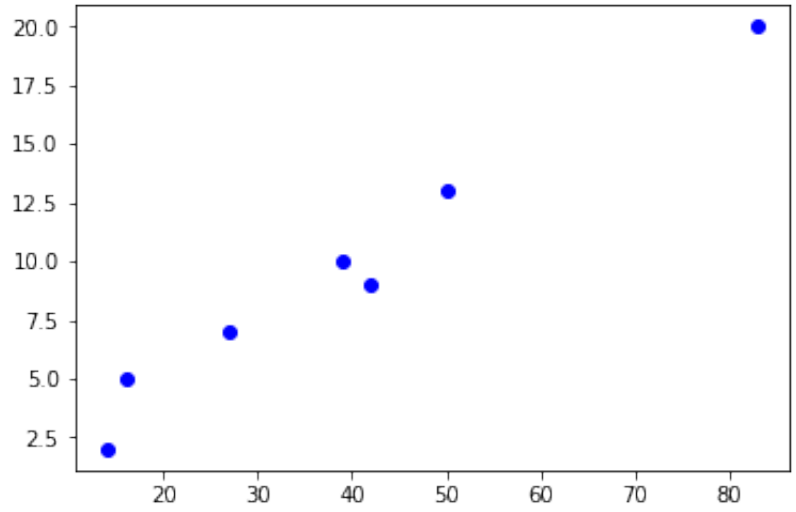
In [54]:
```
import numpy as np
from matplotlib import pyplot as plt

%matplotlib inline
```

In [55]:
```
x_data = np.array([14,16,27,42,83,50,39])
y_data = np.array([2,5,7,9,20,13,10])
plt.plot(x_data,y_data,'bo')
```

Out[55]:
```
[<matplotlib.lines.Line2D at 0x106865d68>]
```



In [56]:
```
n = len(x_data)
x_mean = np.sum(x_data)/n
y_mean = np.sum(y_data)/n
print(x_mean,y_mean)
```

```
38.714285714285715 9.428571428571429
```

In [57]:
```
S_xx = np.sum((x_data-x_mean)**2)
S_yy = np.sum((y_data-y_mean)**2)
S_xy = np.sum((x_data-x_mean)*(y_data-y_mean))
print(S_xx,S_yy,S_xy)
```

```
3363.428571428571 205.71428571428572 819.8571428571428
```

In [58]:
```
theta_1 = S_xy/S_xx
theta_0 = y_mean - theta_1*x_mean
print(theta_0,theta_1)
```

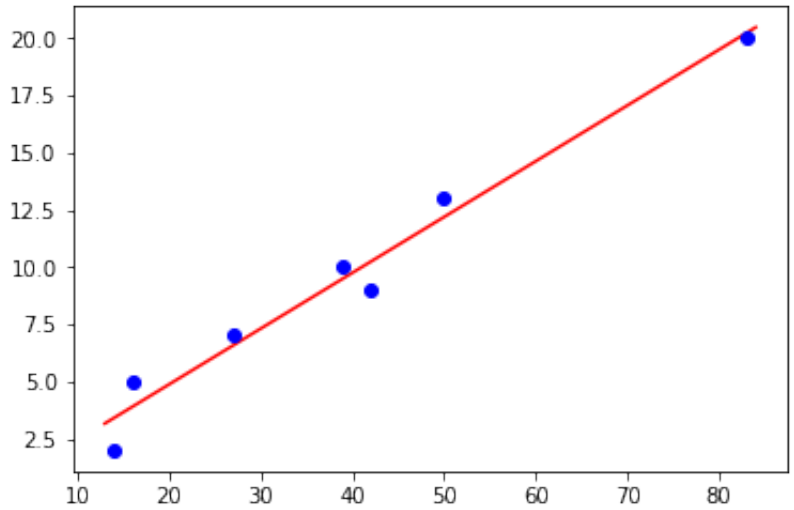
```
-0.008282364933741349 0.24375637104994904
```

In [59]:
```
y = lambda x: theta_1*x+theta_0

def plot_line(y, data_points):
    x_vals = [i for i in range(int(min(data_points)-1),int(max(data
_points))+2)]
    y_vals = [y(x) for x in x_vals]
    plt.plot(x_vals,y_vals,'r')

plot_line(y,x_data)
plt.plot(x_data,y_data,'bo')
```

Out[59]:
```
[<matplotlib.lines.Line2D at 0x1066ab080>]
```



In [60]:
```
SSE = S_yy - 2*theta_1*S_xy + theta_1**2*S_xx
#print(SSE)
SSE = np.sum((y_data - y(x_data))**2)
print("SSE = ",SSE)
SST = np.sum((y_data - y_mean)**2)
print("SST = ",SST)
R_squared = 1 - SSE/SST
print("R_squared = ",R_squared)
```

```
SSE =  5.868883792048929
SST =  205.71428571428572
R_squared =  0.9714707037886511
```

In [61]:
```
r_corr = S_xy/(np.sqrt(S_xx*S_yy))
print(r_corr,np.sqrt(R_squared))
```

```
0.9856321341091974 0.9856321341091975
```

In [62]:
```
MSE = SSE/(n-2)
print("MSE = ", MSE)
```

```
MSE =  1.1737767584097858
```

In [63]:
```
S_E = np.sqrt(MSE)
print("S_E = ", S_E)
```

```
S_E =  1.0834097832352196
```

In [64]:
```
S_theta_0 = S_E*np.sqrt(1/n+x_mean**2/(np.sum(x_data**2)-n*x_mean**
2))
print("S_theta_0 = ", S_theta_0)
#print("x^2 = ", np.sum(x_data**2))
#print("n*x_mean^2 = ", n*x_mean**2)
#print(S_E*np.sqrt(1/n+x_mean**2/(np.sum(x_data**2)-n*x_mean**2)))
```

```
S_theta_0 =  0.8311049969529885
```

In [65]:
```
S_theta_1 = S_E/np.sqrt(np.sum(x_data**2)-n*x_mean**2)
print("S_theta_1 = ", S_theta_1)
```

```
S_theta_1 =  0.018681065725824817
```

In [66]:
```
from scipy.stats import t
t_quant = t.ppf(0.95, n-2)
print("t_quant = ", t_quant)
```

```
t_quant =  2.015048372669157
```

In [67]:
```
print(theta_1-t_quant*S_theta_1,theta_1+t_quant*S_theta_1)
t.interval(0.9, n-2, loc=theta_1, scale=S_theta_1)
```

```
0.20611311995940018 0.2813996221404979
```

Out[67]:
```
(0.20611311995940018, 0.2813996221404979)
```

In [68]:
```
print(theta_0-t_quant*S_theta_0,theta_0+t_quant*S_theta_0)
t.interval(0.9, n-2, loc=theta_0, scale=S_theta_0)
```

```
-1.6829991365610655 1.6664344066935828
```

Out[68]:
```
(-1.6829991365610661, 1.6664344066935828)
```