

```
In [1]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn import preprocessing
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

```
In [2]: df=pd.read_csv(r"C:\Users\LENOVO\Downloads\used_cars_data.csv")
df
```

Out[2]:

	S.No.	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_1
0	0	Maruti Wagon R LXI CNG	Mumbai	2010	72000	CNG	Manual	
1	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	
2	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	
3	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	
4	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Sec
...
7248	7248	Volkswagen Vento Diesel Trendline	Hyderabad	2011	89411	Diesel	Manual	
7249	7249	Volkswagen Polo GT TSI	Mumbai	2015	59000	Petrol	Automatic	
7250	7250	Nissan Micra Diesel XV	Kolkata	2012	28000	Diesel	Manual	
7251	7251	Volkswagen Polo GT TSI	Pune	2013	52262	Petrol	Automatic	T
7252	7252	Mercedes-Benz E-Class 2009-2013 E 220 CDI Avan...	Kochi	2014	72443	Diesel	Automatic	

7253 rows × 14 columns



In [3]: `df.head(10)`

Out[3]:

	S.No.	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type
0	0	Maruti Wagon R LXI CNG	Mumbai	2010	72000	CNG	Manual	First
1	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	First
2	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First
3	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	First
4	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Second
5	5	Hyundai EON LPG Era Plus Option	Hyderabad	2012	75000	LPG	Manual	First
6	6	Nissan Micra Diesel XV	Jaipur	2013	86999	Diesel	Manual	First
7	7	Toyota Innova Crysta 2.8 GX AT 8S	Mumbai	2016	36000	Diesel	Automatic	First
8	8	Volkswagen Vento Diesel Comfortline	Pune	2013	64430	Diesel	Manual	First
9	9	Tata Indica Vista Quadrajet LS	Chennai	2012	65932	Diesel	Manual	Second

```
In [4]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7253 entries, 0 to 7252
Data columns (total 14 columns):
#   Column                Non-Null Count  Dtype
---  -
0   S.No.                 7253 non-null  int64
1   Name                  7253 non-null  object
2   Location              7253 non-null  object
3   Year                  7253 non-null  int64
4   Kilometers_Driven     7253 non-null  int64
5   Fuel_Type             7253 non-null  object
6   Transmission          7253 non-null  object
7   Owner_Type            7253 non-null  object
8   Mileage               7251 non-null  object
9   Engine                7207 non-null  object
10  Power                 7207 non-null  object
11  Seats                 7200 non-null  float64
12  New_Price             1006 non-null  object
13  Price                 6019 non-null  float64
dtypes: float64(2), int64(3), object(9)
memory usage: 793.4+ KB
```

In [5]: df.describe

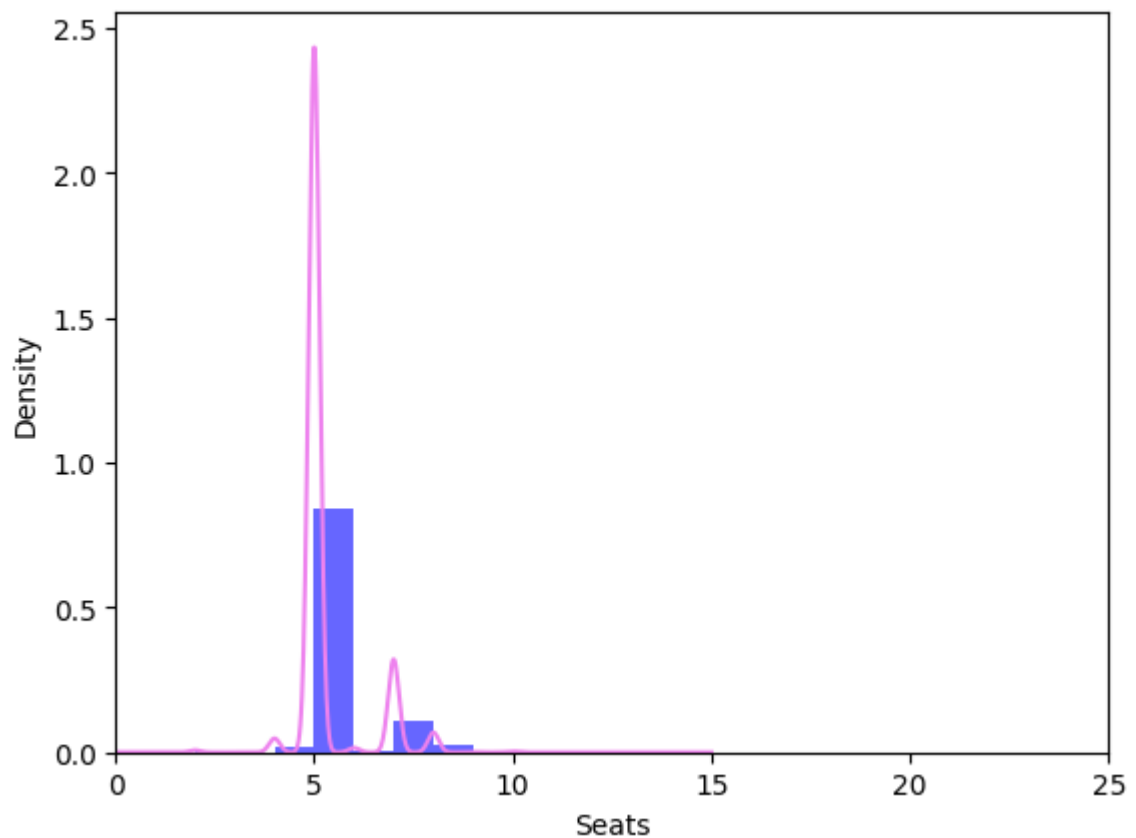
```
Out[5]: <bound method NDFrame.describe of          S.No.
Name      Location
0          0          Maruti Wagon R LXI CNG      Mumbai \
1          1      Hyundai Creta 1.6 CRDi SX Option      Pune
2          2          Honda Jazz V      Chennai
3          3      Maruti Ertiga VDI      Chennai
4          4      Audi A4 New 2.0 TDI Multitronic      Coimbatore
...      ...      ...      ...
7248      7248      Volkswagen Vento Diesel Trendline      Hyderabad
7249      7249      Volkswagen Polo GT TSI      Mumbai
7250      7250      Nissan Micra Diesel XV      Kolkata
7251      7251      Volkswagen Polo GT TSI      Pune
7252      7252      Mercedes-Benz E-Class 2009-2013 E 220 CDI Avan...      Kochi

      Year  Kilometers_Driven  Fuel_Type  Transmission  Owner_Type      Mileage
0      2010          72000          CNG          Manual          First      26.6 km/kg
\
1      2015          41000          Diesel          Manual          First      19.67 kmpl
2      2011          46000          Petrol          Manual          First      18.2 kmpl
3      2012          87000          Diesel          Manual          First      20.77 kmpl
4      2013          40670          Diesel          Automatic        Second      15.2 kmpl
...      ...      ...      ...      ...      ...
7248      2011          89411          Diesel          Manual          First      20.54 kmpl
7249      2015          59000          Petrol          Automatic        First      17.21 kmpl
7250      2012          28000          Diesel          Manual          First      23.08 kmpl
7251      2013          52262          Petrol          Automatic        Third      17.2 kmpl
7252      2014          72443          Diesel          Automatic        First      10.0 kmpl

      Engine      Power  Seats  New_Price  Price
0      998 CC      58.16 bhp      5.0      NaN      1.75
1      1582 CC      126.2 bhp      5.0      NaN      12.50
2      1199 CC      88.7 bhp      5.0      8.61 Lakh      4.50
3      1248 CC      88.76 bhp      7.0      NaN      6.00
4      1968 CC      140.8 bhp      5.0      NaN      17.74
...      ...      ...      ...      ...      ...
7248      1598 CC      103.6 bhp      5.0      NaN      NaN
7249      1197 CC      103.6 bhp      5.0      NaN      NaN
7250      1461 CC      63.1 bhp      5.0      NaN      NaN
7251      1197 CC      103.6 bhp      5.0      NaN      NaN
7252      2148 CC      170 bhp      5.0      NaN      NaN
```

[7253 rows x 14 columns]>

```
In [6]: ax=df["Seats"].hist(bins=10,density=True,stacked=True,color='blue',alpha=0.6)
df['Seats'].plot(kind='density',color='violet')
ax.set(xlabel='Seats')
plt.xlim(-0,25)
plt.show()
```



```
In [7]: print(df["Seats"].mean(skipna=True))
print(df["Seats"].median(skipna=True))
```

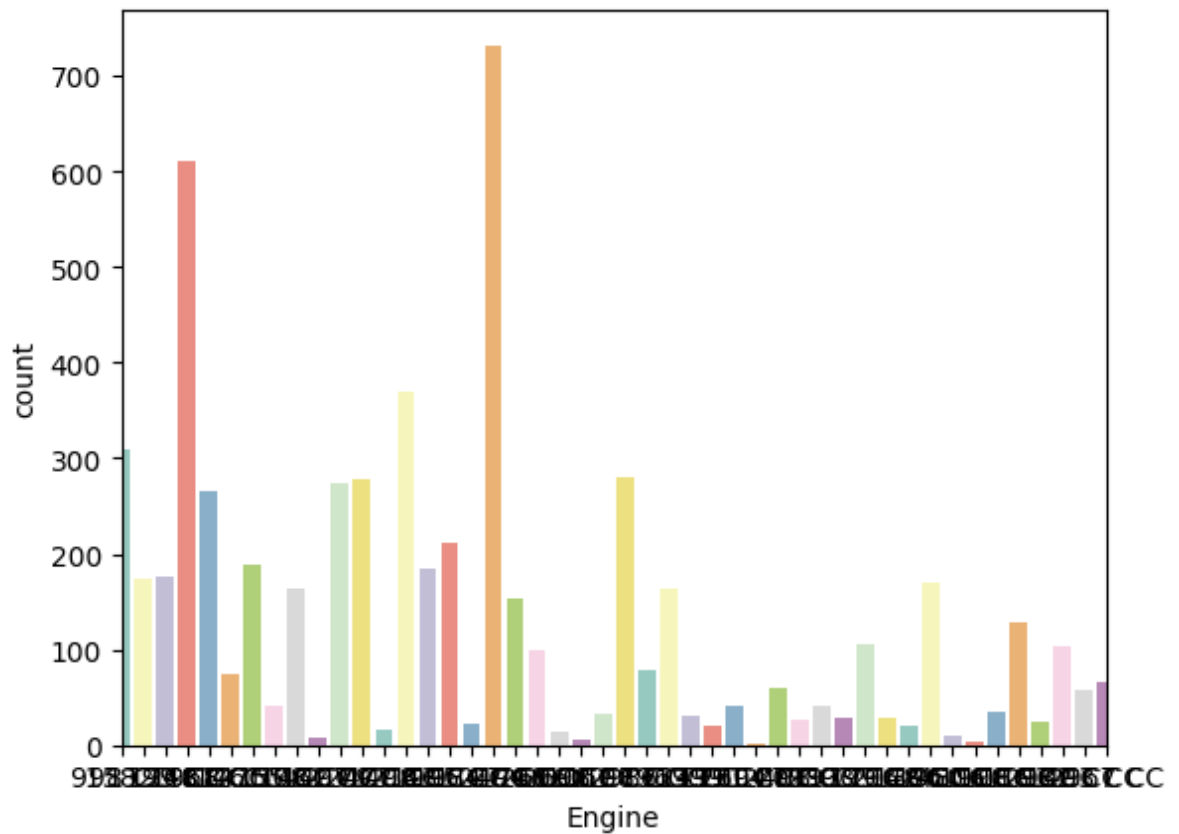
```
5.279722222222222
5.0
```

```
In [8]: print(df["New_Price"].isnull().sum()/df.shape[0])
print(df["Price"].isnull().sum()/df.shape[0])
print(df["Mileage"].isnull().sum()/df.shape[0])
print(df["Engine"].isnull().sum()/df.shape[0])
print(df["Power"].isnull().sum()/df.shape[0])
```

```
0.8612987729215497
0.1701364952433476
0.0002757479663587481
0.006342203226251206
0.006342203226251206
```

```
In [9]: print(df["Engine"].value_counts())
sns.countplot(x='Engine',data=df,palette='Set3')
plt.xlim(-0,45)
plt.show()
```

```
Engine
1197 CC    732
1248 CC    610
1498 CC    370
998 CC     309
1198 CC    281
...
1489 CC     1
1422 CC     1
2706 CC     1
1978 CC     1
1389 CC     1
Name: count, Length: 150, dtype: int64
```



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

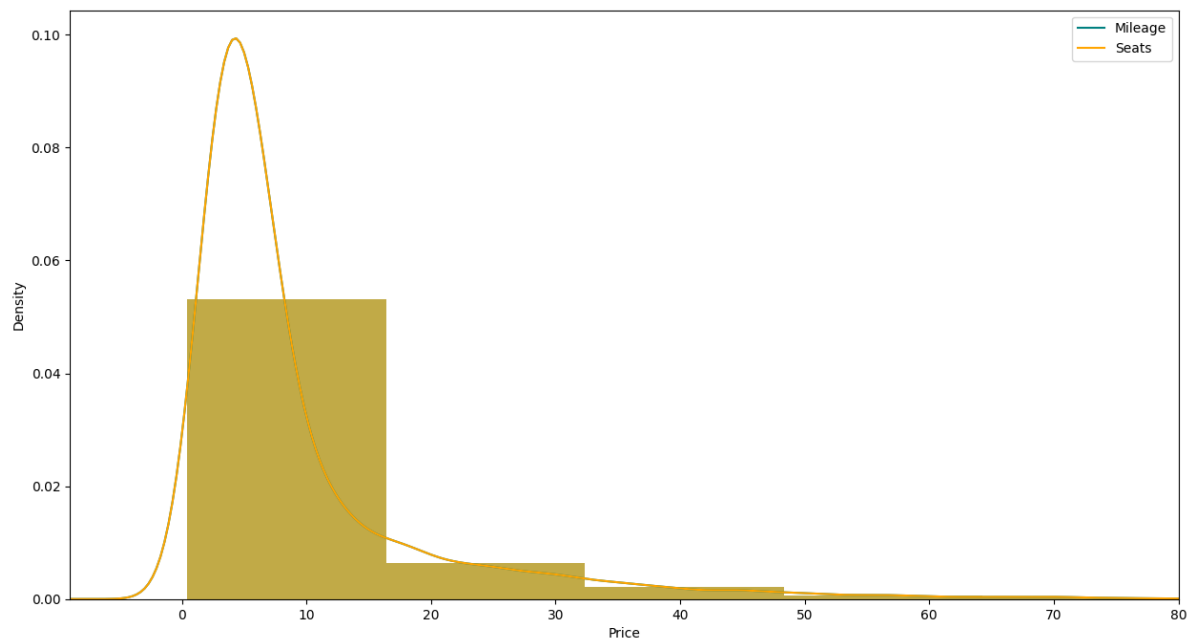
```
Out[11]: S.No.          0
         Name          0
         Location      0
         Year          0
         Kilometers_Driven  0
         Fuel_Type      0
         Transmission   0
         Owner_Type     0
         Mileage        2
         Engine        46
         Power         46
         Seats         53
         New_Price     6247
         Price        1234
         dtype: int64
```

In [13]: `df.head()`

```
Out[13]:
```

	S.No.	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type
0	0	Maruti Wagon R LXI CNG	Mumbai	2010	72000	CNG	Manual	First
1	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	First
2	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First
3	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	First
4	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Second

```
In [14]: plt.figure(figsize=(15,8))
ax=df["Price"].hist(bins=10,density=True,stacked=True,color='teal',alpha=0.6)
df["Price"].plot(kind='density',color='teal')
ax=df["Price"].hist(bins=10,density=True,stacked=True,color='orange',alpha=0.6)
df["Price"].plot(kind='density',color='orange')
ax.legend(['Mileage','Seats'])
ax.set(xlabel='Price')
plt.xlim(-9,80)
plt.show()
```

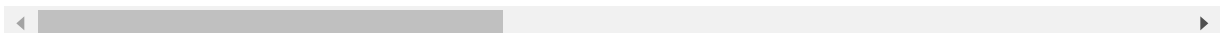


```
In [15]: training=pd.get_dummies(df,columns=["S.No."])
final_train=training
final_train.head()
```

Out[15]:

	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage
0	Maruti Wagon R LXI CNG	Mumbai	2010	72000	CNG	Manual	First	26. km/k
1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	First	19.6 km/k
2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First	18. km/k
3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	First	20.7 km/k
4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Second	15. km/k

5 rows × 7266 columns




```
In [19]: plt.figure(figsize=(15,8))
ax=sns.kdeplot(final_train["Price"][final_train.Year==1],color="darkturquoise")
sns.kdeplot(final_train["Kilometers_Driven"][final_train.Year==0],color="lightcoral")
plt.legend(['Cars', 'density'])
ax.set(xlabel='Year')
```

C:\Users\LENOVO\AppData\Local\Temp\ipykernel_14040\1420679919.py:2: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

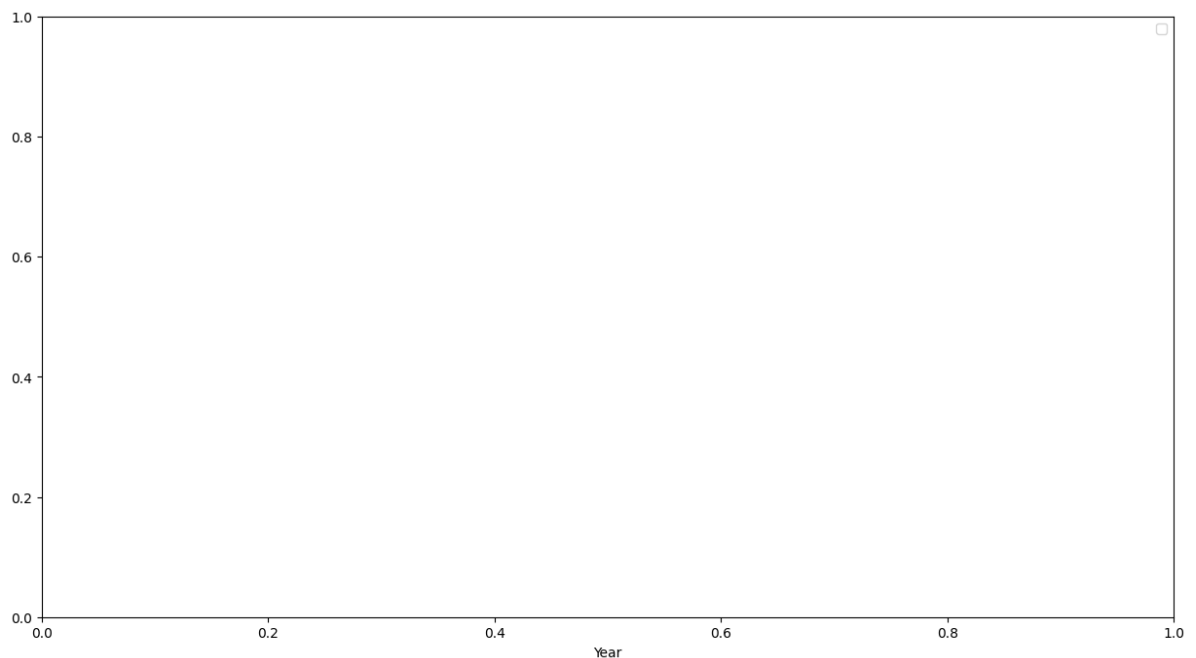
```
ax=sns.kdeplot(final_train["Price"][final_train.Year==1],color="darkturquoise",shade=True)
```

C:\Users\LENOVO\AppData\Local\Temp\ipykernel_14040\1420679919.py:3: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

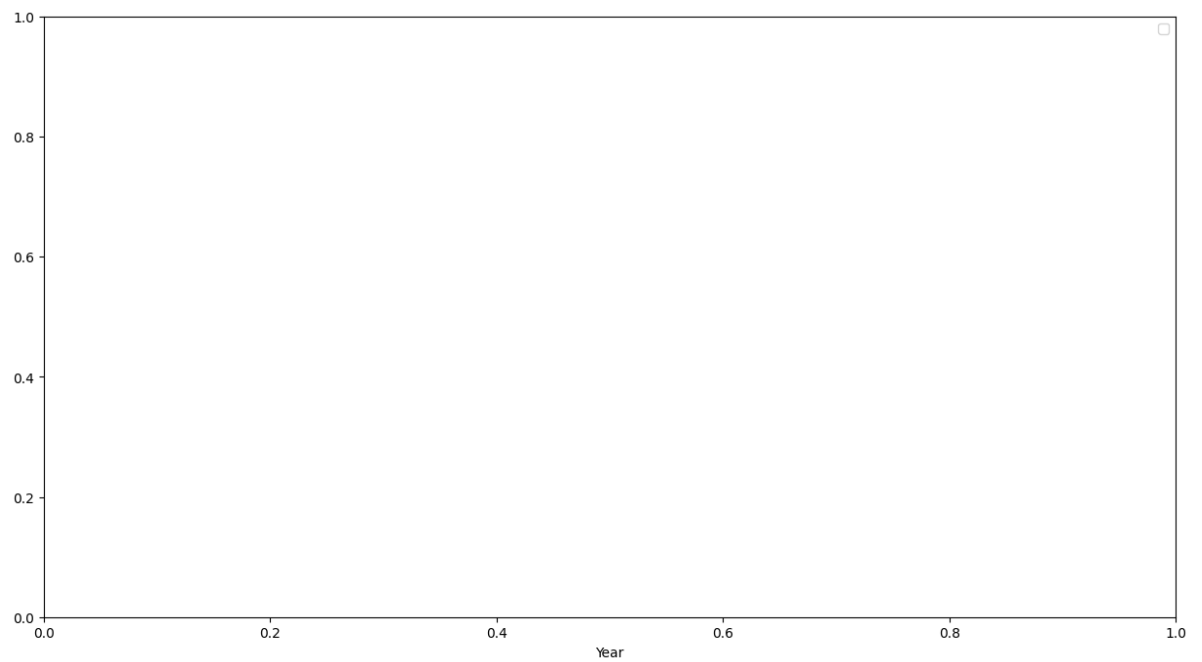
```
sns.kdeplot(final_train["Kilometers_Driven"][final_train.Year==0],color="lightcoral",shade=True)
```

Out[19]: [Text(0.5, 0, 'Year')]

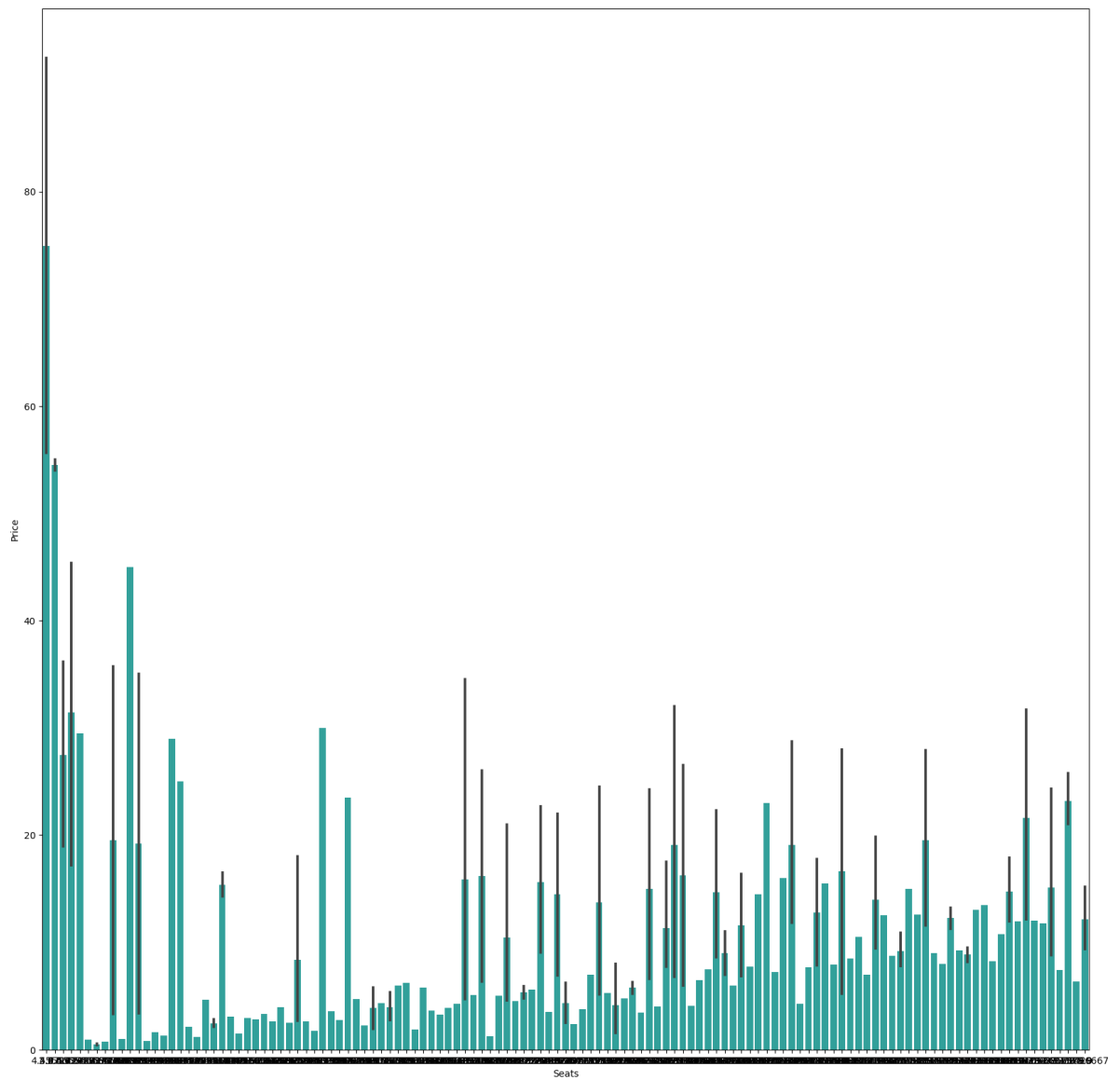


In [18]:

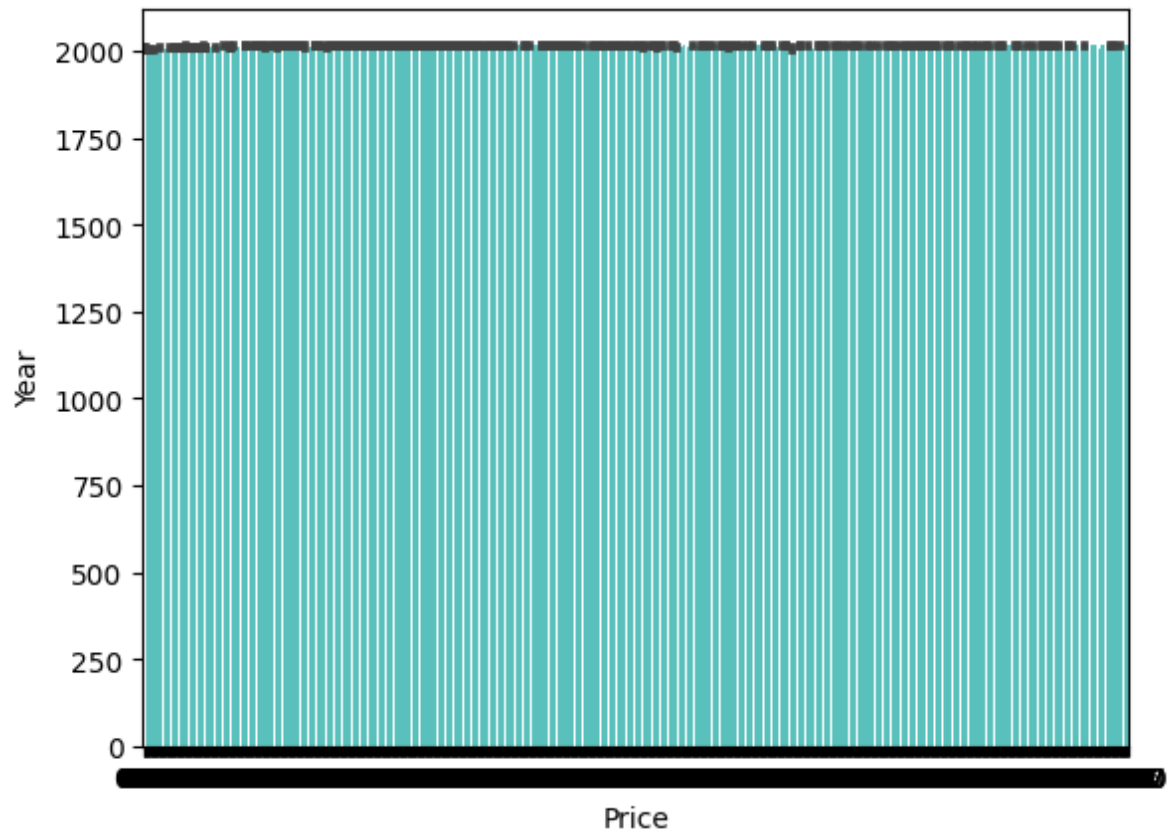
Out[18]: [Text(0.5, 0, 'Year')]



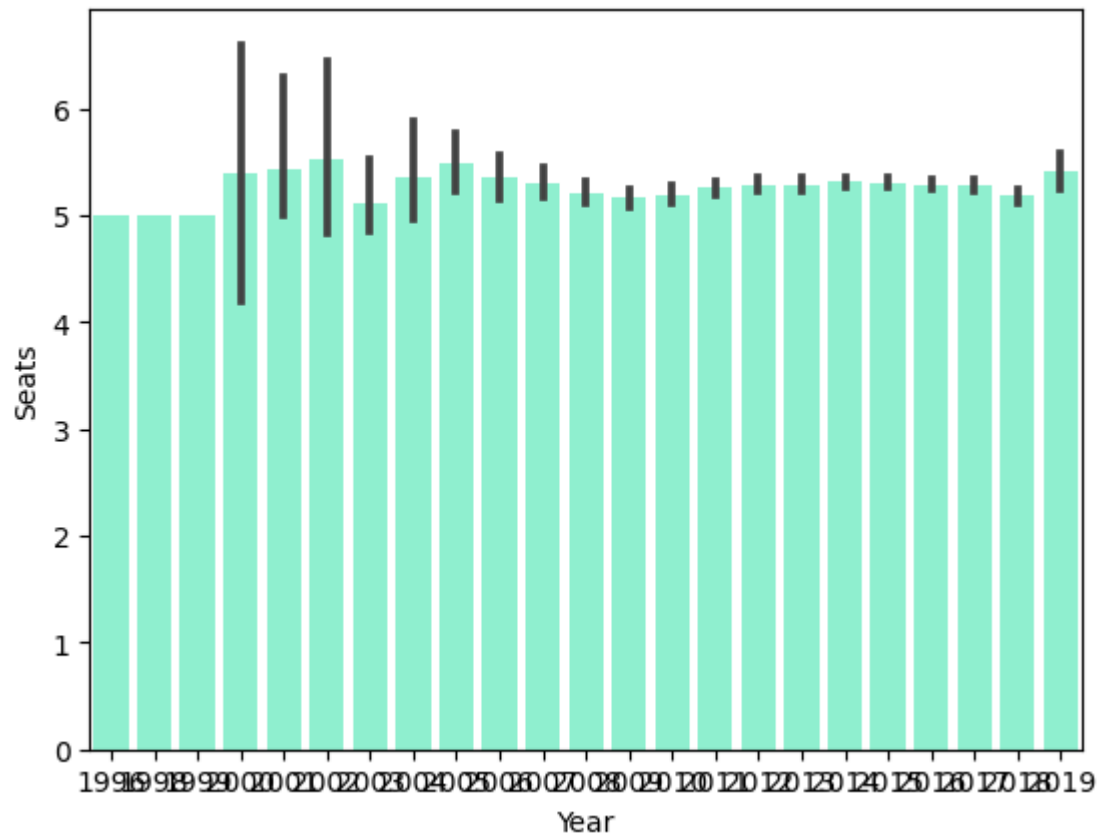
```
In [20]: plt.figure(figsize=(20,20))
avg_survival_byage=final_train[['Seats','Price']].groupby(['Price'],as_index=False)
sns.barplot(x='Seats',y='Price',data=avg_survival_byage,color="LightSeaGreen")
plt.show()
```



```
In [22]: sns.barplot(x='Price',y='Year',data=final_train,color="mediumturquoise")  
plt.show()
```



```
In [23]: import seaborn as sns
import matplotlib.pyplot as plt
sns.barplot(x='Year',y='Seats',data=df,color='aquamarine')
plt.show()
```



In []:

In []: