In [1]:
```
import numpy as np
import pandas as pd
```

In [2]:
```
dataset=pd.read_csv(r"D:\ML_Course\Works_on_python\Linear_Regression\Salary_Data.csv")
```

In [3]:
```
dataset.head()
```

Out[3]:

	YearsExperience	Salary
0	1.1	39343.0
1	1.3	46205.0
2	1.5	37731.0
3	2.0	43525.0
4	2.2	39891.0

In [4]:
```
dataset.isnull().any()
```

Out[4]:
```
YearsExperience    False
Salary             False
dtype: bool
```

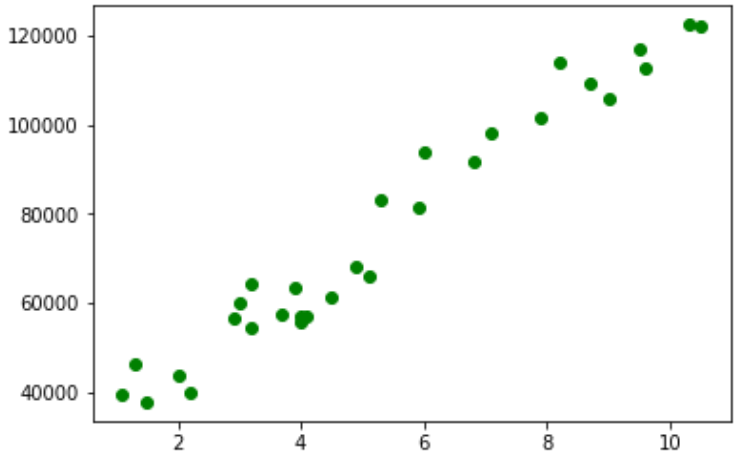
In [5]:
```
dataset.describe() # to get statistical relation
```

Out[5]:

	YearsExperience	Salary
count	30.000000	30.000000
mean	5.313333	76003.000000
std	2.837888	27414.429785
min	1.100000	37731.000000
25%	3.200000	56720.750000
50%	4.700000	65237.000000
75%	7.700000	100544.750000
max	10.500000	122391.000000

In [7]:
```
import matplotlib.pyplot as plt
plt.scatter(dataset["YearsExperience"],dataset["Salary"],color="green")
```

Out[7]:
```
<matplotlib.collections.PathCollection at 0x25b7bbc3748>
```



In [10]:
```
x=dataset.iloc[:,0:1].values #input
y=dataset.iloc[:,1:2].values #output
```

In [11]:
```
x
```

Out[11]:
```
array([[ 1.1],
       [ 1.3],
       [ 1.5],
       [ 2. ],
       [ 2.2],
       [ 2.9],
       [ 3. ],
       [ 3.2],
       [ 3.2],
       [ 3.7],
       [ 3.9],
       [ 4. ],
       [ 4. ],
       [ 4.1],
       [ 4.5],
       [ 4.9],
       [ 5.1],
       [ 5.3],
       [ 5.9],
       [ 6. ],
       [ 6.8],
       [ 7.1],
       [ 7.9],
       [ 8.2],
       [ 8.7],
       [ 9. ],
       [ 9.5],
       [ 9.6],
       [10.3],
       [10.5]])
```

In [12]:
```
y
```

Out[12]:
```
array([[ 39343.],
       [ 46205.],
       [ 37731.],
       [ 43525.],
       [ 39891.],
       [ 56642.],
       [ 60150.],
       [ 54445.],
       [ 64445.],
       [ 57189.],
       [ 63218.],
       [ 55794.],
       [ 56957.],
       [ 57081.],
       [ 61111.],
       [ 67938.],
       [ 66029.],
       [ 83088.],
       [ 81363.],
       [ 93940.],
       [ 91738.],
       [ 98273.],
       [101302.],
       [113812.],
       [109431.],
       [105582.],
       [116969.],
       [112635.],
       [122391.],
       [121872.]])
```

In [13]:
```
x.shape
```

Out[13]:
```
(30, 1)
```

In [14]:
```
y.shape
```

Out[14]:
```
(30, 1)
```

In [16]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=0)
```

In [17]:
```
x_test.shape
```

Out[17]:
```
(6, 1)
```

In [18]:
```
x_train.shape
```

Out[18]:
```
(24, 1)
```

In [19]:
```
y_test.shape
```

Out[19]:
```
(6, 1)
```

In [20]:
```
y_train.shape
```

Out[20]:
```
(24, 1)
```

In [21]:
```
#train algo

from sklearn.linear_model import LinearRegression
lr=LinearRegression()
lr.fit(x_train,y_train)
```

Out[21]:
```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None,
         normalize=False)
```

In [22]:
```
#test algo

y_pred=lr.predict(x_test)
```

In [25]:
```
x_test
```

Out[25]:
```
array([[ 1.5],
       [10.3],
       [ 4.1],
       [ 3.9],
       [ 9.5],
       [ 8.7]])
```

In [26]:
```
y_test
```

Out[26]:
```
array([[ 37731.],
       [122391.],
       [ 57081.],
       [ 63218.],
       [116969.],
       [109431.]])
```

In [24]:
```
y_pred # see the values of y_test to see accuracy
```

Out[24]:
```
array([[ 40748.96184072],
       [122699.62295594],
       [ 64961.65717022],
       [ 63099.14214487],
       [115249.56285456],
       [107799.50275317]])
```

In [27]:
```
from sklearn.metrics import r2_score
accuracy=r2_score(y_test,y_pred)
```

In [28]:
```
accuracy
```

Out[28]:
```
0.988169515729126
```

In [29]:
```
y=lr.predict([[12]]) # a person working for 12 years will have this salary
```

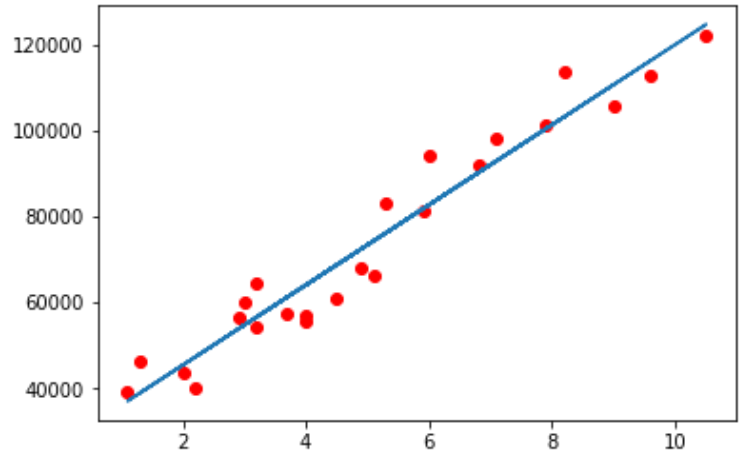
In [30]:
```
y # prediction for random value
```

Out[30]:
```
array([[138531.00067138]])
```

In [31]:
```
#best fit line for x_train

plt.scatter(x_train,y_train,color="red")#x_train is question and y_train is ans
plt.plot(x_train,lr.predict(x_train))#input x_train and predicted value
```

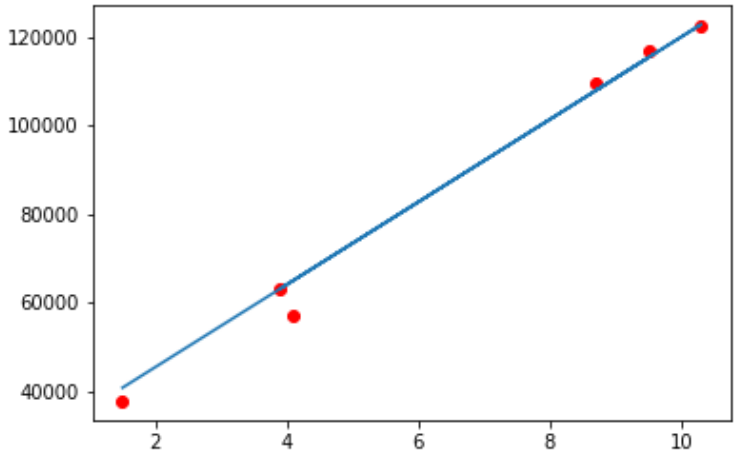
Out[31]:
```
[<matplotlib.lines.Line2D at 0x25b7e2094e0>]
```



In [34]:
```
#best fit line for x_test

plt.scatter(x_test,y_test,color="red")
plt.plot(x_test,y_pred)
```

Out[34]:
```
[<matplotlib.lines.Line2D at 0x25b7e6ac278>]
```



In [ ]:
```

```