In [23]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn import metrics
from sklearn.model_selection import train_test_split
```

In [24]:
```
data =pd.read_csv('Sindrom Zoom Fatique Bagi Mahasiswa.csv')
data
```

Out[24]:

	Nama lengkap	Umur	Jenis Kelamin	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13
0	Naufal Raihan	19 - 22 tahun	Laki - laki	3	3	2	3	3	3	3	2	2	3	3	2	1
1	VaroVero	19 - 22 tahun	Laki - laki	3	2	2	1	2	1	1	2	2	1	1	1	1
2	Varrel Rizalvyno Zaidan Firdaus	19 - 22 tahun	Laki - laki	3	2	2	2	3	3	3	2	2	1	2	3	2
3	Citra Annisaa Nurul Ain	19 - 22 tahun	Perempuan	3	3	3	3	2	3	2	2	2	1	2	2	1
4	Fitrian Alif Putra Iantono	19 - 22 tahun	Laki - laki	3	2	3	2	1	3	2	3	2	2	2	3	3
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
104	Siti Rohani	19 - 22 tahun	Perempuan	3	3	2	2	2	3	2	2	2	2	3	2	3
105	SAFIRA AULIA FARHANI	< 19 tahun	Perempuan	2	3	3	3	2	3	2	2	1	3	2	2	3
106	Galuh Alifia Damayanti	19 - 22 tahun	Perempuan	3	3	2	2	2	2	1	3	2	1	2	3	1
107	Gue	19 - 22 tahun	Perempuan	2	3	3	3	3	3	3	3	3	3	3	3	3
108	Sherly Sukanto	19 - 22 tahun	Perempuan	3	3	2	3	2	3	3	2	3	3	3	3	3

109 rows × 16 columns

In [25]:
```
x = data[['P1','P2','P3','P4','P5','P6','P7','P8','P10','P11','P12','P13']].values
y = data['P9'].values
```

In [26]:
```
x_train, x_test, y_train, y_test = train_test_split(x, y, train_size = 0.9, test_size = 0.1, random_state =0)
```

In [27]:
```
regressor = LinearRegression()
regressor.fit(x_train, y_train)
```

Out[27]:
```
LinearRegression()
```

In [28]:
```
print(regressor.intercept_)
```

```
0.5889551466086071
```

In [29]:
```
print(regressor.coef_)
```

```
[ 0.08810205 -0.07097558  0.05078346 -0.00352036  0.21615227  0.10422511
  0.15270402  0.03284655  0.21518029 -0.33745741  0.05414245  0.23923824]
```

In [30]:
```
coeff_df = pd.DataFrame(regressor.coef_, ['P1','P2','P3','P4','P5','P6','P7','P8','P10','P11','P12','P13'], columns=['Coefficient'])
coeff_df
```

Out[30]:

	Coefficient
P1	0.088102
P2	-0.070976
P3	0.050783
P4	-0.003520
P5	0.216152
P6	0.104225
P7	0.152704
P8	0.032847
P10	0.215180
P11	-0.337457
P12	0.054142
P13	0.239238

In [31]:
```
Y_pred = regressor.predict(x_test)
```

In [32]:
```
df = pd.DataFrame({'Actual': y_test, 'Predicted': Y_pred})
df
```

Out[32]:

	Actual	Predicted
0	2	2.199856
1	3	1.760710
2	3	2.598038
3	2	2.471943
4	2	2.176022
5	2	2.071797
6	3	2.513436
7	3	2.509936
8	2	1.431925
9	2	2.068277
10	2	2.120910

In [33]:
```
print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, Y_pred))
#print('Mean Squared Error:', metrics.mean_squared_error(y_test, Y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test, Y_pred)))
```

```
Mean Absolute Error: 0.3904327375794796
Root Mean Squared Error: 0.5058855096258789
```

In [ ]:
```

```