Import Librarys

In [52]:
```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
```

Read Data

In [53]:
```
data = pd.read_csv("1.01. Simple linear regression.csv")
```

In [54]:
```
data.head()
```

Out[54]:

	SAT	GPA
0	1714	2.40
1	1664	2.52
2	1760	2.54
3	1685	2.74
4	1693	2.83

In [55]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 84 entries, 0 to 83
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   SAT     84 non-null     int64  
 1   GPA     84 non-null     float64
dtypes: float64(1), int64(1)
memory usage: 1.4 KB
```

In [56]:
```
data.describe()
```

Out[56]:

	SAT	GPA
count	84.000000	84.000000
mean	1845.273810	3.330238
std	104.530661	0.271617
min	1634.000000	2.400000
25%	1772.000000	3.190000
50%	1846.000000	3.380000
75%	1934.000000	3.502500
max	2050.000000	3.810000

Clean Data

In [57]:
```
data.isna().sum()
```

Out[57]:
```
SAT    0
GPA    0
dtype: int64
```

In [58]:
```
x = data[["SAT"]]
y = data[["GPA"]]
```

In [59]:
```
x
```

Out[59]:

	SAT
0	1714
1	1664
2	1760
3	1685
4	1693
...	...
79	1936
80	1810
81	1987
82	1962
83	2050

84 rows × 1 columns

In [60]:
```
y
```

Out[60]:

	GPA
0	2.40
1	2.52
2	2.54
3	2.74
4	2.83
...	...
79	3.71
80	3.71
81	3.73
82	3.76
83	3.81

84 rows × 1 columns

Split The Data

In [61]:
```
x_train , x_test , y_train , y_test = train_test_split(x , y , test_size = 0.2 , ra
```

In [62]:
```
x_train
```

Out[62]:

	SAT
23	1687
56	1730
27	1821
8	1792
79	1936
...	...
75	2015
20	1761
9	1850
28	2020
15	1872

67 rows × 1 columns

In [63]:
```
y_train
```

Out[63]:

	GPA
23	3.21
56	3.47
27	3.28
8	3.01
79	3.71
...	...
75	3.62
20	3.19
9	3.01
28	3.28
15	3.17

67 rows × 1 columns

In [64]:
```
x_test
```

Out[64]:

	SAT
41	1850
21	1722
48	1857
42	1966
0	1714
82	1962
59	1891
76	1997
36	1808
12	1735
65	1832
47	1956
2	1760
5	1670
1	1664
45	1925
31	1934

In [65]:
```
y_test
```

Out[65]:

	GPA
41	3.38
21	3.19
48	3.41
42	3.38
0	2.40
82	3.76
59	3.48
76	3.64
36	3.32
12	3.08
65	3.52
47	3.40
2	2.54
5	2.91
1	2.52
45	3.40
31	3.28

Prepair The Model

In [66]:
```
model = LinearRegression()
```

In [67]:
```
model
```

Out[67]:
```
LinearRegression
LinearRegression()
```

Fit The Data

In [68]:
```
model.fit(x_train , y_train)
```

Out[68]:
```
LinearRegression
LinearRegression()
```

Preddictions

In [69]:
```
prediction = model.predict(x_test)
```

In [ ]:

In [70]:
```
prediction[1]
```

Out[70]:
```
array([3.19701933])
```

In [71]:
```
y_test.iloc[1]
```

Out[71]:
```
GPA    3.19
Name: 21, dtype: float64
```

Cost Function

In [72]:
```
cost = mean_squared_error(y_test , prediction)
```

In [73]:
```
cost
```

Out[73]:
```
0.1013650964889309
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: