Program 8 : Write a prgram to demonstrate Multiple Linear Regression

In [2]:

**import** numpy **as** np

**import** pandas **as** pd

**import** matplotlib.pyplot **as** plt

**from** sklearn.datasets **import** make\_regression

**from** sklearn.model\_selection **import** train\_test\_split

**from** sklearn.linear\_model **import** LinearRegression

**from** sklearn.metrics **import** mean\_squared\_error

*# Generate random regression dataset with multiple features*

X, y **=** make\_regression(n\_samples**=**100, n\_features**=**3, noise**=**10, random\_state**=**42)

*# Split the dataset into training and testing sets*

X\_train, X\_test, y\_train, y\_test **=** train\_test\_split(X, y, test\_size**=**0.2, random\_stat

*# Instantiate and fit the multiple linear regression model*

model **=** LinearRegression() model**.**fit(X\_train, y\_train)

*# Make predictions on the test data*

y\_pred **=** model**.**predict(X\_test)

*# Calculate and print the Mean Squared Error*

mse **=** mean\_squared\_error(y\_test, y\_pred) print("Mean Squared Error:", mse)

*# Plot actual values*

plt**.**scatter(range(len(y\_test)), y\_test, label**=**'Actual values') plt**.**xlabel("Index")

plt**.**ylabel("Actual values") plt**.**title("Actual Values") plt**.**legend()

plt**.**show()

*# Plot predicted values*

plt**.**scatter(range(len(y\_test)), y\_pred, color**=**'red', label**=**'Predicted values') plt**.**xlabel("Index")

plt**.**ylabel("Predicted values") plt**.**title("Predicted Values") plt**.**legend()

plt**.**show()

Mean Squared Error: 123.84680824798083



