Predicting the percentage based on student study

In [35]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

Reading the file

In [36]:
```
df = pd.read_csv("C:\\Users\\Brijesh gautam\\Downloads\\student_scores.csv")
```

In [37]:
```
df.head()
```

Out[37]:

	Hours	Scores
0	2.5	21
1	5.1	47
2	3.2	27
3	8.5	75
4	3.5	30

In [38]:
```
pd.set_option('display.max_rows', None)
```

In [39]:
```
df
```

Out[39]:

	Hours	Scores
0	2.5	21
1	5.1	47
2	3.2	27
3	8.5	75
4	3.5	30
5	1.5	20
6	9.2	88
7	5.5	60
8	8.3	81
9	2.7	25
10	7.7	85
11	5.9	62
12	4.5	41
13	3.3	42
14	1.1	17
15	8.9	95
16	2.5	30
17	1.9	24
18	6.1	67
19	7.4	69
20	2.7	30
21	4.8	54
22	3.8	35
23	6.9	76
24	7.8	86

In [40]:
```
df['Hours'].unique()
```

Out[40]:
```
array([2.5, 5.1, 3.2, 8.5, 3.5, 1.5, 9.2, 5.5, 8.3, 2.7, 7.7, 5.9, 4.5,
       3.3, 1.1, 8.9, 1.9, 6.1, 7.4, 4.8, 3.8, 6.9, 7.8])
```

In [43]:
```
df.describe()
```

Out[43]:

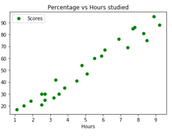
	Hours	Scores
count	25.000000	25.000000
mean	5.012000	51.480000
std	2.525094	25.286887
min	1.100000	17.000000
25%	2.700000	30.000000
50%	4.800000	47.000000
75%	7.400000	75.000000
max	9.200000	95.000000

In [46]:
```
df.dtypes
```

Out[46]:
```
Hours      float64
Scores       int64
dtype: object
```

Plotting the graph to see the relation between x and y axis

In [52]:
```
df.plot("Hours", "Scores", style = 'go')
plt.title("Percentage vs Hours studied")
plt.xlabel("Hours")
plt.ylabel("Scores")
plt.show()
```



It is a postive coreallton as point goes from low value to high value

Now we have to split the data by using the Scikit-Learn built-in model train_test_split

In [57]:
```
from sklearn.model_selection import train_test_split
```

Here assigning the independent and dependent values, as y is dependent value

In [84]:
```
x = df.drop(['Scores'],axis = 1)
y = df.Scores
```

Splitting the data

In [85]:
```
x_train,x_test,y_train,y_test = train_test_split(x, y ,test_size = 0.2, random_state = 0)
```

80% data is taken into x_train , remaining 20% data taken into x_test

In [86]:
```
x_train
```

Out[86]:

	Hours
22	3.8
17	1.9
24	7.8
23	6.9
14	1.1
1	5.1
10	7.7
13	3.3
8	8.3
6	9.2
18	6.1
4	3.5
9	2.7
7	5.5
20	2.7
3	8.5
0	2.5
21	4.8
15	8.9
12	4.5

In [116]:
```
x_test
```

Out[116]:

	Hours
5	1.5
2	3.2
19	7.4
16	2.5
11	5.9

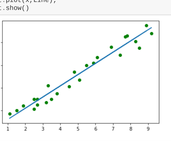
Fitting the data into Linear Regression model

In [88]:
```
from sklearn.linear_model import LinearRegression

regressor = LinearRegression()
regressor.fit(x_train, y_train)
y_pred = regressor.predict(x_test)
```

Applying the formula to get the line for better understanding

In [89]:
```
Line = regressor.coef_*x + regressor.intercept_
plt.scatter(x,y, color= 'green')
plt.plot(x,Line);
plt.show()
```



In [90]:
```
print(x_test)
```

```
   Hours
5    1.5
2    3.2
19   7.4
16   2.5
11   5.9
```

After fitting the data into model, we got the predicted values shown below comparing to actual data

In [91]:
```
df_1 = pd.DataFrame({"Actual" : y_test, "Predicted" : y_pred})
df_1
```

Out[91]:

	Actual	Predicted
5	20	16.884145
2	27	33.732261
19	69	75.357018
16	30	26.794801
11	62	60.491033

Predicitng the percentage based on student study

In [102]:
```
hours = [[9.25]]
predicted = regressor.predict(hours)
print("If student study 9.25hrs/day ,The percentage will be ",predicted)
```

```
If student study 9.25hrs/day ,The percentage will be  [93.69173249]
```