```
In [1]: import matplotlib.pyplot as plt
        import pandas as pd
        import pylab as pl
        import numpy as np
        %matplotlib inline
```

```
In [2]: df = pd.read_csv("FuelConsumptionCo2.csv")

        df.head()
```

Out[2]:

	MODELYEAR	MAKE	MODEL	VEHICLECLASS	ENGINESIZE	CYLINDERS	TRANSMISSION	FUELTYPE	FUELCONSUMPTION_CITY	FUELCONSUMPTIOI
0	2014	ACURA	ILX	COMPACT	2.0	4	AS5	Z	9.9	
1	2014	ACURA	ILX	COMPACT	2.4	4	M6	Z	11.2	
2	2014	ACURA	ILX HYBRID	COMPACT	1.5	4	AV7	Z	6.0	
3	2014	ACURA	MDX 4WD	SUV - SMALL	3.5	6	AS6	Z	12.7	
4	2014	ACURA	RDX AWD	SUV - SMALL	3.5	6	AS6	Z	12.1	

```
In [3]: cdf = df[['ENGINESIZE','CYLINDERS','FUELCONSUMPTION_CITY','FUELCONSUMPTION_HWY','FUELCONSUMPTION_COMB','CO2EMISSIONS']]
        cdf.head(9)
```

Out[3]:

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_CITY	FUELCONSUMPTION_HWY	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	9.9	6.7	8.5	196
1	2.4	4	11.2	7.7	9.6	221
2	1.5	4	6.0	5.8	5.9	136
3	3.5	6	12.7	9.1	11.1	255
4	3.5	6	12.1	8.7	10.6	244
5	3.5	6	11.9	7.7	10.0	230
6	3.5	6	11.8	8.1	10.1	232
7	3.7	6	12.8	9.0	11.1	255
8	3.7	6	13.4	9.5	11.6	267

```
In [4]: plt.scatter(cdf.ENGINESIZE, cdf.CO2EMISSIONS,  color='blue')
        plt.xlabel("Engine size")
        plt.ylabel("Emission")
        plt.show()
```
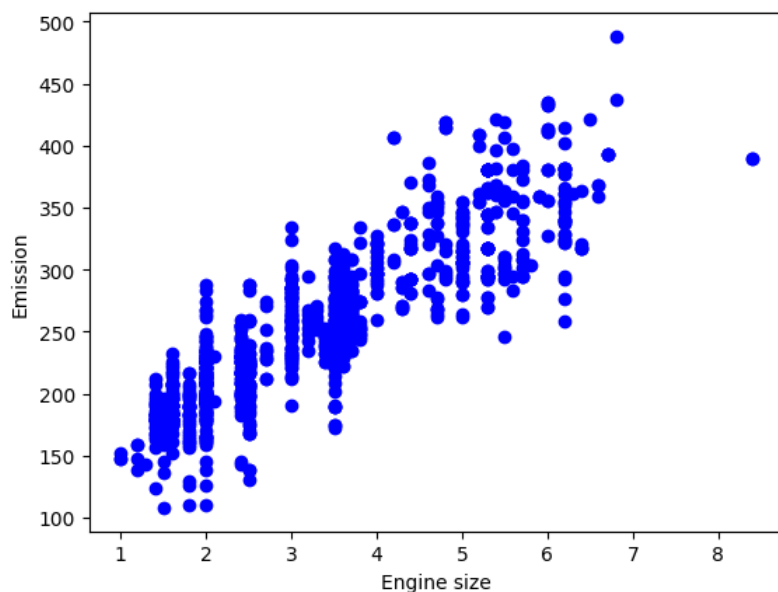


```
In [5]: msk = np.random.rand(len(df)) < 0.8
        train = cdf[msk]
        test = cdf[~msk]
```

```
In [6]: plt.scatter(train.ENGINESIZE, train.CO2EMISSIONS,  color='blue')
        plt.xlabel("Engine size")
        plt.ylabel("Emission")
        plt.show()
```
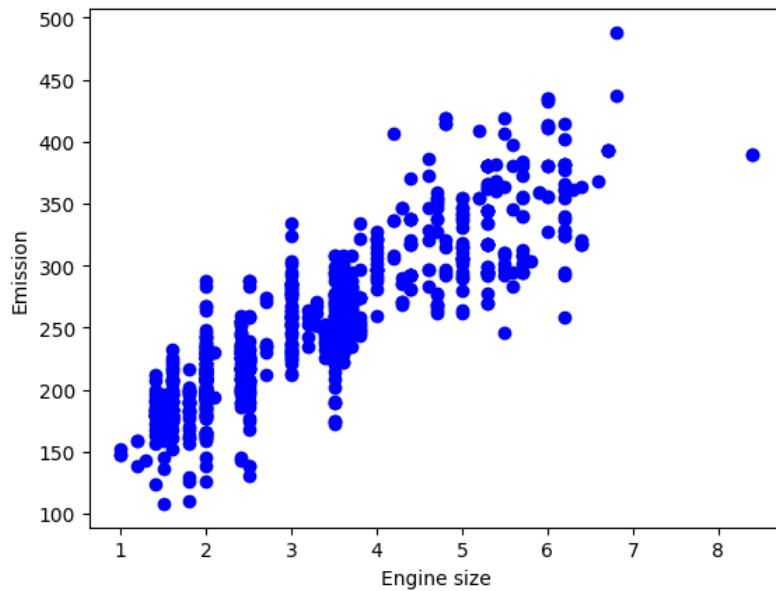
```
In [7]: from sklearn import linear_model
        regr = linear_model.LinearRegression()
        x = np.asanyarray(train[['ENGINESIZE','CYLINDERS','FUELCONSUMPTION_COMB']])
        y = np.asanyarray(train[['CO2EMISSIONS']])
        regr.fit (x, y)
        print ('Coefficients: ', regr.coef_)
```

```
Coefficients:  [[11.20939729  6.78451406  9.887996  ]]
```

```
In [8]: y_hat= regr.predict(test[['ENGINESIZE','CYLINDERS','FUELCONSUMPTION_COMB']])
        x = np.asanyarray(test[['ENGINESIZE','CYLINDERS','FUELCONSUMPTION_COMB']])
        y = np.asanyarray(test[['CO2EMISSIONS']])
        print("Residual sum of squares: %.2f"
              % np.mean((y_hat - y) ** 2))

        print('Variance score: %.2f' % regr.score(x, y))
```

```
Residual sum of squares: 588.11
Variance score: 0.86
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:432: UserWarning: X has feature names, but LinearRegression was fitte
d without feature names
  warnings.warn(
```

```
In [ ]:
```