In [1]:

```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

```
Matplotlib is building the font cache; this may take a moment.
```

In [10]:

```
data=pd.read_csv("book.txt")
print(data.shape)
```

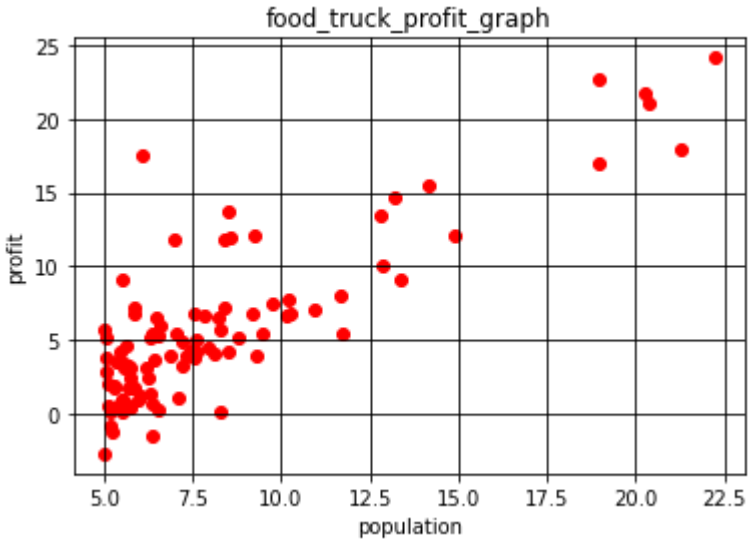
```
(97, 2)
```

```

```

In [11]:

```
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_truck_profit_graph")
plt.grid(True,color='k')
plt.show()
```



In [12]:

```
k=LinearRegression()
k.fit(x,y)

```

Out[12]:

```
LinearRegression()
```

In [16]:

```
print("c value:",k.intercept_)
```

```
c value: [-3.89578088]
```

In [17]:

```
print("\n value:",k.coef_)
```

```
 value: [[1.19303364]]
```

In [23]:

```
y_pred=k.predict(x)

```

In [27]:

```
plt.scatter(x,y,color='red')
plt.plot(x,y_pred,color="blue")
plt.title("salary vs experience_(training_set)")
plt.xlabel("years of experience ")
plt.ylabel("salary")
plt.show()
```

In [19]:

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

Out[19]:

```
8.953942751950358
```

In [32]:

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

Out[32]:

```
0.7020315537841397
```

In [34]:

```
n1=4.5
n2=6.5
print("profit from 4500 people city is:",k.predict([[n1]])*10000,'$')
print("profit from 6500 people city is:",k.predict([[n2]])*10000,'$')
```

```
profit from 4500 people city is: [[14728.70520541]] $
profit from 6500 people city is: [[38589.37808921]] $
```

Develop a Linear Regression model for the given

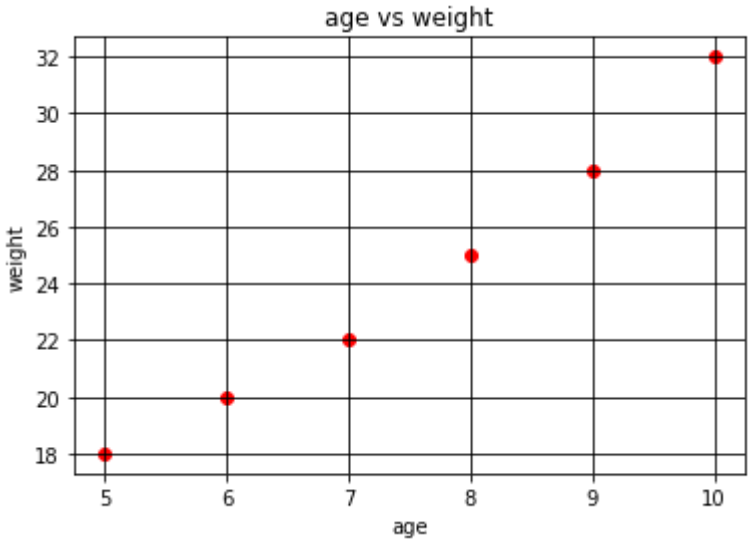
age and weight data. x=np.array([5,6,7,8,9,10]) #age y=np.array([18,20,22,25,28,32]) #weight

In [36]:

```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
```

In [42]:

```
x=np.array([[5,6,7,8,9,10]])
y=np.array([[18,20,22,25,28,32]])
%matplotlib inline
plt.scatter(x,y,c='r',label='scatter_data')
plt.xlabel("age")
plt.ylabel("weight")
plt.title("age vs weight")
plt.grid(True,color='k')
plt.show()
```



In [43]:

```
k=LinearRegression()
k.fit(x,y)

```

Out[43]:

```
LinearRegression()
```

In [44]:

```
print("c value:",k.intercept_)
```

```
c value: [18. 20. 22. 25. 28. 32.]
```

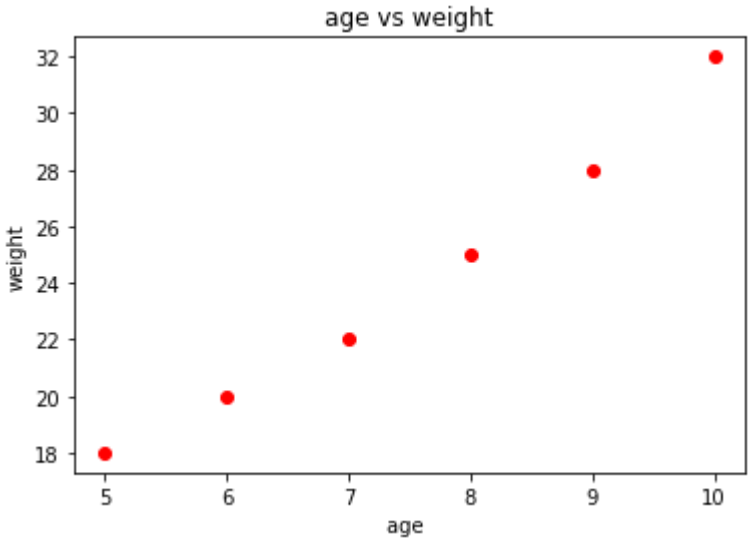
In [45]:

```
print("\n value:",k.coef_)
```

```
 value: [[0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0.]]
```

In [47]:

```
y_pred=k.predict(x)
plt.scatter(x,y,color='red')
plt.plot(x,y_pred,color="blue")
plt.title("age vs weight")
plt.xlabel("age ")
plt.ylabel("weight")
plt.show()
```



In [48]:

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

Out[48]:

```
0.0
```

In [49]:

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

```
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\metrics\_regression.py:
682: UndefinedMetricWarning: R^2 score is not well-defined with less than
two samples.
  warnings.warn(msg, UndefinedMetricWarning)
```

Out[49]:

```
nan
```

In [ ]:

```

```