

Building a house price prediction system with Scikit-learn plus a simple user interface is a great beginner-to-intermediate project. I'll walk you through it step by step so you can actually build and run it.

```
D:\ai>cd flask  
D:\ai\flask>pip install flask joblib scikit-learn pandas  
Requirement already satisfied: flask in c:\users\big_data\an
```

Train a House Price Model:-

CSV dataset instead of manually creating a DataFrame. Here's a clean, working example.

❑ 1. Create a CSV File

Create a file named `house_data.csv` in your project folder.

Example CSV content:

```
area,bedrooms,price  
1000,2,300000  
1200,2,350000  
1500,3,450000  
1800,3,480000  
2000,3,500000  
2200,4,550000  
2500,4,600000  
2700,4,650000  
3000,5,700000
```

❑ 2. Updated Python Code **train_model.py** (Using CSV)

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
import joblib

# Load dataset from CSV
data = pd.read_csv('house_data.csv')

# Show first few rows (optional)
print(data.head())

# Features & target
X = data[['area', 'bedrooms']]
y = data['price']

# Split dataset
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)

# Create model
model = LinearRegression()

# Train model
model.fit(X_train, y_train)

# Save model
joblib.dump(model, 'house_model.pkl')

print("Model trained and saved successfully!")
```

▶❑ Run the Code

In the same folder, run:

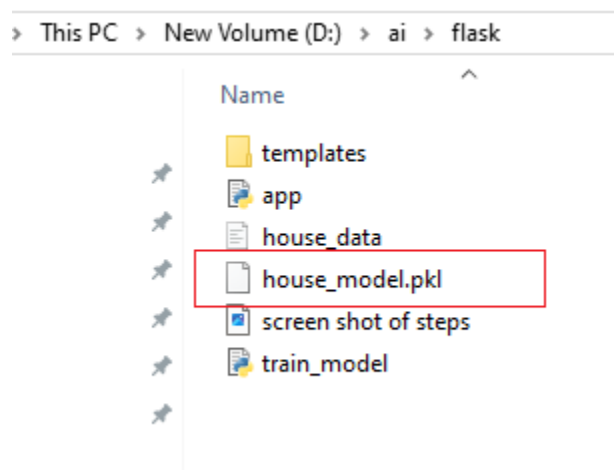
```
python train_model.py
```

```

D:\ai\flask>python train_model.py
   area  bedrooms   price
0   1000         2  300000
1   1200         2  350000
2   1500         3  450000
3   1800         3  480000
4   2000         3  500000
Model trained and saved successfully!

```

Open your flask folder you will see house_model.pkl file as shown below



Let's connect your trained model to a **web app using Flask** so users can input values and get predictions.

Project Structure

Create this folder setup:

```

falsk/
├── app.py (backend flask code)
├── house_model.pkl (model)
├── house_data.csv (data)
├── templates/
│   └── index.html (userinterface)

```

1. Flask Backend (`app.py`)

```
from flask import Flask, render_template, request
import joblib

# Initialize app
app = Flask(__name__)

# Load trained model
model = joblib.load('house_model.pkl')

# Home route
@app.route('/')
def home():
    return render_template('index.html')

# Prediction route
@app.route('/predict', methods=['POST'])
def predict():
    try:
        area = float(request.form['area'])
        bedrooms = int(request.form['bedrooms'])

        prediction = model.predict([[area, bedrooms]])[0]

        return render_template(
            'index.html',
            prediction_text=f"Estimated Price: {round(prediction, 2)}"
        )
    except:
        return render_template(
            'index.html',
            prediction_text="Error: Please enter valid values"
        )

# Run app
if __name__ == "__main__":
    app.run(debug=True)
```

□ 2. HTML User Interface (`templates/index.html`)

```
<!DOCTYPE html>
<html>
<head>
  <title>House Price Predictor</title>
</head>
<body>

  <h2>House Price Prediction</h2>

  <form action="/predict" method="post">
    <label>Area (sq ft):</label><br>
    <input type="number" name="area" required><br><br>

    <label>Bedrooms:</label><br>
    <input type="number" name="bedrooms" required><br><br>

    <input type="submit" value="Predict">
  </form>

  <h3>{{ prediction_text }}</h3>

</body>
</html>
```

□ 3. Install Requirements

```
pip install flask joblib scikit-learn pandas
```

▶ □ 4. Run the App

In terminal:

```
python app.py
```

```
D:\ai\flask>
D:\ai\flask>python app.py
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a pr
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 746-919-328
```

Then open browser:

<http://127.0.0.1:5000/>



House Price Prediction

Area (sq ft):

Bedrooms:

Estimated Price: 496268.66

