Linear Regression

```python
In [1]:
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

# --- Step 1: Create sample dataset ---
data = {
    "Hours_Studied": [1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
    "Test_Score":    [35, 40, 50, 55, 60, 65, 70, 75, 85, 90]
}

df = pd.DataFrame(data)

# --- Step 2: Split features and target ---
X = df[["Hours_Studied"]]  # feature must be 2D
y = df["Test_Score"]       # target

# --- Step 3: Train-test split ---
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, rando

# --- Step 4: Create Linear Regression model ---
model = LinearRegression()
model.fit(X_train, y_train)

# --- Step 5: Predictions ---
y_pred = model.predict(X_test)

# --- Step 6: Evaluation ---
print("Coefficients:", model.coef_)
print("Intercept:", model.intercept_)
print("Mean Squared Error:", mean_squared_error(y_test, y_pred))
print("R^2 Score:", r2_score(y_test, y_pred))

# --- Step 7: Plot regression line ---
plt.scatter(X, y, color="blue", label="Actual Data")
plt.plot(X, model.predict(X), color="red", linewidth=2, label="Regression Line
plt.xlabel("Hours Studied")
plt.ylabel("Test Score")
plt.title("Linear Regression: Hours Studied vs Test Score")
plt.legend()
plt.show()
```

```
Coefficients: [5.77586207]
Intercept: 30.732758620689648
Mean Squared Error: 5.218861474435167
R^2 Score: 0.9896911378282762
```

Linear Regression: Hours Studied vs Test Score