Imports and data import

In [1]:

```
import numpy as np
import pandas as pd
import math
import matplotlib.pyplot as plt
import numpy as np
from scipy.stats import pearsonr
from sklearn import linear_model
from sklearn.metrics import mean_squared_error, r2_score

data = pd.read_csv("regression.txt", sep=",", header = None, names=["Populati
```

Question 1

In [2]:

```
data.tail()
```

Out[2]:

	Population	Profit
92	5.8707	7.20290
93	5.3054	1.98690
94	8.2934	0.14454
95	13.3940	9.05510
96	5.4369	0.61705

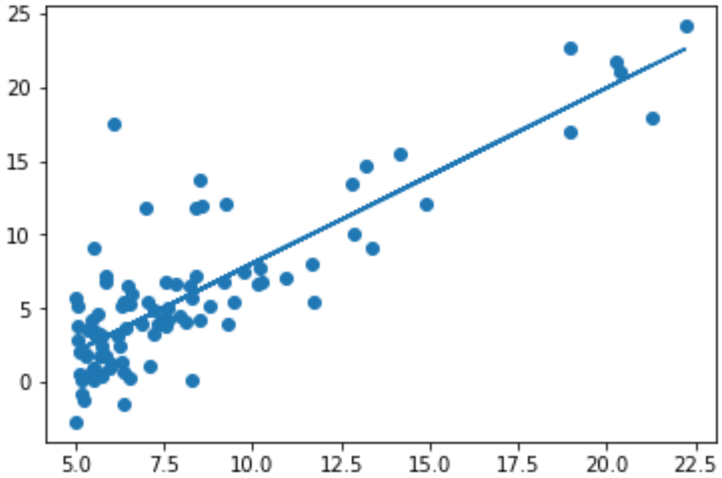
Question 2

In linear regression, there are some assumptions, including that the relationship between X and Y is linear The scatter plot below shows that there is a linear relationship between the population and the profit, and its corresponding pearson's correlation coefficient confirms a strong positive linear relationship between the two variables. Therefore, linear regression is appropriate to predict the profit based on the population.

In [3]:

```
population = data.iloc[:,0].values
profit = data.iloc[:,1].values
a, b = np.polyfit(population, profit, 1)
plt.scatter(population, profit)
plt.plot(population, a*population+b)
plt.show()

corr, _ = pearsonr(population, profit)
print('Pearsons correlation: %.3f' % corr)
```



```
Pearsons correlation: 0.838
```

Question 3

In [4]:

```
regr = linear_model.LinearRegression()
regr.fit(population.reshape(-1, 1), profit)

# The coefficients
print("Coefficients: \n", regr.coef_)
# The mean squared error
print("Mean squared error: %.2f" % mean_squared_error(profit, population))
# The coefficient of determination: 1 is perfect prediction
print("Coefficient of determination (R squared): %.2f" % r2_score(profit, pop
```

```
Coefficients: 
 [1.19303364]
Mean squared error: 14.89
Coefficient of determination (R squared): 0.50
```

Prediction

In [6]:

```
population_test = np.array(12.7423)
population_y_pred = regr.predict(population_test.reshape(1, -1))
print("Profit in city with population", 12.7423, "is", population_y_pred)
```

```
Profit in city with population 12.7423 is [11.30621173]
```