unnamed: 0	id	Gender	Customer Type	Age	Type of Travel	Class	Flight Distance	Inflight wifi service	Departure, time con
25971	25971	78463	Male	disloyal Customer	34	Business travel	Business	526	3	
25972	25972	71167	Male	Loyal Customer	23	Business travel	Business	646	4	
25973	25973	37675	Female	Loyal Customer	17	Personal Travel	Eco	828	2	
25974	25974	90086	Male	Loyal Customer	14	Business travel	Business	1127	3	
25975	25975	34799	Female	Loyal Customer	42	Personal Travel	Eco	264	2	

5 rows × 25 columns



Number of Rows and Columns of data

```
In [9]: df.shape
```

Out[9]: (25976, 25)

Check for the null data

```
In [11]: df.isna().sum()
```

Out[11]:

Unnamed: 0	0
id	0
Gender	0
Customer Type	0
Age	0
Type of Travel	0
Class	0
Flight Distance	0
Inflight wifi service	0
Departure/Arrival time convenient	0
Ease of Online booking	0
Gate location	0
Food and drink	0
Online boarding	0
Seat comfort	0
Inflight entertainment	0
On-board service	0
Leg room service	0
Baggage handling	0
Checkin service	0
Inflight service	0
Cleanliness	0
Departure Delay in Minutes	0
Arrival Delay in Minutes	83
satisfaction	0
dtype: int64	

Basic Information about columns

```
In [12]: df.info()
```

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 25976 entries, 0 to 25975

Data columns (total 25 columns):

#	Column	Non-Null Count	Dtype
0	Unnamed: 0	25976 non-null	int64
1	id	25976 non-null	int64
2	Gender	25976 non-null	object
3	Customer Type	25976 non-null	object
4	Age	25976 non-null	int64
5	Type of Travel	25976 non-null	object
6	Class	25976 non-null	object
7	Flight Distance	25976 non-null	int64
8	Inflight wifi service	25976 non-null	int64
9	Departure/Arrival time convenient	25976 non-null	int64
10	Ease of Online booking	25976 non-null	int64
11	Gate location	25976 non-null	int64
12	Food and drink	25976 non-null	int64
13	Online boarding	25976 non-null	int64
14	Seat comfort	25976 non-null	int64
15	Inflight entertainment	25976 non-null	int64
16	On-board service	25976 non-null	int64
17	Leg room service	25976 non-null	int64
18	Baggage handling	25976 non-null	int64
19	Checkin service	25976 non-null	int64
20	Inflight service	25976 non-null	int64
21	Cleanliness	25976 non-null	int64
22	Departure Delay in Minutes	25976 non-null	int64
23	Arrival Delay in Minutes	25893 non-null	float64
24	satisfaction	25976 non-null	object

dtypes: float64(1), int64(19), object(5)

memory usage: 5.0+ MB

In [13]: *# Show the column names*

```
df.columns
```

Out[13]: Index(['Unnamed: 0', 'id', 'Gender', 'Customer Type', 'Age', 'Type of Travel', 'Class', 'Flight Distance', 'Inflight wifi service', 'Departure/Arrival time convenient', 'Ease of Online booking', 'Gate location', 'Food and drink', 'Online boarding', 'Seat comfort', 'Inflight entertainment', 'On-board service', 'Leg room service', 'Baggage handling', 'Checkin service', 'Inflight service', 'Cleanliness', 'Departure Delay in Minutes', 'Arrival Delay in Minutes', 'satisfaction'], dtype='object')

In [19]: *# Drop the Customer Id Column*

```
df.drop('id',axis=1,inplace=True)
```

In [21]: *# Query - What are the different classes of customers*

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

Out[21]: array(['Eco', 'Business', 'Eco Plus'], dtype=object)

In [22]: *# Query - What are the different Types of Travel*

```
df['Type of Travel'].unique()
```

Out[22]: array(['Business travel', 'Personal Travel'], dtype=object)

In [23]: *# Query - What are the different Types of customers*

```
df['Customer Type'].unique()
```

```
Out[23]: array(['Loyal Customer', 'disloyal Customer'], dtype=object)
```

What is the Average Customer Rating for Inflight Services ?

```
In [24]: df['Inflight service'].mean()
```

```
Out[24]: 3.649253156760086
```

What is the Average Customer Rating for Ease of Online Booking ?

```
In [103... df['Ease of Online booking'].mean()
```

```
Out[103... 2.756775485063135
```

Average Customer Flight Distance

```
In [28]: print('Average Flight Distance by Customers: ',df['Flight Distance'].mean())
```

```
Average Flight Distance by Customers: 1193.788458577148
```

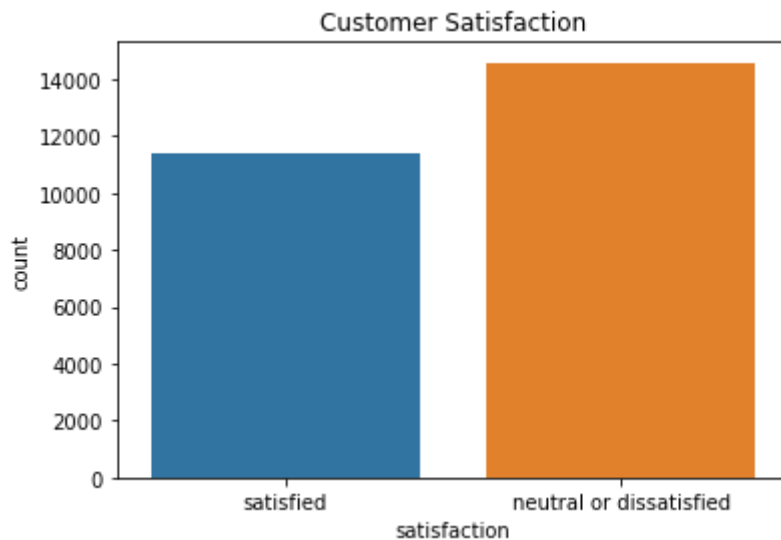
Basic Statistical Information about the dataset

```
In [104... df.describe()
```

```
Out[104...
```

	Age	Flight Distance	Inflight wifi service	Departure/Arrival time convenient	Ease of Online booking	Gate location
count	25976.000000	25976.000000	25976.000000	25976.000000	25976.000000	25976.000000
mean	39.620958	1193.788459	2.724746	3.046812	2.756775	2.977094
std	15.135685	998.683999	1.335384	1.533371	1.412951	1.282133
min	7.000000	31.000000	0.000000	0.000000	0.000000	1.000000
25%	27.000000	414.000000	2.000000	2.000000	2.000000	2.000000
50%	40.000000	849.000000	3.000000	3.000000	3.000000	3.000000
75%	51.000000	1744.000000	4.000000	4.000000	4.000000	4.000000
max	85.000000	4983.000000	5.000000	5.000000	5.000000	5.000000

```
In [33]: sns.countplot(x=df.satisfaction)
plt.title('Customer Satisfaction')
plt.show()
```



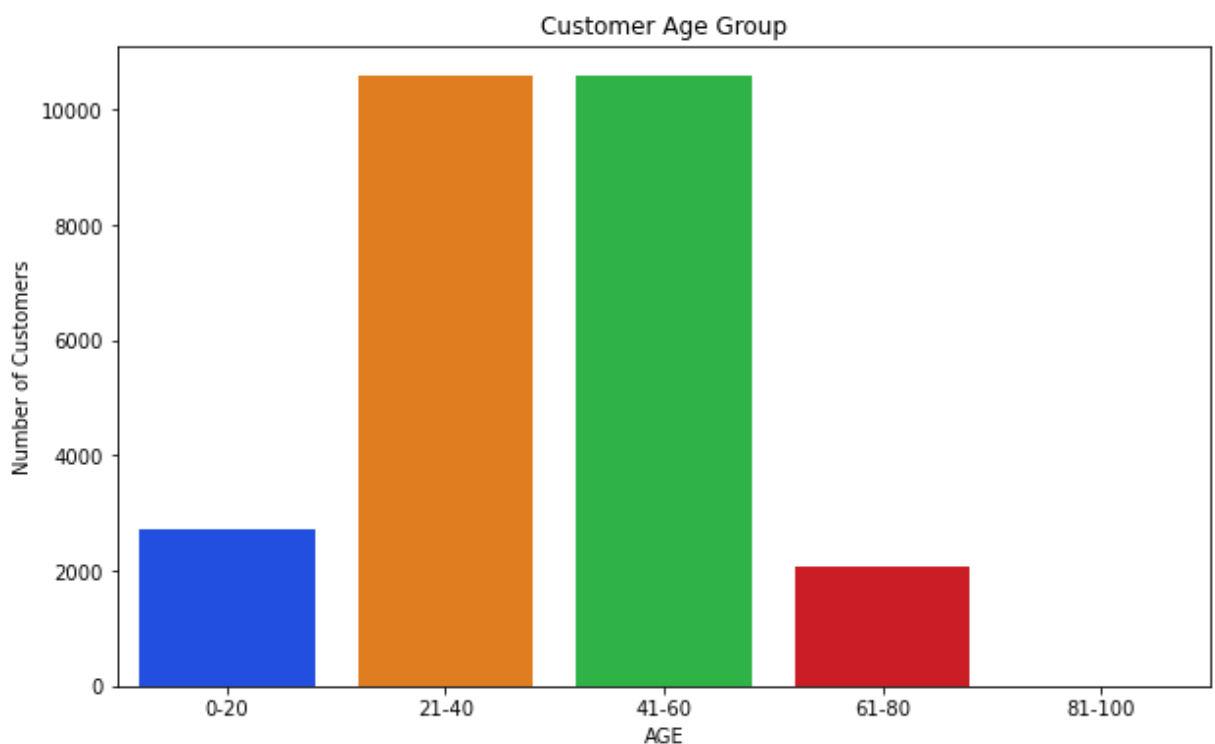
```
In [35]: print(df.Age.max(),df.Age.min())
```

85 7

```
In [42]: age1_20 = df.Age[(df.Age >=1) & (df.Age <=20)]
age21_40 = df.Age[(df.Age >=21) & (df.Age <=40)]
age41_60 = df.Age[(df.Age >=41) & (df.Age <=60)]
age61_80 = df.Age[(df.Age >=61) & (df.Age <=80)]
age81_100 = df.Age[(df.Age >=81) & (df.Age <=100)]

age_x=['0-20', '21-40', '41-60', '61-80', '81-100']
age_y=[len(age1_20.values),len(age21_40.values),len(age41_60.values),len(age61_80.va

plt.figure(figsize=(10,6))
sns.barplot(x=age_x,y=age_y,palette='bright')
plt.title('Customer Age Group')
plt.xlabel('AGE')
plt.ylabel('Number of Customers')
plt.show()
```



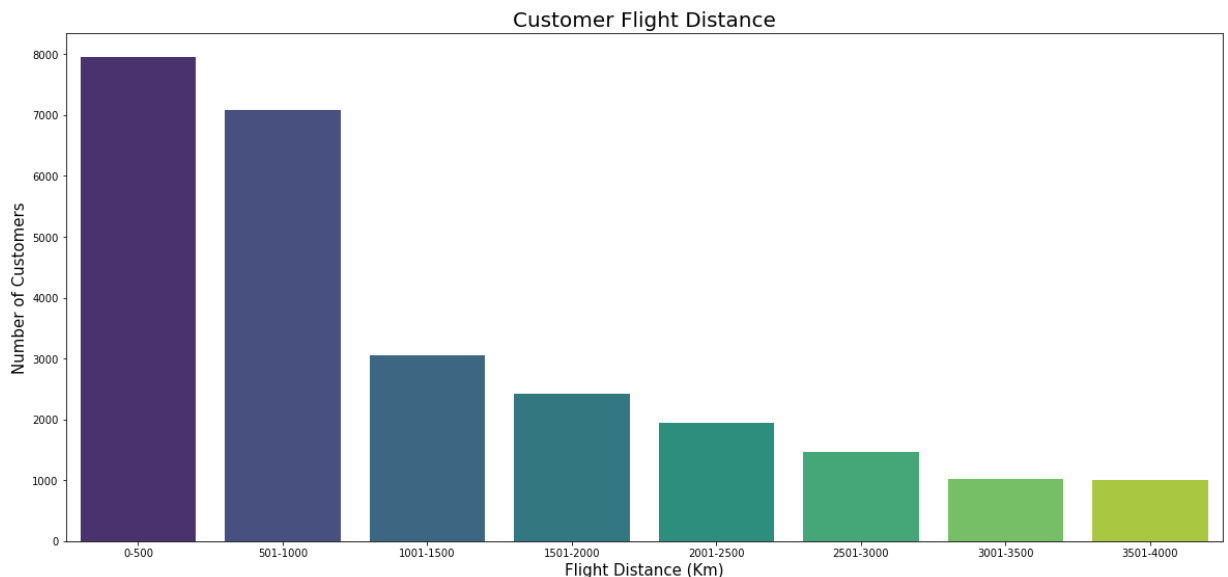
```
In [43]: print(df['Flight Distance'].min(),df['Flight Distance'].max())
```

31 4983

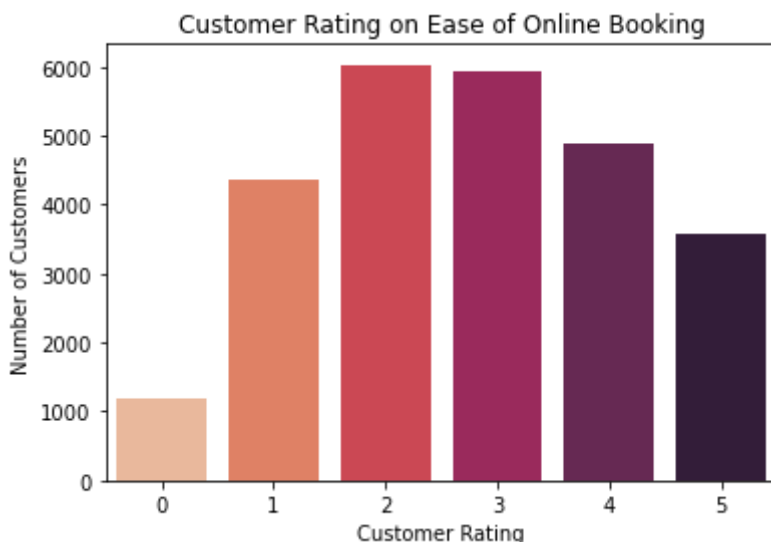
```
In [54]: f1_500 = df['Flight Distance'][(df['Flight Distance'] >=1) & (df['Flight Distance'] < 501)]
f501_1000 = df['Flight Distance'][(df['Flight Distance'] >=501) & (df['Flight Distance'] < 1001)]
f1001_1500 = df['Flight Distance'][(df['Flight Distance'] >=1001) & (df['Flight Distance'] < 1501)]
f1501_2000 = df['Flight Distance'][(df['Flight Distance'] >=1501) & (df['Flight Distance'] < 2001)]
f2001_2500 = df['Flight Distance'][(df['Flight Distance'] >=2001) & (df['Flight Distance'] < 2501)]
f2501_3000 = df['Flight Distance'][(df['Flight Distance'] >=2501) & (df['Flight Distance'] < 3001)]
f3001_3500 = df['Flight Distance'][(df['Flight Distance'] >=3001) & (df['Flight Distance'] < 3501)]
f3501_4000 = df['Flight Distance'][(df['Flight Distance'] >=3501) & (df['Flight Distance'] < 4001)]
f4001_4500 = df['Flight Distance'][(df['Flight Distance'] >=4001) & (df['Flight Distance'] < 4501)]
f4501_5000 = df['Flight Distance'][(df['Flight Distance'] >=4501) & (df['Flight Distance'] < 5001)]

fl_x = ['0-500', '501-1000', '1001-1500', '1501-2000', '2001-2500', '2501-3000', '3001-3500', '3501-4000']
fl_y = [len(f1_500.values), len(f501_1000.values), len(f1001_1500.values), len(f1501_2000.values), len(f2001_2500.values), len(f2501_3000.values), len(f3001_3500.values), len(f3501_4000.values)]

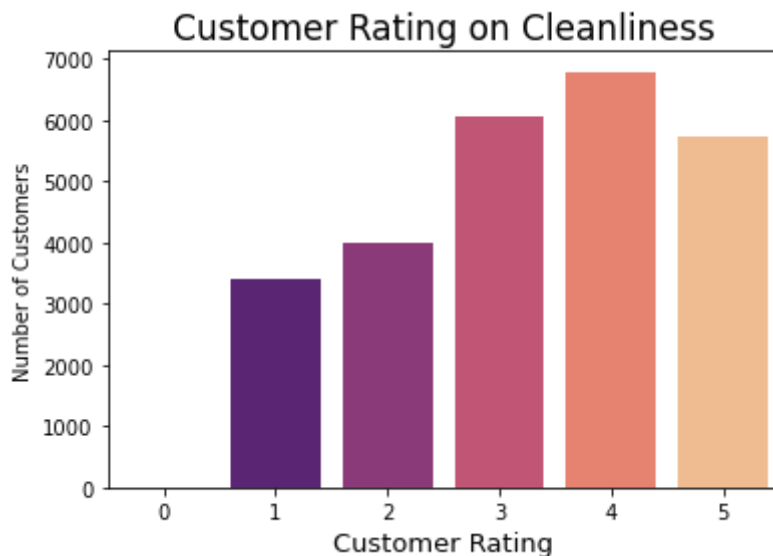
plt.figure(figsize=(20,9))
sns.barplot(x=fl_x,y=fl_y,palette='viridis')
plt.title('Customer Flight Distance',size=20)
plt.xlabel('Flight Distance (Km)',size=15)
plt.ylabel('Number of Customers',size=15)
plt.show()
```



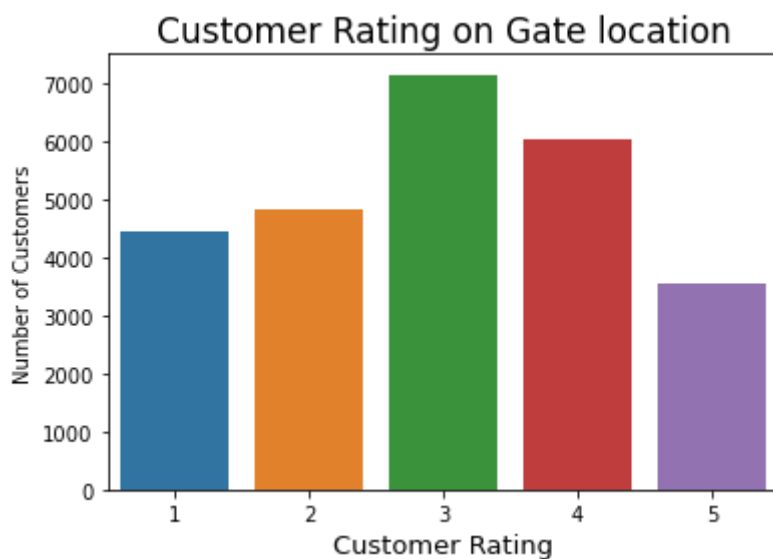
```
In [60]: sns.countplot(x=df['Ease of Online booking'],palette='rocket_r')
plt.title('Customer Rating on Ease of Online Booking')
plt.ylabel('Number of Customers')
plt.xlabel("Customer Rating")
plt.show()
```



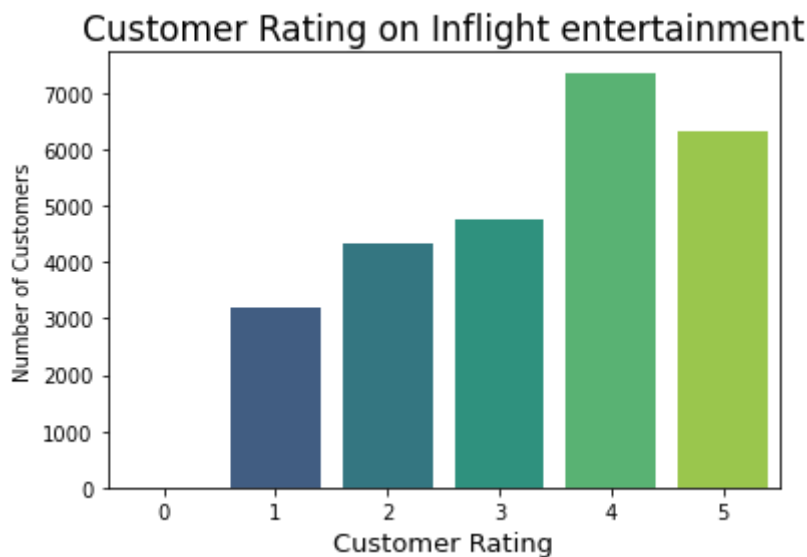
```
In [65]: sns.countplot(x=df['Cleanliness'],palette='magma')
plt.title('Customer Rating on Cleanliness',size=17)
plt.ylabel('Number of Customers',size=10)
plt.xlabel("Customer Rating",size=13)
plt.show()
```



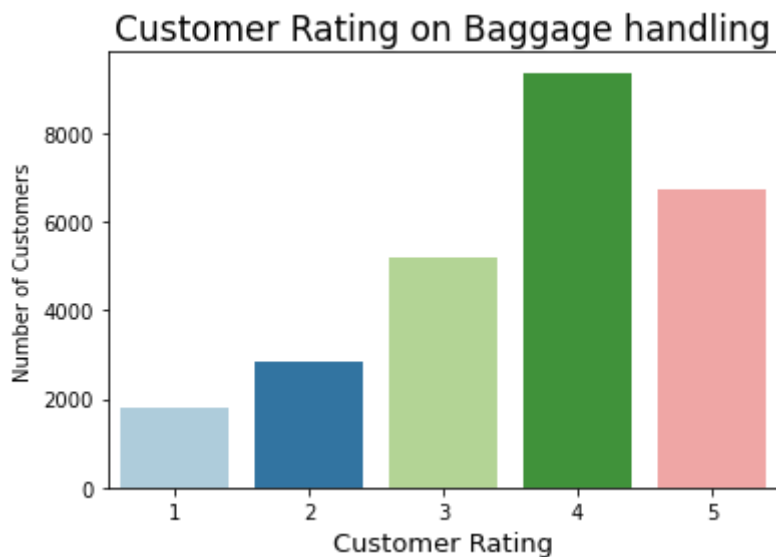
```
In [69]: sns.countplot(x=df['Gate location'])
plt.title('Customer Rating on Gate location',size=17)
plt.ylabel('Number of Customers',size=10)
plt.xlabel("Customer Rating",size=13)
plt.show()
```



```
In [71]: sns.countplot(x=df['Inflight entertainment'],palette='viridis')
plt.title('Customer Rating on Inflight entertainment',size=17)
plt.ylabel('Number of Customers',size=10)
plt.xlabel("Customer Rating",size=13)
plt.show()
```



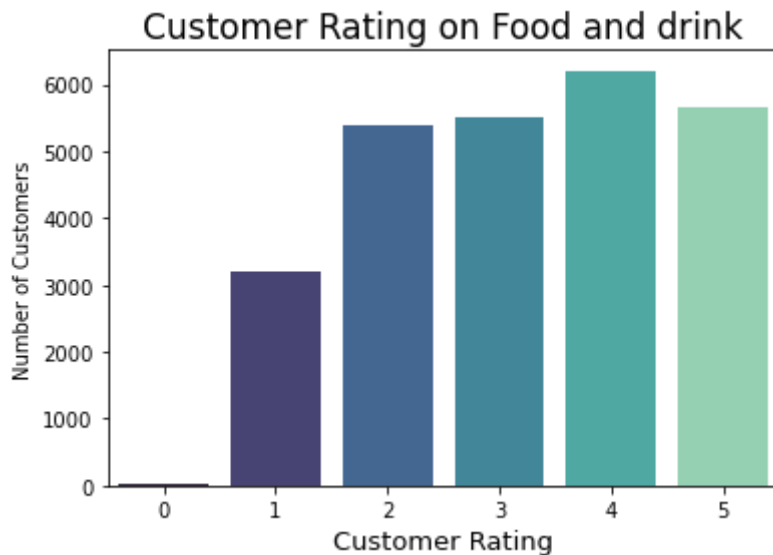
```
In [72]: sns.countplot(x=df['Baggage handling'],palette='Paired')
plt.title('Customer Rating on Baggage handling',size=17)
plt.ylabel('Number of Customers',size=10)
plt.xlabel("Customer Rating",size=13)
plt.show()
```



```
In [73]: sns.countplot(x=df['Checkin service'],palette='crest')
plt.title('Customer Rating on Checkin service',size=17)
plt.ylabel('Number of Customers',size=10)
plt.xlabel("Customer Rating",size=13)
plt.show()
```




```
In [74]: sns.countplot(x=df['Food and drink'],palette='mako')
plt.title('Customer Rating on Food and drink',size=17)
plt.ylabel('Number of Customers',size=10)
plt.xlabel("Customer Rating",size=13)
plt.show()
```



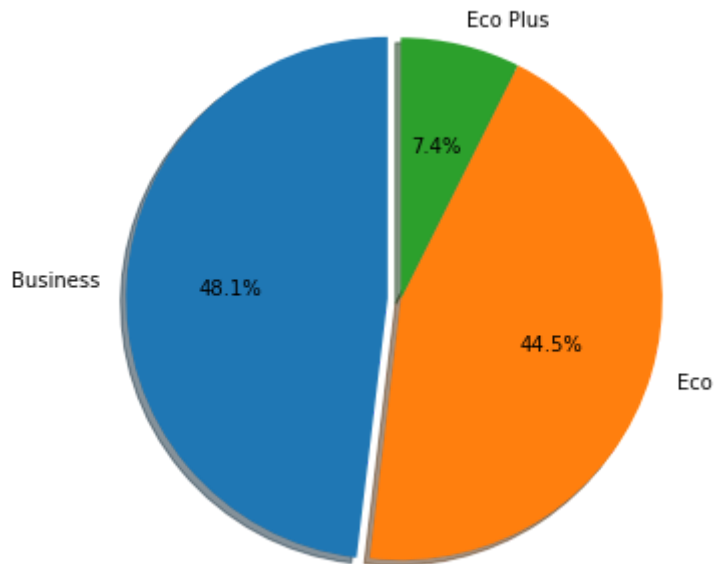
```
In [82]: pie=df.groupby('Class').count().reset_index()
```

```
In [102... slices=list(pie['Customer Type'])
name=list(pie['Class'])

plt.figure(figsize=(10,6))
plt.pie(slices, labels=name, startangle=90,autopct='%1.1f%%',shadow=True,explode=[0.
plt.title('Different Classes of Customers',size=15)

plt.show()
```

Different Classes of Customers



Conclusion

- This is the Analysis of Airline Passenger Satisfaction dataset using Python. I have used Seaborn Library to visualize the data. Different figures were plotted to obtain the customer satisfaction based on the different ratings given by customers. This analysis was done as a part of Mini Project for Data Analytics Course offered by EduBridge.

In []:

In []: