House Price Prediction Model Deployment

========================================

This project demonstrates deploying a machine learning model (linear regression) to predict house prices using two methods: Gradio GUI and Flask API.

---

Prerequisites

-------------

1. Python 3.7+

2. Install required packages:

pip install gradio flask numpy scikit-learn pandas

---

Deployment Methods

------------------

1. Gradio GUI (User-Friendly Interface)

----------------------------------------

Steps:

1. Run the Gradio app:

python gradio\_app.py

2. A public link will be generated (e.g., "Running on public URL: https://xxx.gradio.app"). Open it in a browser.

3. Input:

- Select "Number of Rooms" (1-23).

- Enter "House Area" in square meters.

4. Click "Submit" to see the predicted price.

---

2. Flask API (Web Integration)

------------------------------

Steps:

1. Rename "house\_model.pkl" to "model.pkl".

2. Run the Flask app:

python app.py

3. Access the web interface at http://localhost:5000.

4. Input:

- Select "Number of Rooms" (1-5 in default UI; edit index.html to extend range).

- Enter "House Area" in square meters.

5. Click "Predict" to view the result.

---

Usage Instructions

------------------

- Input Features:

\* Number of Rooms: Integer between 1-23 (matches training data).

\* House Area (m²): Float value (e.g., 100, 200.5).

- Output:

\* Predicted price in Rwandan Francs (Rwf), formatted as "Predicted Price: X,XXX.XX Rwf".

---

Notes

-----

1. The model (house\_model.pkl) was trained on data with rooms ranging 1-23.

2. For production, disable Flask debug mode (set debug=False in app.py).

---

Repository Structure

--------------------

house\_dataset.csv # Training data

house\_model.pkl # Trained model

gradio\_app.py # Gradio deployment

app.py # Flask API

index.html # Flask frontend

requirements.txt # Dependencies (optional; create with "pip freeze > requirements.txt")

---

Deploy and predict seamlessly!