In [29]:

```python
import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
data=pd.read_csv("data1.csv")
print(data.shape)
data.head()
```
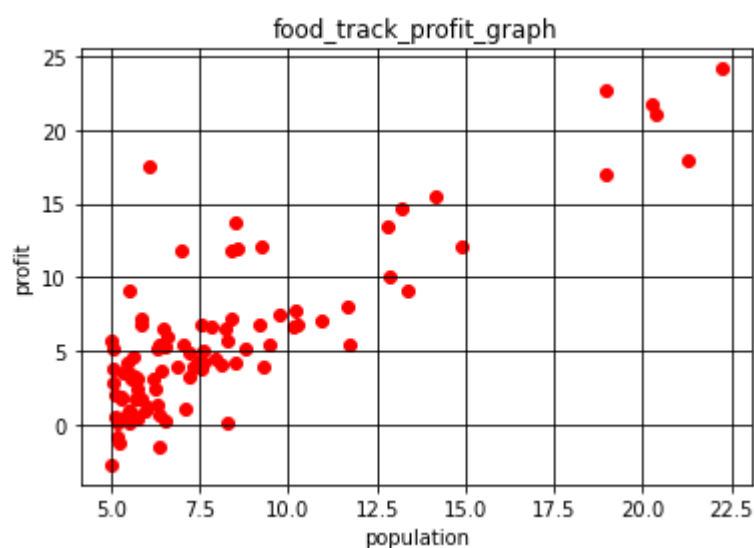
(97, 2)

Out[29]:

	population	profit
0	6.1101	17.5920
1	5.5277	9.1302
2	8.5186	13.6620
3	7.0032	11.8540
4	5.8598	6.8233

In [30]:

```python
x=data[["population"]].values
y=data[["profit"]].values
%matplotlib inline
plt.scatter(x,y,c="r",label="scatter_data")
plt.xlabel("population")
plt.ylabel("profit")
plt.title("food_track_profit_graph")
plt.grid(True,color='k')
plt.show()
```

In [31]:

```
k=LinearRegression()
k.fit(x,y)
print('c value:',k.intercept_)
```
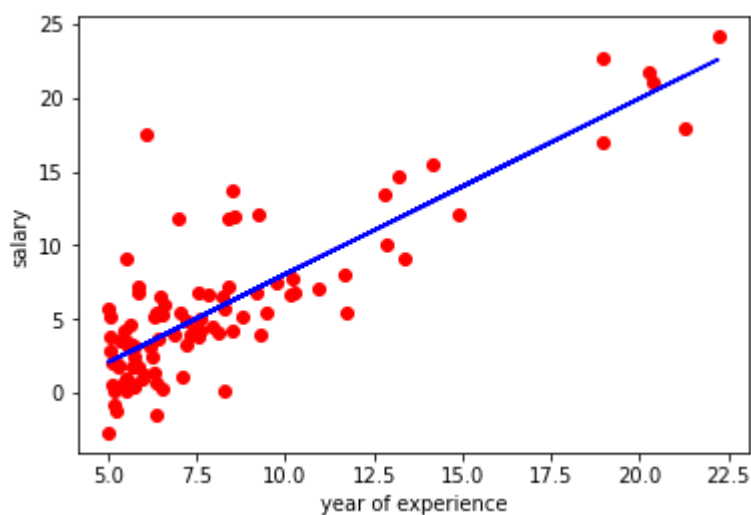
c value: [-3.89578088]

In [33]:

```
print("m values:",k.coef_)
```

m values: [[1.19303364]]

In [35]:

```
y_pred=k.predict(x)
plt.scatter(x,y,color='red')
plt.plot(x,y_pred,color='blue')
plt.xlabel("year of experience")
plt.ylabel("salary")
plt.show()
```


In [37]:

```
from sklearn.metrics import r2_score
r_sq=r2_score(y,y_pred)
r_sq
```

Out[37]:

0.7020315537841397

In [38]:

```
from sklearn.metrics import mean_squared_error
rmse=mean_squared_error(y,y_pred)
rmse
```

Out[38]:

8.953942751950358

In [42]:

```python
n1=4.5
n2=6.5
print("profit from 45000 people is",k.predict([[n1]])*10000,'$')
print("profit from 45000 people is",k.predict([[n2]])*10000,'$')
```

```
profit from 45000 people is [[14728.70520541]] $
profit from 45000 people is [[38589.37808921]] $
```

In [43]:

```python
x=np.array([5,6,7,8,9,10])
y=np.array([18,20,22,25,28,32])
print("profit from 45000 people is",k.predict([[n1]])*10000,'$')
print("profit from 45000 people is",k.predict([[n2]])*10000,'$')
```

```
profit from 45000 people is [[14728.70520541]] $
profit from 45000 people is [[38589.37808921]] $
```

In [52]:

```
y_pred
```

Out[52]:

Out[52]:

```
array([[ 3.39377399],
       [ 2.6989512 ],
       [ 6.26719552],
       [ 4.45927234],
       [ 3.09515767],
       [ 6.10530086],
       [ 5.02381586],
       [ 6.33818102],
       [ 3.84247394],
       [ 2.13452698],
       [ 2.91727635],
       [13.00234766],
       [ 2.94507404],
       [ 6.13572322],
       [ 2.833764  ],
       [ 2.52202431],
       [ 3.69835548],
       [ 2.22460102],
       [ 3.77494824],
       [ 4.53992141],
       [ 3.48802365],
       [20.28701109],
       [ 2.65409313],
       [ 3.65146926],
       [ 2.74333205],
       [18.70624151],
       [11.40845471],
       [ 9.17628876],
       [11.82363042],
       [22.59314512],
       [ 2.37050903],
       [ 3.96559502],
       [ 7.13763287],
       [ 3.13333475],
       [ 5.90033768],
       [ 5.56903223],
       [ 5.7629002 ],
       [ 2.79272364],
       [11.41799898],
       [ 3.68403908],
       [ 2.55483273],
       [ 4.31527318],
       [10.07225703],
       [ 2.99243747],
       [ 5.43934948],
       [ 4.56652606],
       [ 2.1531383 ],
       [ 3.02548451],
       [10.06271276],
       [ 2.71553436],
       [ 5.09993141],
       [ 2.43648379],
       [ 4.96118159],
       [ 5.17497322],
       [ 3.65946258],
       [ 3.69060076],
       [ 3.58955081],
       [ 2.83257096],
```

In [ ]:

In [ ]:

In [ ]: