Step 1: Data Preprocessing

In [1]:
```python
#import library
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [2]:
```python
#import dataset and split into x and y column
dataset = pd.read_csv('dataset.csv')
X = dataset.iloc[ : ,   : 1 ].values
Y = dataset.iloc[ : , 1 ].values
```

In [3]:
```python
#import dataset split library
from sklearn.model_selection import train_test_split
#split dataset
X_train, X_test, Y_train, Y_test = train_test_split( X, Y, test_size = 0.25, random_state = 0)
```

Step 2: Fitting Simple Linear Regression Model to the training set

In [4]:
```python
#import linearRegression library
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor = regressor.fit(X_train, Y_train)
```

Step 3: Predecting the Result

In [5]:
```python
Y_pred = regressor.predict(X_test)
print(Y_pred)
```

```
[11.40738019  3.7738426   6.19580767  3.58397466  2.54453841  2.16843058
  3.59183543  8.31870051  3.55241063  5.92551796  5.08864782  2.01242444
  4.76381644  6.06253731  6.26776399  2.24582898  2.53329145  2.62375083
  3.01908731  3.7379249   2.87626309 21.62759632  4.36315881  6.99591357
  4.47187937]
```

Step 4: Visualization

Visualising the Training results

In [6]:
```python
plt.scatter(X_train , Y_train, color = 'B')
plt.plot(X_train , regressor.predict(X_train), color ='G')
```

Out[6]: [<matplotlib.lines.Line2D at 0x26618ca4748>]
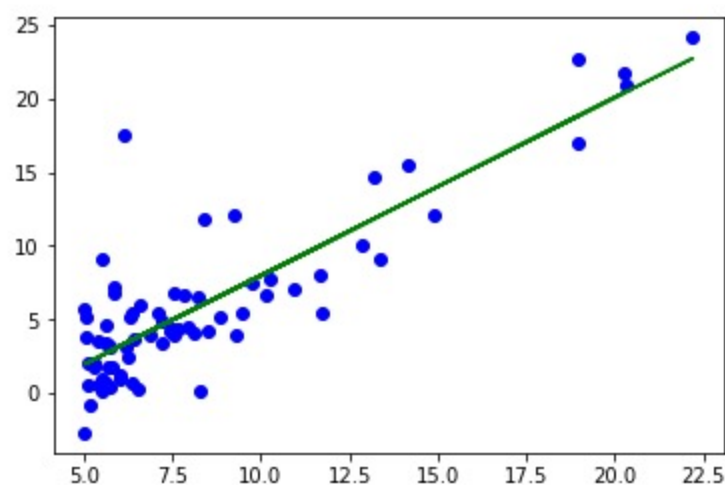
Visualizing the test results

In [7]:
```python
plt.scatter(X_test , Y_test, color = 'B')
plt.plot(X_test , regressor.predict(X_test), color ='G')
```

Out[7]: [<matplotlib.lines.Line2D at 0x26618f94b00>]