In [136…
```
#Load libraries
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [174…
```
#Load Data
df=pd.read_excel("C:/Users/jcadmin/Downloads/insurance_2.xlsx")
```

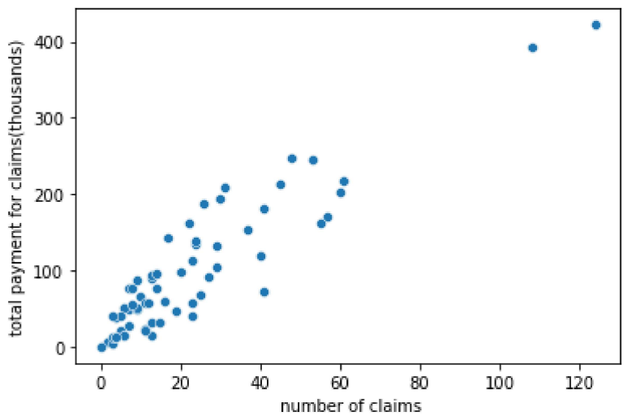
In [175…
```
df.head() #view the first 5 rows of the data
```

Out[175…

	number of claims	total payment for claims(thousands)
0	108	392.5
1	19	46.2
2	13	15.7
3	124	422.2
4	40	119.4

In [176…
```
sns.scatterplot(data=df, x=' number of claims', y='total payment for claims(thousands)')
```

Out[176…
```
<AxesSubplot:xlabel=' number of claims', ylabel='total payment for claims(thousands)'>
```



In [177…
```
#Split Data
feature_cols=[' number of claims']
X = df[feature_cols]
y = df['total payment for claims(thousands)']

print('Shape of X = ', X.shape)
print('Shape of y= ', y.shape)
```

```
Shape of X =  (63, 1)
Shape of y=  (63,)
```

In [178…
```
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=51)

print('Shape of X_train = ', X_train.shape)
print('Shape of X_test = ', X_test.shape)
print('Shape of y_train = ', y_train.shape)
print('Shape of y_test = ', y_test.shape)
```

```
Shape of X_train =  (50, 1)
```

```
Shape of X_test =  (13, 1)
Shape of y_train =  (50,)
Shape of y_test =  (13,)
```

In [179…
```
#Linear regression
from sklearn.linear_model import LinearRegression
lr=LinearRegression()
lr.fit(X_train,y_train)
```

Out[179…
```
LinearRegression()
```

In [180…
```
lr.coef_
```

Out[180…
```
array([3.44905649])
```

In [181…
```
lr.intercept_
```

Out[181…
```
16.430342199367033
```

In [182…
```
#Predict the total claim amt
lr.predict([[15]])
```

Out[182…
```
array([68.16618955])
```

In [183…
```
lr.score(X, y)
```

Out[183…
```
0.8322450120594063
```

In [184…
```
#Evaluating the model
y_pred=lr.predict(X_test)
y_pred
```

Out[184…
```
array([119.9020369 ,  30.22656816, 388.9284431 ,  47.47185061,
       40.57373763, 144.04543232,  44.02279412,  47.47185061,
       75.06430253,  61.26807657,  54.36996359, 154.39260179,
       81.96241551])
```

In [185…
```
y_test
```

Out[185…
```
56    194.5
29     38.1
0     392.5
58     87.4
14     48.8
43    152.8
48     76.1
22     52.1
50    142.1
40     89.9
12     23.5
4     119.4
1      46.2
Name: total payment for claims(thousands), dtype: float64
```

In [186…
```
from sklearn.metrics import mean_squared_error
import numpy as np
```

In [188…
```
mse= mean_squared_error(y_test,y_pred)
root_mse=np.sqrt(mse)

print('MSE = ',mse)
print('Root_mse = ',root_mse)
```

```
MSE =  1322.9615294508005
Root_mse =  36.37253812219874
```

In [ ]: