```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
```

```
X = np.array([
[1,1],
[2,1],
[3,2],
[4,3],
[5,3]
])
y = np.array([20, 40, 50, 65, 80])
```

```
model = LinearRegression()
model.fit(X,y)
```

```
▾ LinearRegression
LinearRegression()
```

```
y_pred = model.predict(X)
```

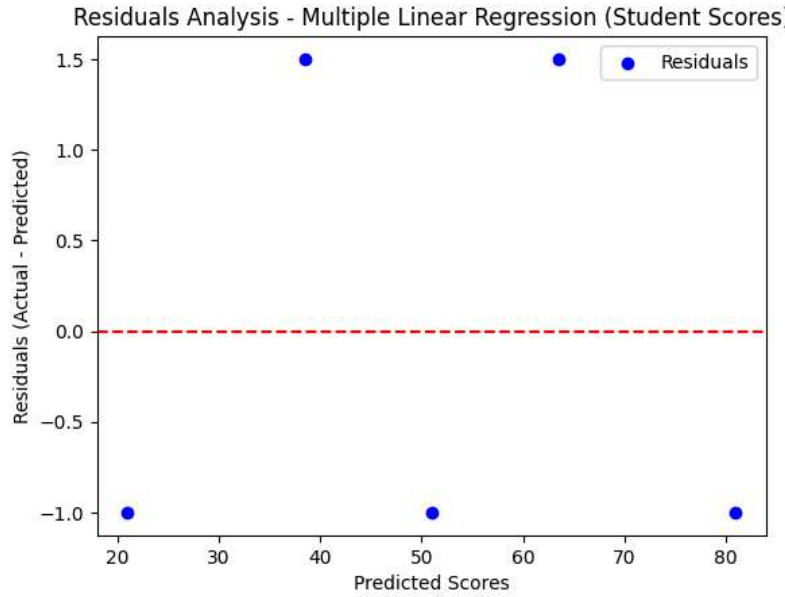
```
residuals = y - y_pred
```

```
mae = mean_absolute_error(y, y_pred)
mse = mean_squared_error(y,  y_pred)
rmse = np.sqrt(mse)
r2 = r2_score(y, y_pred)
```

```
print("Coefficients (b1=Hours, b2 = Practice_Tests):", model.coef_)
print("Intercept (b0):", model.intercept_)
print("Mean Absolute Error (MAE):", mae)
print("Mean Squared Error (MSE):", mse)
print("Root Mean Squared Error (RMSE):", rmse)
print("R2 score (R2 score):",r2)
```

```
Coefficients (b1=Hours, b2 = Practice_Tests): [17.5 -5. ]
Intercept (b0): 8.500000000000007
Mean Absolute Error (MAE): 1.2000000000000015
Mean Squared Error (MSE): 1.5000000000000013
Root Mean Squared Error (RMSE): 1.2247448713915896
R2 score (R2 score): 0.9964622641509434
```

```
plt.scatter(y_pred, residuals, color='blue', label = "Residuals")
plt.axhline(y=0,color = 'red', linestyle="--")
plt.xlabel("Predicted Scores")
plt.ylabel("Residuals (Actual - Predicted)")
plt.title("Residuals Analysis - Multiple Linear Regression (Student Scores)")
plt.legend()
plt.show()
```



Start coding or generate with AI.