

In [10]:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.tree import DecisionTreeRegressor
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
```

In [2]:

```
df = pd.read_csv('aabbcc.csv')
```

In [4]:

```
X = df[['R&D Spend', 'Administration', 'Marketing Spend']]
y = df['Profit']
```

In [11]:

```
linear_reg_model = LinearRegression()
decision_tree_model = DecisionTreeRegressor()
random_forest_model = RandomForestRegressor()
```

In [12]:

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

In [13]:

```
linear_reg_model.fit(X_train, y_train)
decision_tree_model.fit(X_train, y_train)
random_forest_model.fit(X_train, y_train)
```

Out[13]:

```
▼ RandomForestRegressor
RandomForestRegressor()
```

In [14]:

```
def evaluate_model(model, X_test, y_test):
    y_pred = model.predict(X_test)
    mse = mean_squared_error(y_test, y_pred)
    mae = mean_absolute_error(y_test, y_pred)
    r2 = r2_score(y_test, y_pred)
    return mse, mae, r2

linear_reg_mse, linear_reg_mae, linear_reg_r2 = evaluate_model(linear_reg_model, X_test, y_test)
decision_tree_mse, decision_tree_mae, decision_tree_r2 = evaluate_model(decision_tree_model, X_test, y_test)
random_forest_mse, random_forest_mae, random_forest_r2 = evaluate_model(random_forest_model, X_test, y_test)
```

In [15]:

```
models = {
    'Linear Regression': (linear_reg_mse, linear_reg_mae, linear_reg_r2),
    'Decision Tree': (decision_tree_mse, decision_tree_mae, decision_tree_r2),
    'Random Forest': (random_forest_mse, random_forest_mae, random_forest_r2)
}

best_model = min(models, key=lambda k: models[k][0])

print("Model Metrics:")
for model_name, (mse, mae, r2) in models.items():
    print(f"{model_name}:")
    print(f"    Mean Squared Error (MSE): {mse}")
    print(f"    Mean Absolute Error (MAE): {mae}")
    print(f"    R-squared (R2): {r2}")
    print()

print(f"The best model is: {best_model}")
```

Model Metrics:

Linear Regression:

Mean Squared Error (MSE): 80926321.2229516
Mean Absolute Error (MAE): 6979.1522523704
R-squared (R2): 0.900065308303732

Decision Tree:

Mean Squared Error (MSE): 135346231.77903003
Mean Absolute Error (MAE): 8616.894999999999
R-squared (R2): 0.8328629827639696

Random Forest:

Mean Squared Error (MSE): 92586033.09363148
Mean Absolute Error (MAE): 6706.5196399999495
R-squared (R2): 0.8856669062331182

The best model is: Linear Regression

In []: