

Title: Real Estate Price Prediction – Task 1

Subtitle: Using AI/ML to predict house prices

BAMMIDI SIVA KUMAR

07/08/2025

# What We Are Doing

We want the computer to guess how much a house costs.

To do this, we will give the computer:

- The city the house is in
- How many bedrooms it has
- How big it is (square feet)
- How old it is (years)
- How good its extra features are (like lift, parking, garden)

Then the computer will tell the price.

# our example data

City	Bedrooms	Size	Age	features score	price
Palasa	3	1200	5	8	75
Vizag	2	900	3	7	60
Srikakulam	4	1500	10	9	85

we will use this table to teach the computer

# Screenshot the table output

The screenshot displays the Anaconda Notebook interface in a web browser. The browser's address bar shows the URL `localhost:8888/lab/tree/real_estate_task1.ipynb?`. The notebook is titled `real_estate_task1.ipynb` and is running in `Python [conda env:base] *` mode.

The notebook contains two code cells:

```
[1]: #real estate price prediction
    ### created by : siva kumar

[2]: #import pandas library helps us work with table like data
    import pandas as pd#from now onwards i will call pandas as pd in my code its like shortcut simple

    data = {
        'Location': ['palasa','vizag','srikakulam'],
        'Bedrooms': [3, 2, 4],
        'Area_sqft': [1200, 900, 1500],
        'Age_Years': [5, 3, 10],
        'Amenities_Score': [8, 7, 9],
        'Price_Lakhs': [75, 60, 85]
    }

    df = pd.DataFrame(data)

    df

    X = df[['Bedrooms', 'Area_sqft', 'Age_Years', 'Amenities_Score']]
    y = df['Price_Lakhs']

    from sklearn.linear_model import LinearRegression

    model = LinearRegression()

    model.fit(X, y)

    predicted_prices = model.predict(X)

    comparison_df = pd.DataFrame({
```

The left sidebar shows the Anaconda Toolbox with options for creating a new project or notebook, and the Anaconda Cloud section. The right sidebar contains common and advanced tools. The bottom status bar indicates the notebook is in `Simple` mode, showing `0` errors and `1` warning, and the current position is `Ln 1, Col 1`.

# output screenshot

The screenshot displays the Anaconda Lab web interface. The browser address bar shows the URL `localhost:8888/lab/tree/real_estate_task1.ipynb?`. The left sidebar contains the Anaconda Toolbox with options like 'Create a New Project', 'Create a New Notebook', and 'My Projects'. The main area shows a Jupyter Notebook titled 'real\_estate\_task1.ipynb' with a Python kernel. The notebook contains a code cell with a print statement and a table of data.

```
# print(predicted_prices)
# This line shows the guessed prices on the screen.
# It helps us see what the machine predicted so we can check if it's close to the real prices.

# We are NOT giving the price.
# We ARE Letting the machine Learn from old data and guess for new data.
# That's what Machine Learning really means - Learning and predicting!

# -----
```

	Location	Actual Price (Lakhs)	Predicted Price (Lakhs)
0	palasa	75	75.0
1	vizag	60	60.0
2	srikakulam	85	85.0

The bottom status bar indicates the current mode is 'Command' and the file is 'real\_estate\_task1.ipynb'.

# How the computer learns

- We store the table in Python using pandas.
- We split the table into two parts:
  - -  $X \rightarrow$  All the details (bedrooms, size, age, features)
  - -  $y \rightarrow$  The price
- We use a tool called Linear Regression to find patterns.

```
df = pd.DataFrame(data)

df

X = df[['Bedrooms', 'Area_sqft', 'Age_Years', 'Amenities_Score']]
y = df['Price_Lakhs']

from sklearn.linear_model import LinearRegression

model = LinearRegression()

model.fit(X, y)
```

# Guessing the price

City	Actual price	Predicted price
palasa	75	75.0
vizag	60	60.0
srikakulam	85	85.0

# What we learned

- Computers can learn from examples
- More examples = better predictions
- Can be used for many things, not just house prices