

```
In [2]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sn
from sklearn import linear_model

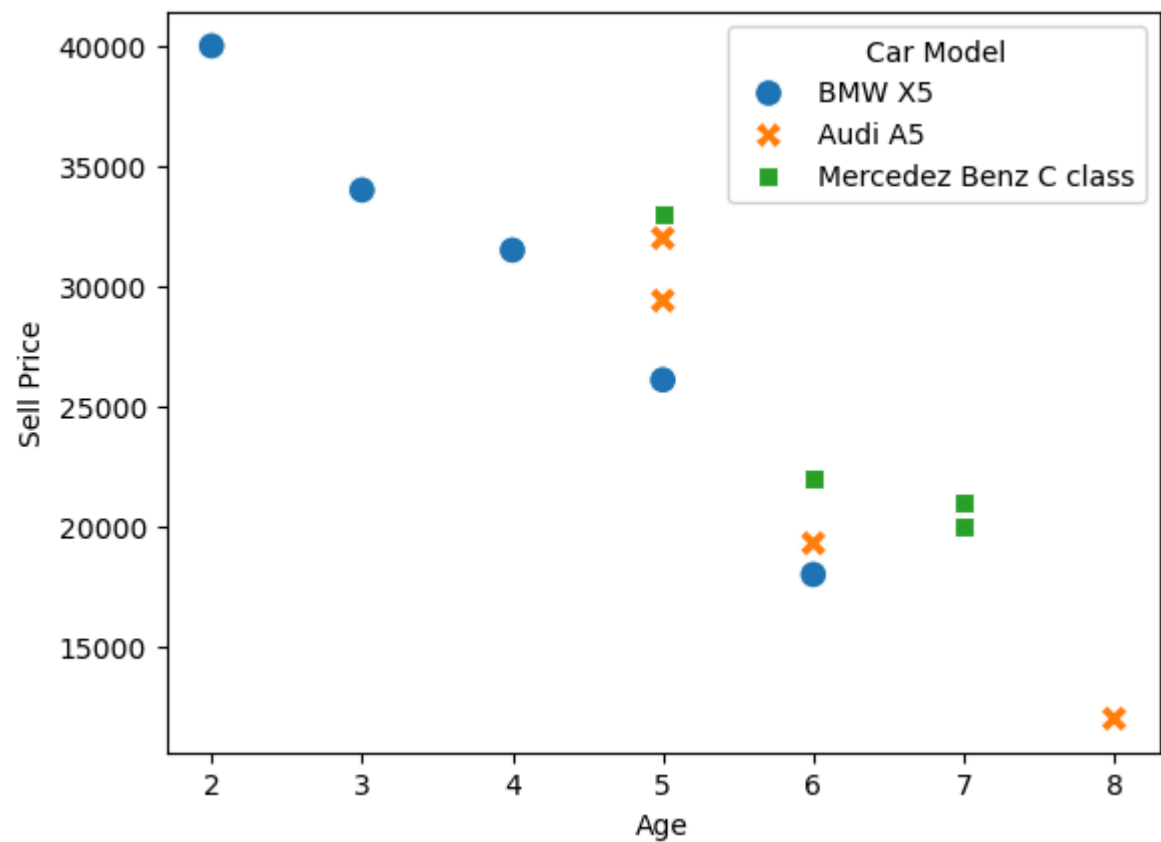
In [4]: df = pd.read_csv("Dataset_6.csv")
df
```

Out[4]:

	Car Model	Mileage	Sell Price(\$)	Age(yrs)
0	BMW X5	69000	18000	6
1	BMW X5	35000	34000	3
2	BMW X5	57000	26100	5
3	BMW X5	22500	40000	2
4	BMW X5	46000	31500	4
5	Audi A5	59000	29400	5
6	Audi A5	52000	32000	5
7	Audi A5	72000	19300	6
8	Audi A5	91000	12000	8
9	Mercedes Benz C class	67000	22000	6
10	Mercedes Benz C class	83000	20000	7
11	Mercedes Benz C class	79000	21000	7
12	Mercedes Benz C class	59000	33000	5

```
In [8]: plt.xlabel("Age")
plt.ylabel("Sell Price")
sns.scatterplot(data=df, x='Age(yrs)', y='Sell Price($)', hue='Car Model', style='Car Model', s=100)

Out[8]: <Axes: xlabel='Age', ylabel='Sell Price'>
```



```
In [12]: dummies = pd.get_dummies(df['Car Model']).astype(int)
dummies
```

Out[12]:

	Audi A5	BMW X5	Mercedes Benz C class
0	0	1	0
1	0	1	0
2	0	1	0
3	0	1	0
4	0	1	0
5	1	0	0
6	1	0	0
7	1	0	0
8	1	0	0
9	0	0	1
10	0	0	1
11	0	0	1
12	0	0	1

```
In [14]: merged = pd.concat([df, dummies], axis=1)
merged
```

Out[14]:

	Car Model	Mileage	Sell Price(\$)	Age(yrs)	Audi A5	BMW X5	Mercedes Benz C class
0	BMW X5	69000	18000	6	0	1	0
1	BMW X5	35000	34000	3	0	1	0
2	BMW X5	57000	26100	5	0	1	0
3	BMW X5	22500	40000	2	0	1	0
4	BMW X5	46000	31500	4	0	1	0
5	Audi A5	59000	29400	5	1	0	0
6	Audi A5	52000	32000	5	1	0	0
7	Audi A5	72000	19300	6	1	0	0
8	Audi A5	91000	12000	8	1	0	0
9	Mercedes Benz C class	67000	22000	6	0	0	1
10	Mercedes Benz C class	83000	20000	7	0	0	1
11	Mercedes Benz C class	79000	21000	7	0	0	1
12	Mercedes Benz C class	59000	33000	5	0	0	1

```
In [46]: final = merged.drop(['Car Model','Mercedes Benz C class','Sell Price($)', axis='columns')
final # mileage and age are the independent variables in the data and price dependent variable.
```

Out[46]:

	Mileage	Age(yrs)	Audi A5	BMW X5
0	69000	6	0	1
1	35000	3	0	1
2	57000	5	0	1
3	22500	2	0	1
4	46000	4	0	1
5	59000	5	1	0
6	52000	5	1	0
7	72000	6	1	0
8	91000	8	1	0
9	67000	6	0	0
10	83000	7	0	0
11	79000	7	0	0
12	59000	5	0	0

```
In [48]: y = df['Sell Price($)']
y
```

Out[48]:

0	18000
1	34000
2	26100
3	40000
4	31500
5	29400
6	32000
7	19300
8	12000
9	22000
10	20000
11	21000
12	33000

Name: Sell Price(\$), dtype: int64

```
In [50]: model = linear_model.LinearRegression()
model.fit(final,y)
```

Out[50]:

LinearRegression

LinearRegression()

```
In [52]: model.score(final,y)
```

Out[52]: 0.9417050937281082

```
In [42]: # 0,1-- BMW X5
#1,0-- Audi A5. and 6 represents age of the vechicle
#0,0 -- Mercedes Benz C class. here 34000 represents mileage
model.predict([[34000,6,1,0]])
```

C:\Users\MyPc\anaconda3\Lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
warnings.warn(

Out[42]: array([35944.2122564])

```
In [54]: # 0,1-- BMW X5
#1,0-- Audi A5. and 6 represents age of the vechicle
#0,0 -- Mercedes Benz C class. here 69000 represents mileage
model.predict([[69000,6,0,1]])
```

C:\Users\MyPc\anaconda3\Lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
warnings.warn(

Out[54]: array([18705.2723644])

```
In [56]: # 0,1-- BMW X5
#1,0-- Audi A5. and 4 represents age of the vechicle
#0,0 -- Mercedes Benz C class. here 45000 represents mileage
model.predict([[45000,4,0,0]])
```

C:\Users\MyPc\anaconda3\Lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
warnings.warn(

Out[56]: array([36991.31721061])

```
In [58]: # 0,1-- BMW X5
#1,0-- Audi A5. and 4 represents age of the vechicle
#0,0 -- Mercedes Benz C class. here 45000 represents mileage
model.predict([[86000,7,0,1]])
```

C:\Users\MyPc\anaconda3\Lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
warnings.warn(

Out[58]: array([11080.74313219])

```
In [60]: model.score(final,y)
```

