Here's a **simple example using scikit-learn** to build a model that predicts House Price

---

✅ Goal

IN Example: House Price Prediction in India (Simulated)

📌 We'll Use:

- `scikit-learn` for model training (`LinearRegression`)
- `pandas` for a custom dataset
- Features relevant to Indian housing

Features We'll Simulate:-

Feature	Type	Description
`Area`	Numeric	Built-up area in square feet
`BHK`	Integer	Number of bedrooms
`Bathrooms`	Integer	Number of bathrooms
`LocationScore`	Numeric	1–10 score for location (proximity to school, metro, etc.)
`Age`	Integer	Age of the property in years

🧠 Step-by-Step Code (Python + scikit-learn)

✅ Required Python Packages

Package	Purpose	Install Command
`scikit-learn`	Machine learning models (like `DecisionTreeClassifier`)	`pip install scikit-learn`
`numpy`	Numerical operations (used by scikit-learn internally)	`pip install numpy`
`pandas` (optional)	Data handling (if you use DataFrames later)	`pip install pandas`

🔧 How to Install

You can install everything in one line using pip:

```
pip install scikit-learn numpy pandas
```

✅ Full Working Code:

```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error

# Step 1: Simulated dataset (India-based features)
data = {
    'Area': [1200, 1000, 1500, 800, 950, 1350, 1600, 1100, 1400, 1250],
    'BHK': [3, 2, 3, 2, 2, 3, 4, 2, 3, 3],
    'Bathrooms': [2, 1, 2, 1, 2, 2, 3, 1, 2, 2],
    'LocationScore': [8, 6, 9, 5, 7, 8, 10, 6, 9, 7],
    'Age': [5, 10, 3, 12, 8, 4, 2, 9, 3, 6],
    'Price': [75, 60, 90, 50, 58, 85, 100, 65, 92, 80]  # in Lakhs
}

df = pd.DataFrame(data)

# Step 2: Feature matrix and target variable
X = df[['Area', 'BHK', 'Bathrooms', 'LocationScore', 'Age']]
y = df['Price']  # in Lakhs
```

```
# Step 3: Split the dataset
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Step 4: Train the model
model = LinearRegression()
model.fit(X_train, y_train)

# Step 5: Predict on test set and calculate RMSE manually
y_pred = model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
rmse = mse ** 0.5
print(f"\nModel RMSE: {rmse:.2f} Lakhs")

# Step 6: Get user input for prediction
print("\n🔢 Enter the following details to estimate house price (in India):")

try:
    area = float(input("Built-up Area (in sq ft): "))
    bhk = int(input("Number of Bedrooms (BHK): "))
    baths = int(input("Number of Bathrooms: "))
    location_score = float(input("Location Score (1-10): "))
    age = int(input("Age of Property (in years): "))

    input_features = pd.DataFrame([[area, bhk, baths, location_score, age]],
                                  columns=['Area', 'BHK', 'Bathrooms',
'LocationScore', 'Age'])
    predicted_price = model.predict(input_features)[0]

    print(f"\n💰 Estimated House Price: ₹{predicted_price:.2f} Lakhs")
except ValueError:
    print("❌ Invalid input. Please enter numeric values.")
```

output:-

Model RMSE: 5.19 Lakhs

🔢 Enter the following details to estimate house price (in India):

Built-up Area (in sq ft): 1300

Number of Bedrooms (BHK): 3

Number of Bathrooms: 2

Location Score (1-10): 9

Age of Property (in years): 3

Estimated House Price: ₹79.53 Lakhs