In [18]:
```
import pandas as pd
import numpy as np
from sklearn import preprocessing
import matplotlib.pyplot as plt
import seaborn as sns
sns.set(style="white")
sns.set(style="whitegrid",color_codes=True)
import warnings
warnings.simplefilter(action='ignore')
```

In [19]:
```
df=pd.read_csv(r"C:\Users\DELL\Downloads\used_cars_data.csv")
df
```

Out[19]:

	S.No.	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_T
0	0	Maruti Wagon R LXI CNG	Mumbai	2010	72000	CNG	Manual	F
1	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	F
2	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	F
3	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	F
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	F
7249	7249	Volkswagen Polo GT TSI	Mumbai	2015	59000	Petrol	Automatic	F
7250	7250	Nissan Micra Diesel XV	Kolkata	2012	28000	Diesel	Manual	F
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	F

7253 rows × 14 columns

In [20]:
```
df.head()
```

Out[20]:

	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 [21]:
```
df.shape
```

Out[21]:
```
(7253, 14)
```

In [23]:
```
df.describe()
```

Out[23]:

	S.No.	Year	Kilometers_Driven	Seats	Price
count	7253.000000	7253.000000	7.253000e+03	7200.000000	6019.000000
mean	3626.000000	2013.365366	5.869906e+04	5.279722	9.479468
std	2093.905084	3.254421	8.442772e+04	0.811660	11.187917
min	0.000000	1996.000000	1.710000e+02	0.000000	0.440000
25%	1813.000000	2011.000000	3.400000e+04	5.000000	3.500000
50%	3626.000000	2014.000000	5.341600e+04	5.000000	5.640000
75%	5439.000000	2016.000000	7.300000e+04	5.000000	9.950000
max	7252.000000	2019.000000	6.500000e+06	10.000000	160.000000

In [24]:
```
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 [25]:
```
df.isnull().sum()
```

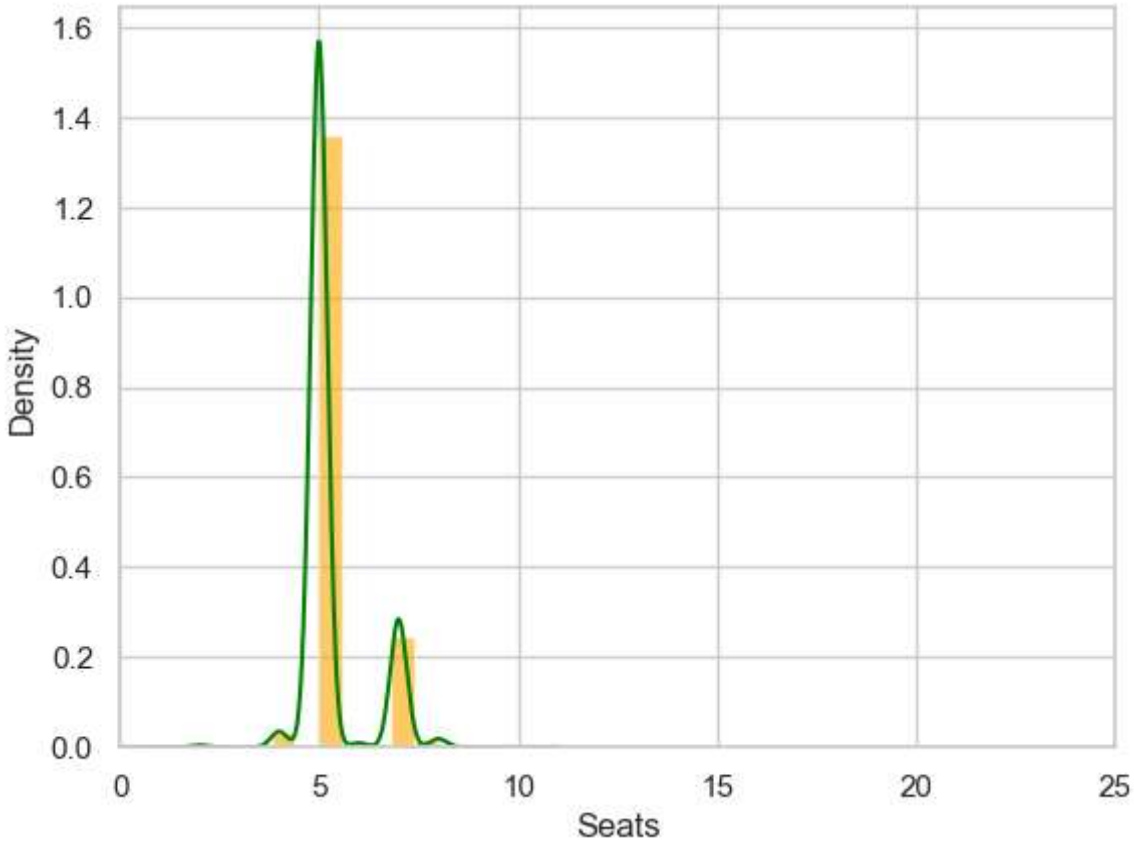
Out[25]:
```
S.No.                 0
Name                  0
Location              0
Year                  0
Kilometers_Driven     0
Fuel_Type             0
Transmission          0
Owner_Type            0
Mileage               2
Engine               46
Power                46
Seats                53
New_Price          6247
Price              1234
dtype: int64
```

In [26]:
```
df.dropna(inplace=True)
```

In [27]:
```
df.isnull().sum()
```

Out[27]:
```
S.No.                0
Name                 0
Location             0
Year                 0
Kilometers_Driven    0
Fuel_Type            0
Transmission         0
Owner_Type           0
Mileage              0
Engine               0
Power                0
Seats                0
New_Price            0
Price                0
dtype: int64
```

In [30]:
```
ax=df["Seats"].hist(bins=10,density=True,stacked=True,color='orange',alpha=0.6)
df["Seats"].plot(kind='density',color='green')
ax.set(xlabel='Seats')
plt.xlim(-0,25)
plt.show()
```



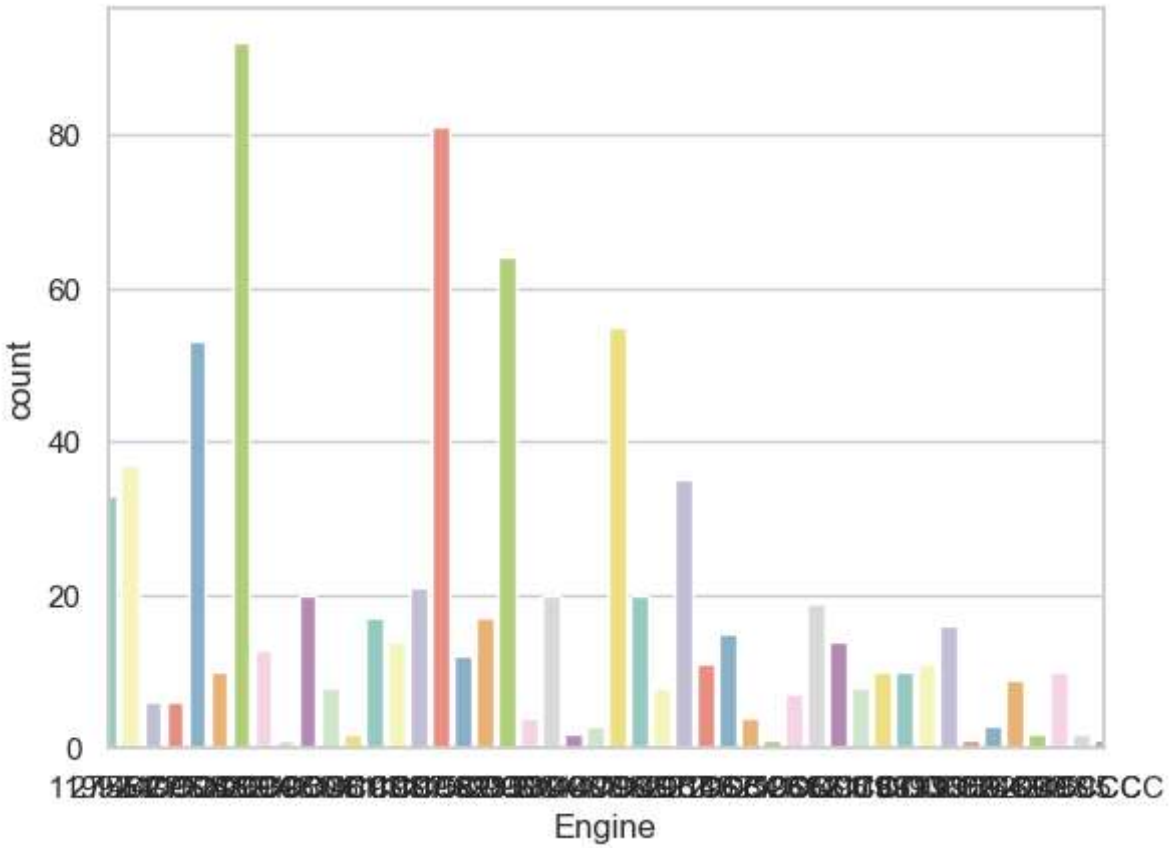
In [31]:
```
print(df["Seats"].mean(skipna=True))
print(df["Seats"].median(skipna=True))
print(df["New_Price"].isnull().sum()/df.shape[0])
print(df["Price"].isnull().sum()/df.shape[0])
print(df["Mileage"].isnull().sum()/df.shape[0])
print(df["Engine"].isnull().sum()/df.shape[0])
print(df["Power"].isnull().sum()/df.shape[0])
```

```
5.30498177399757
5.0
0.0
0.0
0.0
0.0
0.0
```

In [32]:
```
print(df['Engine'].value_counts())
sns.countplot(x='Engine',data=df,palette='Set3')
plt.xlim(-0,45)
plt.show()
```

```
Engine
1248 CC    92
1197 CC    81
998 CC     64
1498 CC    55
1995 CC    53
2755 CC    37
1497 CC    35
1199 CC    33
1198 CC    21
1798 CC    20
2393 CC    20
1591 CC    20
1968 CC    19
1582 CC    17
1461 CC    17
799 CC     16
2143 CC    15
1193 CC    14
1956 CC    14
1999 CC    13
1196 CC    12
999 CC     11
2987 CC    11
1186 CC    10
2993 CC    10
1598 CC    10
2179 CC    10
1364 CC     9
1950 CC     8
1496 CC     8
1998 CC     8
2523 CC     7
2477 CC     6
1462 CC     6
2967 CC     4
2996 CC     4
1194 CC     3
1969 CC     3
1493 CC     3
2894 CC     2
1396 CC     2
2489 CC     2
2498 CC     2
1991 CC     2
2198 CC     1
1997 CC     1
1086 CC     1
2925 CC     1
4951 CC     1
2487 CC     1
1984 CC     1
1395 CC     1
2995 CC     1
2999 CC     1
1595 CC     1
2694 CC     1
```

```
1368 CC     1
1047 CC     1
Name: count, dtype: int64
```



In [33]:
```
data=df.copy()
data['Seats'].fillna(df['Seats'].median(skipna=True),inplace=True)
data.drop('New_Price',axis=1,inplace=True)
data['Price'].fillna(df['Price'].median(skipna=True),inplace=True)
data['Mileage'].fillna(df['Mileage'].value_counts().idxmax(),inplace=True)
data.drop('Engine',axis=1,inplace=True)
data.drop('Power',axis=1,inplace=True)
```

In [34]:
```
data.isnull().sum()
```

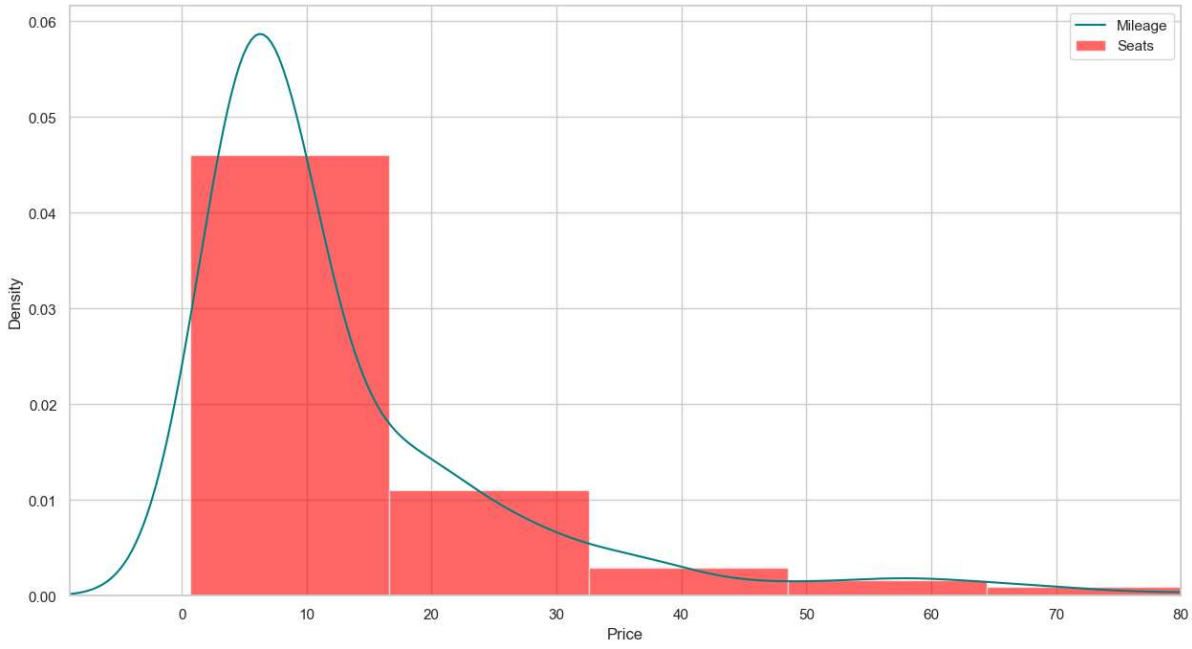
Out[34]:
```
S.No.                0
Name                 0
Location             0
Year                 0
Kilometers_Driven    0
Fuel_Type            0
Transmission         0
Owner_Type           0
Mileage              0
Seats                0
Price                0
dtype: int64
```

In [36]:
```
df.head()
```

Out[36]:

	S.No.	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	M
2	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First	
7	7	Toyota Innova Crysta 2.8 GX AT 8S	Mumbai	2016	36000	Diesel	Automatic	First	
10	10	Maruti Ciaz Zeta	Kochi	2018	25692	Petrol	Manual	First	
15	15	Mitsubishi Pajero Sport 4X4	Delhi	2014	110000	Diesel	Manual	First	
20	20	BMW 3 Series 320d	Kochi	2014	32982	Diesel	Automatic	First	

In [39]:
```
plt.figure(figsize=(15,8))
ax=df["Price"].hist(bins=10,density=True,stacked=True,color='red',alpha=0.6)
df["Price"].plot(kind='density',color='teal')
ax.legend(['Mileage','Seats'])
ax.set(xlabel='Price')
plt.xlim(-9,80)
plt.show()
```



In [40]:
```
training=pd.get_dummies(data,columns=["S.No."])
final_train=training
final_train.head()
```

Out[40]:

	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage
2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First	18.2 kmpl
7	Toyota Innova Crysta 2.8 GX AT 8S	Mumbai	2016	36000	Diesel	Automatic	First	11.36 kmpl
10	Maruti Ciaz Zeta	Kochi	2018	25692	Petrol	Manual	First	21.56 kmpl
15	Mitsubishi Pajero Sport 4X4	Delhi	2014	110000	Diesel	Manual	First	13.5 kmpl
20	BMW 3 Series 320d	Kochi	2014	32982	Diesel	Automatic	First	22.69 kmpl

5 rows × 833 columns

In [41]:
```
sns.barplot(x='Price',y='Year',data=final_train,color='g')
plt.show()
```

In [44]:
```
sns.barplot(x='Year',y='Seats',data=df,color='m')
plt.show()
```