```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error, r2_score

def load_data():
    # Sample synthetic dataset
    data = {
        'RAM': [4, 6, 8, 12, 4, 8, 6, 12, 8, 6],
        'ROM': [64, 128, 256, 256, 32, 64, 128, 512, 128, 64],
        'Battery': [3000, 3500, 4000, 4500, 3000, 5000, 3500, 6000, 4000, 3700],
        'Camera': [12, 16, 48, 64, 8, 12, 16, 108, 48, 20],
        'Screen': [5.5, 6.0, 6.3, 6.7, 5.0, 6.2, 6.1, 6.9, 6.5, 5.8],
        'Price': [15000, 20000, 30000, 40000, 12000, 17000, 19000, 60000, 35000,
18000]
    }
    return pd.DataFrame(data)

def train_model(df):
    X = df[['RAM', 'ROM', 'Battery', 'Camera', 'Screen']]
    y = df['Price']

    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=42)

    model = RandomForestRegressor(n_estimators=100, random_state=42)
    model.fit(X_train, y_train)

    y_pred = model.predict(X_test)

    print(f"\nModel R2 Score: {r2_score(y_test, y_pred):.2f}")
    print(f"Mean Squared Error: {mean_squared_error(y_test, y_pred):.2f}")

    return y_test, y_pred

def visualize(y_test, y_pred):
    plt.plot(y_test.values, label='Actual Price', marker='o')
    plt.plot(y_pred, label='Predicted Price', marker='x')
    plt.title('Mobile Price Prediction - Actual vs Predicted')
    plt.xlabel('Sample Index')
    plt.ylabel('Price (INR)')
    plt.legend()
    plt.grid(True)
    plt.show()

def main():
    print("=== Mobile Price Prediction using Random Forest ===")
    df = load_data()
    y_test, y_pred = train_model(df)
    visualize(y_test, y_pred)

if __name__ == "__main__":
    main()
```