In [1]:

```
import pandas as pd
from sklearn import linear_model
from sklearn.preprocessing import StandardScaler
scale=StandardScaler()
```

In [2]:

```
df=pd.read_csv("C:/Users/SAGNIK SAMANTA/Downloads/data.csv")
df.head()
```

Out[2]:

	Car	Model	Volume	Weight	CO2
0	Toyoty	Aygo	1000	790	99
1	Mitsubishi	Space Star	1200	1160	95
2	Skoda	Citigo	1000	929	95
3	Fiat	500	900	865	90
4	Mini	Cooper	1500	1140	105

In [3]:

```
df.columns
```

Out[3]:

```
Index(['Car', 'Model', 'Volume', 'Weight', 'CO2'], dtype='object')
```

In [4]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 36 entries, 0 to 35
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   Car     36 non-null     object
 1   Model   36 non-null     object
 2   Volume  36 non-null     int64 
 3   Weight  36 non-null     int64 
 4   CO2     36 non-null     int64 
dtypes: int64(3), object(2)
memory usage: 1.5+ KB
```

In [5]:

```
df.describe()
```

Out[5]:

	Volume	Weight	CO2
count	36.000000	36.000000	36.000000
mean	1611.111111	1292.277778	102.027778
std	388.975047	242.123889	7.454571
min	900.000000	790.000000	90.000000
25%	1475.000000	1117.250000	97.750000
50%	1600.000000	1329.000000	99.000000
75%	2000.000000	1418.250000	105.000000
max	2500.000000	1746.000000	120.000000

In [6]:

```
X=df[['Weight', 'Volume']]
y=df["CO2"]
```

In [7]:

```
Scaled_X=scale.fit_transform(X)
print(Scaled_X)
```

```
[[-2.10389253 -1.59336644]
 [-0.55407235 -1.07190106]
 [-1.52166278 -1.59336644]
 [-1.78973979 -1.85409913]
 [-0.63784641 -0.28970299]
 [-1.52166278 -1.59336644]
 [-0.76769621 -0.55043568]
 [ 0.3046118  -0.28970299]
 [-0.7551301  -0.28970299]
 [-0.59595938 -0.0289703 ]
 [-1.30803892 -1.33263375]
 [-1.26615189 -0.81116837]
 [-0.7551301  -1.59336644]
 [-0.16871166 -0.0289703 ]
 [ 0.14125238 -0.0289703 ]
 [ 0.15800719 -0.0289703 ]
 [ 0.3046118  -0.0289703 ]
 [-0.05142797  1.53542584]
 [-0.72580918 -0.0289703 ]
 [ 0.14962979  1.01396046]
 [ 1.2219378  -0.0289703 ]
 [ 0.5685001   1.01396046]
 [ 0.3046118   1.27469315]
 [ 0.51404696 -0.0289703 ]
 [ 0.51404696  1.01396046]
 [ 0.72348212 -0.28970299]
 [ 0.8281997   1.01396046]
 [ 1.81254495  1.01396046]
 [ 0.96642691 -0.0289703 ]
 [ 1.72877089  1.01396046]
 [ 1.30990057  1.27469315]
 [ 1.90050772  1.01396046]
 [-0.23991961 -0.0289703 ]
 [ 0.40932938 -0.0289703 ]
 [ 0.47215993 -0.0289703 ]
 [ 0.4302729   2.31762392]]
```

In [8]:

```
reg_model=linear_model.LinearRegression()
reg_model.fit(Scaled_X,y)
```

Out[8]:

```
LinearRegression()
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [9]:

```
print(reg_model.coef_)
```

```
[1.80269333 2.99358608]
```

Now we have a regression object that are ready to predict CO2 values based on a car's weight and volume: predict the CO2 emission of a car where the weight is 2300kg, and the volume is 1300cm3 :

In [10]:

```
scaled=scale.transform([[2300,1300]])
Predicted_CO2=reg_model.predict([scaled[0]])
print("Predicted_CO2 :",Predicted_CO2)
```

```
Predicted_CO2 : [107.2087328]

C:\Users\SAGNIK SAMANTA\anaconda3\Lib\site-packages\sklearn\base.py:464:
UserWarning: X does not have valid feature names, but StandardScaler was
fitted with feature names
  warnings.warn(
```

In [ ]:

In [ ]: