In [22]:
```
#importing libraries
import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
import numpy as np
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_sc
```

In [23]:
```
df=pd.read_csv("C:\\Users\\DAVIS\\Desktop\\Soft\\500_Person_Gender_Height_
```

In [24]:
```
df.head(6)
```

Out[24]:

	Gender	Height	Weight	Index
0	Male	174	96	4
1	Male	189	87	2
2	Female	185	110	4
3	Female	195	104	3
4	Male	149	61	3
5	Male	189	104	3

In [25]:
```
x=df[['Height', 'Index']]
y=df['Weight']
x
```

Out[25]:

	Height	Index
0	174	4
1	189	2
2	185	4
3	195	3
4	149	3
...	...	...
495	150	5
496	184	4
497	141	5
498	150	5
499	173	5

500 rows × 2 columns

In [26]:
```
y[:5]
```

Out[26]:
```
0     96
1     87
2    110
3    104
4     61
Name: Weight, dtype: int64
```

In [27]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_st
```

In [28]:
```
x_train.shape
```

Out[28]:
```
(400, 2)
```

In [29]:
```
model=LinearRegression()
model.fit(x_train,y_train)
```

Out[29]:
```
LinearRegression()
```
In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [30]:
```
y_pred= model.predict(x_test)
y_pred[:5]
```

Out[30]:
```
array([104.22610236,  74.07616719, 115.18646577, 119.40199016,
       130.36235358])
```

In [34]:
```
r2= r2_score(y_test, y_pred)
print('R2-score:', r2)
```

```
R2-score: 0.747160996343226
```

In [35]:
```
#optimizing
from sklearn.model_selection import GridSearchCV
```

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

model = LinearRegression()
```

In [36]:
```
param={
    'fit_intercept': [True, False],
    'positive': [True, False],
    'copy_X': [True, False],
    'n_jobs': [None],
}
```

In [38]:
```
#cross validation
grid_search = GridSearchCV(model, param, cv=5)
grid_search.fit(x_train, y_train)
```

Out[38]:
```
GridSearchCV(cv=5, estimator=LinearRegression(),
             param_grid={'copy_X': [True, False],
                         'fit_intercept': [True, False], 'n_jobs': [Non
e],
                         'positive': [True, False]})
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [39]:
```
best_para = grid_search.best_params_
print(best_para)
```

```
{'copy_X': True, 'fit_intercept': True, 'n_jobs': None, 'positive': Fals
e}
```

In [47]:
```
#best model
optimized_model = LinearRegression(**best_para)
optimized_model.fit(x_train, y_train)
y_pred= optimized_model.predict(x_test)
y_pred[:6]
```

Out[47]:
```
array([104.22610236,  74.07616719, 115.18646577, 119.40199016,
       130.36235358, 119.72262239])
```

In [55]:
```
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_sc
mean_abs=mean_absolute_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
r2,mean_abs
```

Out[55]:
```
(0.747160996343226, 12.473232690715083)
```

In [ ]:
```

```