In [28]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
```

In [29]:
```
vehicle_data=pd.read_csv("Cars_Dataset.csv")
```

In [30]:
```
vehicle_data.head()
```

Out[30]:

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner
0	ritz	2014	3.35	5.59	27000	Petrol	Dealer	Manual	0
1	sx4	2013	4.75	9.54	43000	Diesel	Dealer	Manual	0
2	ciaz	2017	7.25	9.85	6900	Petrol	Dealer	Manual	0
3	wagon r	2011	2.85	4.15	5200	Petrol	Dealer	Manual	0
4	swift	2014	4.60	6.87	42450	Diesel	Dealer	Manual	0

In [31]:
```
vehicle_data.shape
```

Out[31]:
```
(602, 9)
```

In [32]:
```
vehicle_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 602 entries, 0 to 601
Data columns (total 9 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   Car_Name       602 non-null    object 
 1   Year           602 non-null    int64  
 2   Selling_Price  602 non-null    float64
 3   Present_Price  602 non-null    float64
 4   Kms_Driven     602 non-null    int64  
 5   Fuel_Type      602 non-null    object 
 6   Seller_Type    602 non-null    object 
 7   Transmission   602 non-null    object 
 8   Owner          602 non-null    int64  
dtypes: float64(2), int64(3), object(4)
memory usage: 42.5+ KB
```

In [33]:
```
vehicle_data.describe()
```

Out[33]:

	Year	Selling_Price	Present_Price	Kms_Driven	Owner
count	602.000000	602.000000	602.000000	602.000000	602.000000
mean	2014.127907	5.286296	8.253472	36947.205980	0.107973
std	2.932165	5.116958	8.659543	38854.518596	0.341236
min	2003.000000	0.100000	0.320000	500.000000	0.000000
25%	2013.000000	1.735000	2.090000	15000.000000	0.000000
50%	2015.000000	4.250000	6.950000	32000.000000	0.000000
75%	2016.000000	6.750000	10.650000	48767.000000	0.000000
max	2019.000000	36.250000	93.850000	500000.000000	3.000000

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

Out[34]:
```
Car_Name         0
Year             0
Selling_Price    0
Present_Price    0
Kms_Driven       0
Fuel_Type        0
Seller_Type      0
Transmission     0
Owner            0
dtype: int64
```

In [35]:
```
vehicle_data.duplicated().sum()
```

Out[35]:
```
2
```

In [36]:
```
vehicle_data[vehicle_data.duplicated()]
```

Out[36]:

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner
17	ertiga	2016	7.75	10.79	43000	Diesel	Dealer	Manual	0
93	fortuner	2015	23.00	30.61	40000	Diesel	Dealer	Automatic	0

In [37]:
```
vehicle_data.drop_duplicates(inplace=True)
```

In [38]:
```
vehicle_data.duplicated().sum()
```

Out[38]:
```
0
```

In [39]:
```
vehicle_data.reset_index(drop=True)
```

Out[39]:

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner
0	ritz	2014	3.35	5.59	27000	Petrol	Dealer	Manual	0
1	sx4	2013	4.75	9.54	43000	Diesel	Dealer	Manual	0
2	ciaz	2017	7.25	9.85	6900	Petrol	Dealer	Manual	0
3	wagon r	2011	2.85	4.15	5200	Petrol	Dealer	Manual	0
4	swift	2014	4.60	6.87	42450	Diesel	Dealer	Manual	0
...	...	...	...	...	...	...	...	...	...
595	Fiat Grande Punto 1.3 Emotion Pack 90HP (Diesel)	2018	2.70	2.85	1200	Petrol	Individual	Manual	0
596	Hyundai Santa Fe 2WD AT	2018	2.60	2.72	4100	Petrol	Individual	Manual	0
597	Chevrolet Optra Magnum 1.6 LT Petrol	2016	2.60	3.62	21700	Petrol	Individual	Manual	0
598	Tata New Safari DICOR 2.2 EX 4x2	2015	2.60	4.70	16500	Petrol	Individual	Manual	1
599	Honda Accord 2.4 AT	2014	2.50	2.75	15000	Petrol	Individual	Manual	0

600 rows × 9 columns

In [40]:
```
vehicle_data['Current_Year']=2022
```

In [41]:
```
vehicle_data.head()
```

Out[41]:

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner	Current_Year
0	ritz	2014	3.35	5.59	27000	Petrol	Dealer	Manual	0	2022
1	sx4	2013	4.75	9.54	43000	Diesel	Dealer	Manual	0	2022
2	ciaz	2017	7.25	9.85	6900	Petrol	Dealer	Manual	0	2022
3	wagon r	2011	2.85	4.15	5200	Petrol	Dealer	Manual	0	2022
4	swift	2014	4.60	6.87	42450	Diesel	Dealer	Manual	0	2022

In [42]:
```
vehicle_data['Car_Age']=vehicle_data['Current_Year'] - vehicle_data['Year']
```

In [43]:
```
vehicle_data.head()
```

Out[43]:

	Car_Name	Year	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner	Current_Year	Car_Age
0	ritz	2014	3.35	5.59	27000	Petrol	Dealer	Manual	0	2022	8
1	sx4	2013	4.75	9.54	43000	Diesel	Dealer	Manual	0	2022	9
2	ciaz	2017	7.25	9.85	6900	Petrol	Dealer	Manual	0	2022	5
3	wagon r	2011	2.85	4.15	5200	Petrol	Dealer	Manual	0	2022	11
4	swift	2014	4.60	6.87	42450	Diesel	Dealer	Manual	0	2022	8

In [44]:
```
vehicle_data.drop(columns=['Car_Name','Year','Current_Year'],inplace=True)
```

In [45]:
```
vehicle_data.head()
```

Out[45]:

	Selling_Price	Present_Price	Kms_Driven	Fuel_Type	Seller_Type	Transmission	Owner	Car_Age
0	3.35	5.59	27000	Petrol	Dealer	Manual	0	8
1	4.75	9.54	43000	Diesel	Dealer	Manual	0	9
2	7.25	9.85	6900	Petrol	Dealer	Manual	0	5
3	2.85	4.15	5200	Petrol	Dealer	Manual	0	11
4	4.60	6.87	42450	Diesel	Dealer	Manual	0	8

In [46]:
```
print(vehicle_data.Owner.value_counts())
print(vehicle_data.Fuel_Type.value_counts())
print(vehicle_data.Transmission.value_counts())
print(vehicle_data.Seller_Type.value_counts())
```

```
0    539
1     59
3      2
Name: Owner, dtype: int64
Petrol    478
Diesel    118
CNG         4
Name: Fuel_Type, dtype: int64
Manual       501
Automatic     99
Name: Transmission, dtype: int64
Dealer        388
Individual    212
Name: Seller_Type, dtype: int64
```

In [47]:
```
vehicle_data=pd.get_dummies(vehicle_data,columns=['Fuel_Type'],drop_first=True)
```

In [48]:
```
vehicle_data=pd.get_dummies(vehicle_data,columns=['Transmission','Seller_Type'])
```

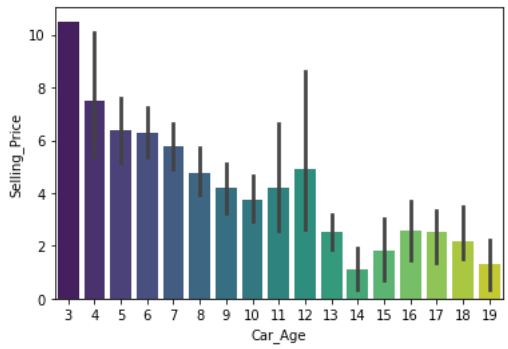
In [49]:
```
vehicle_data.head()
```

Out[49]:

	Selling_Price	Present_Price	Kms_Driven	Owner	Car_Age	Fuel_Type_Diesel	Fuel_Type_Petrol	Transmission_Automatic	Transmission_Manual	Seller_Type_I
0	3.35	5.59	27000	0	8	0	1	0	1	
1	4.75	9.54	43000	0	9	1	0	0	1	
2	7.25	9.85	6900	0	5	0	1	0	1	
3	2.85	4.15	5200	0	11	0	1	0	1	
4	4.60	6.87	42450	0	8	1	0	0	1	

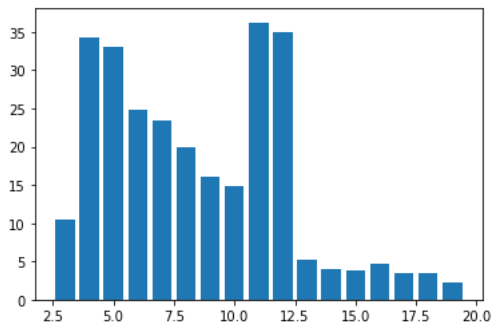
In [50]:
```
sns.barplot(x='Car_Age',y='Selling_Price',data=vehicle_data,palette='viridis')
```

Out[50]:
```
<AxesSubplot:xlabel='Car_Age', ylabel='Selling_Price'>
```



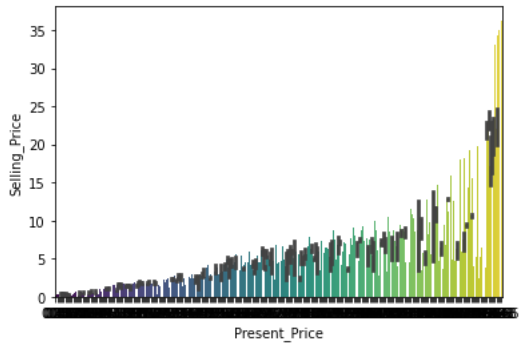
In [51]:
```
x=vehicle_data['Car_Age']
y=vehicle_data['Selling_Price']
plt.bar(x,y)
```

Out[51]:
```
<BarContainer object of 600 artists>
```



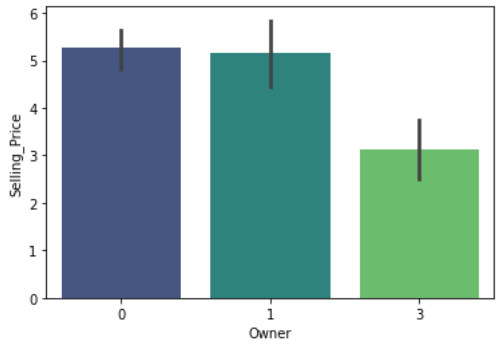
In [52]:
```
sns.barplot(x='Present_Price',y='Selling_Price',data=vehicle_data,palette='viridis')
```

Out[52]: <AxesSubplot:xlabel='Present_Price', ylabel='Selling_Price'>



In [53]:
```
sns.barplot(x='Owner',y='Selling_Price',data=vehicle_data,palette='viridis')
```

Out[53]: <AxesSubplot:xlabel='Owner', ylabel='Selling_Price'>



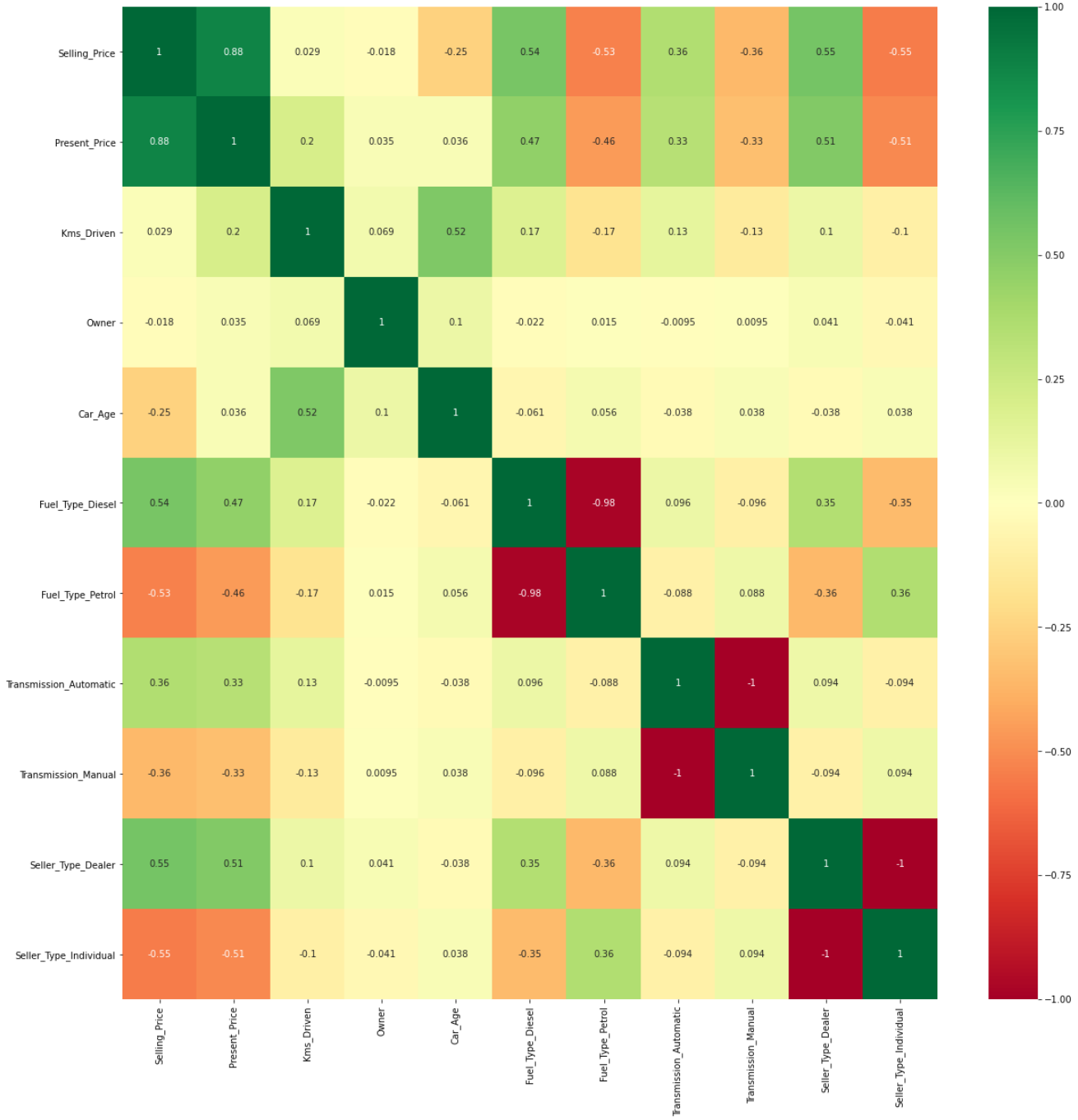
In [54]:
```
vehicle_data.corr()
```

Out[54]:

	Selling_Price	Present_Price	Kms_Driven	Owner	Car_Age	Fuel_Type_Diesel	Fuel_Type_Petrol	Transmission_Automatic	Transmiss
Selling_Price	1.000000	0.877692	0.028683	-0.017895	-0.252245	0.544203	-0.532450	0.357515	
Present_Price	0.877692	1.000000	0.203855	0.035320	0.036252	0.468230	-0.460160	0.333546	
Kms_Driven	0.028683	0.203855	1.000000	0.069445	0.517082	0.172897	-0.173227	0.128623	
Owner	-0.017895	0.035320	0.069445	1.000000	0.101527	-0.021899	0.014755	-0.009534	
Car_Age	-0.252245	0.036252	0.517082	0.101527	1.000000	-0.060675	0.056401	-0.037946	
Fuel_Type_Diesel	0.544203	0.468230	0.172897	-0.021899	-0.060675	1.000000	-0.979381	0.096361	
Fuel_Type_Petrol	-0.532450	-0.460160	-0.173227	0.014755	0.056401	-0.979381	1.000000	-0.087800	
Transmission_Automatic	0.357515	0.333546	0.128623	-0.009534	-0.037946	0.096361	-0.087800	1.000000	
Transmission_Manual	-0.357515	-0.333546	-0.128623	0.009534	0.037946	-0.096361	0.087800	-1.000000	
Seller_Type_Dealer	0.547843	0.510499	0.101225	0.040504	-0.038210	0.348193	-0.356111	0.093748	
Seller_Type_Individual	-0.547843	-0.510499	-0.101225	-0.040504	0.038210	-0.348193	0.356111	-0.093748	

In [55]:
```
plt.figure(figsize=(20,20))
sns.heatmap(vehicle_data.corr(),annot=True,cmap="RdYlGn")
```

Out[55]: <AxesSubplot:>



In [56]:
```
vehicle_data.corr()['Selling_Price']
```

Out[56]:
```
Selling_Price            1.000000
Present_Price            0.877692
Kms_Driven               0.028683
Owner                   -0.017895
Car_Age                 -0.252245
Fuel_Type_Diesel         0.544203
Fuel_Type_Petrol        -0.532450
Transmission_Automatic   0.357515
Transmission_Manual     -0.357515
Seller_Type_Dealer       0.547843
Seller_Type_Individual  -0.547843
Name: Selling_Price, dtype: float64
```

In [57]:
```
y = vehicle_data['Selling_Price']
x = vehicle_data.drop('Selling_Price',axis=1)
```

In [58]:
```
x.shape
```

Out[58]: (600, 10)

In [59]:
```
y.shape
```

Out[59]: (600,)

In [60]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=1)
```

In [61]:
```
x_train.shape
```

Out[61]: (480, 10)

In [62]:
```
x_test.shape
```

Out[62]: (120, 10)

In [63]:
```
y_train.shape
```

Out[63]: (480,)

In [64]:
```
y_test.shape
```

Out[64]: (120,)

In [65]:
```
from sklearn.linear_model import LinearRegression
lm = LinearRegression()
```

In [66]:
```
lm.fit(x_train,y_train)
```

Out[66]: LinearRegression()

In [67]:
```
lm.score(x_test,y_test)
```

Out[67]: 0.9014483233111038

In [68]:
```
from sklearn.linear_model import Lasso
lo = Lasso()
```

In [69]:
```
lo.fit(x_train,y_train)
```

Out[69]: Lasso()

In [70]:
```
lo.score(x_test,y_test)
```

Out[70]: 0.861215971006845

In [71]:
```
from sklearn.linear_model import Ridge
re=Ridge()
```

In [72]:
```
re.fit(x_train,y_train)
```

Out[72]: Ridge()

In [73]:
```
re.score(x_test,y_test)
```

Out[73]: 0.9010592411652187

In [ ]:
```

```