```
import pandas as pd
```

```
from google.colab import files
uploaded = files.upload()
```

```
Choose Files randomeforestdataset.csv
• randomeforestdataset.csv(text/csv) - 26717 bytes, last modified: 4/16/2024 - 100% done
Saving randomeforestdataset.csv to randomeforestdataset.csv
```

```
dataset = pd.read_csv('randomeforestdataset.csv')
dataset = dataset.drop(['car_ID'],axis=1)
```

```
print(dataset.shape)
print(dataset.head(5))
```

```
(205, 25)
   symboling                   CarName fueltype aspiration doornumber  \
0          3        alfa-romero giulia      gas        std        two
1          3       alfa-romero stelvio      gas        std        two
2          1  alfa-romero Quadrifoglio      gas        std        two
3          2               audi 100 ls      gas        std       four
4          2                audi 100ls      gas        std       four

       carbody drivewheel enginelocation  wheelbase  carlength  ...  \
0  convertible        rwd          front       88.6      168.8  ...
1  convertible        rwd          front       88.6      168.8  ...
2    hatchback        rwd          front       94.5      171.2  ...
3        sedan        fwd          front       99.8      176.6  ...
4        sedan        4wd          front       99.4      176.6  ...

   enginesize  fuelsystem  boreratio stroke compressionratio  horsepower  \
0         130        mpfi       3.47   2.68              9.0         111
1         130        mpfi       3.47   2.68              9.0         111
2         152        mpfi       2.68   3.47              9.0         154
3         109        mpfi       3.19   3.40             10.0         102
4         136        mpfi       3.19   3.40              8.0         115

  peakrpm  citympg  highwaympg    price
0    5000       21          27  13495.0
1    5000       21          27  16500.0
2    5000       19          26  16500.0
3    5500       24          30  13950.0
4    5500       18          22  17450.0

[5 rows x 25 columns]
```

```
Xdata = dataset.drop('price',axis='columns')
numericalCols=Xdata.select_dtypes(exclude=['object']).columns
X=Xdata[numericalCols]
X
```

	symboling	wheelbase	carlength	carwidth	carheight	curbweight	enginesize	bo
0	3	88.6	168.8	64.1	48.8	2548	130	
1	3	88.6	168.8	64.1	48.8	2548	130	
2	1	94.5	171.2	65.5	52.4	2823	152	
3	2	99.8	176.6	66.2	54.3	2337	109	
4	2	99.4	176.6	66.4	54.3	2824	136	
...	...	...	...	...	...	...	...	
200	-1	109.1	188.8	68.9	55.5	2952	141	
201	-1	109.1	188.8	68.8	55.5	3049	141	
202	-1	109.1	188.8	68.9	55.5	3012	173	
203	-1	109.1	188.8	68.9	55.5	3217	145	
204	-1	109.1	188.8	68.9	55.5	3062	141	

205 rows × 14 columns

Next steps: Generate code with X View recommended plots

```
Y = dataset['price']
Y
```

```
0      13495.0
1      16500.0
```

```
2      16500.0
3      13950.0
4      17450.0
        ...
200    16845.0
201    19045.0
202    21485.0
203    22470.0
204    22625.0
Name: price, Length: 205, dtype: float64
```

```
from sklearn.preprocessing import scale
cols = X.columns
X = pd.DataFrame(scale(X))
X.columns = cols
X
```

	symboling	wheelbase	carlength	carwidth	carheight	curbweight	enginesize	bo
0	1.743470	-1.690772	-0.426521	-0.844782	-2.020417	-0.014566	0.074449	0
1	1.743470	-1.690772	-0.426521	-0.844782	-2.020417	-0.014566	0.074449	0
2	0.133509	-0.708596	-0.231513	-0.190566	-0.543527	0.514882	0.604046	-2
3	0.938490	0.173698	0.207256	0.136542	0.235942	-0.420797	-0.431076	-0
4	0.938490	0.107110	0.207256	0.230001	0.235942	0.516807	0.218885	-0
...	...	...	...	...	...	...	...	
200	-1.476452	1.721873	1.198549	1.398245	0.728239	0.763241	0.339248	1
201	-1.476452	1.721873	1.198549	1.351515	0.728239	0.949992	0.339248	1
202	-1.476452	1.721873	1.198549	1.398245	0.728239	0.878757	1.109571	0
203	-1.476452	1.721873	1.198549	1.398245	0.728239	1.273437	0.435538	-1
204	-1.476452	1.721873	1.198549	1.398245	0.728239	0.975021	0.339248	1

205 rows × 14 columns

Next steps: Generate code with X View recommended plots

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(X,Y,test_size=0.20,random_state=0)
```

```
from sklearn.ensemble import RandomForestRegressor
model = RandomForestRegressor()
model.fit(x_train, y_train)
```

```
RandomForestRegressor
RandomForestRegressor()
```

```
ypred = model.predict(x_test)

from sklearn.metrics import r2_score
r2score = r2_score(y_test,ypred)
print("R2Score",r2score*100)
```

```
R2Score 90.1599948196672
```