

**Name:** Hamza Al Hajj Chehade

**Batch Code:** LISUM02

**Submission Date:** 15-08-2021

**Submitted to:** Data Glacier

## DEPLOYMENT ON FLASK:

### ➤ Step1:

**Develop the ML model:** The competition goal is to predict the price of a specific house when providing the **average area income**, **average area house age**, **average number of rooms**, **average number of bedrooms**, **average number of populations**. These are the important features of the USA\_Housing dataset in which by the help of them we can predict the price of the house. However, the **Address** feature has no importance in predicting the price so it should be neglected.

```
1  #Importing the Libraries:
2  import numpy as np
3  import pandas as pd
4  import matplotlib as plt
5
6  #Importing the Dataset:
7  usa_housing=pd.read_csv('C:/Users/mizoh/Desktop/Data Glacier/USA_Housing/USA_Housing.csv')
8  X=usa_housing.iloc[:, :-2]
9  y=usa_housing.iloc[:, 5]
10
11 #Splitting the dataset into training set and test set:
12 from sklearn.model_selection import train_test_split
13 X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.3)
14
15 #Linear Regression modeling:
16 from sklearn.linear_model import LinearRegression
17 regressor=LinearRegression()
18 regressor.fit(X,y)
```

### ➤ Step2:

Save training model to a file using pickle library.

```
#Pickling:
import pickle
pickle.dump(regressor,open('USA_Housing.pkl','wb'))
```

## ➤ Step3:

```
1 import pickle
2 import numpy as np
3 from flask import Flask,render_template,request
4
5 app=Flask(__name__)
6 model=pickle.load(open('USA_Housing.pkl','rb'))
7 @app.route('/')
8 def home():
9     return render_template('index.html')
10
11 @app.route('/predict/', methods=["POST"])
12 def predict():
13     int_features=[float(x) for x in request.form.values()]
14     final_features=[np.array(int_features)]
15     prediction=model.predict(final_features)
16
17     output=round(prediction[0],4)
18
19     return render_template('index.html',prediction_text='The Price of The House is $ {}'.format(output))
20
21 if __name__ == '__main__':
22     app.run(debug=True,port=5000)
```

- Creating the instance of the *Flask()* and load the model.
- Bounded “/” with the *predict()* in which predict method gets the data from the loaded model passed by the requester.
- *model.predict()* method takes input and convert them into numpy array and the results are stored into the variable output.
- Returning this object using *render\_template*.

## ➤ Step4:

Open the WEB APP using the CMD Prompt by typing python app.py

```
Anaconda Prompt (anaconda3) - python app.py

(base) C:\Users\mizoh>cd Desktop

(base) C:\Users\mizoh\Desktop>cd Data Glacier

(base) C:\Users\mizoh\Desktop\Data Glacier>cd USA_Housing

(base) C:\Users\mizoh\Desktop\Data Glacier\USA_Housing>python app.py
* Serving Flask app "app" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Restarting with windowsapi reloader
* Debugger is active!
* Debugger PIN: 329-164-691
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

## ➤ **Step5:**

Creating the Web App by typing the URL in the browser

### Prediction Of The Price

The Price of The House is \$ 13814198.5806

## ➤ **Step6:**

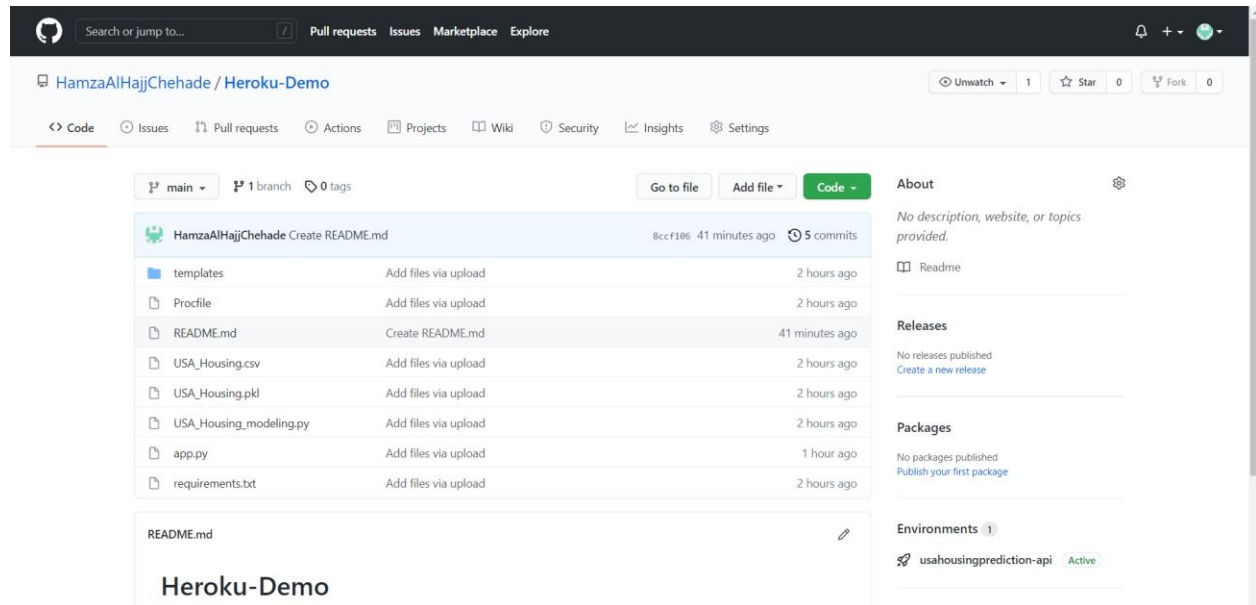
Create **Procfile** file which specifies the commands that are executed by a Heroku app on startup. **web: gunicorn app:app.**

Create the **requirements.txt** which specifies the libraries used in the code and its version

## ➤ **Step7:**

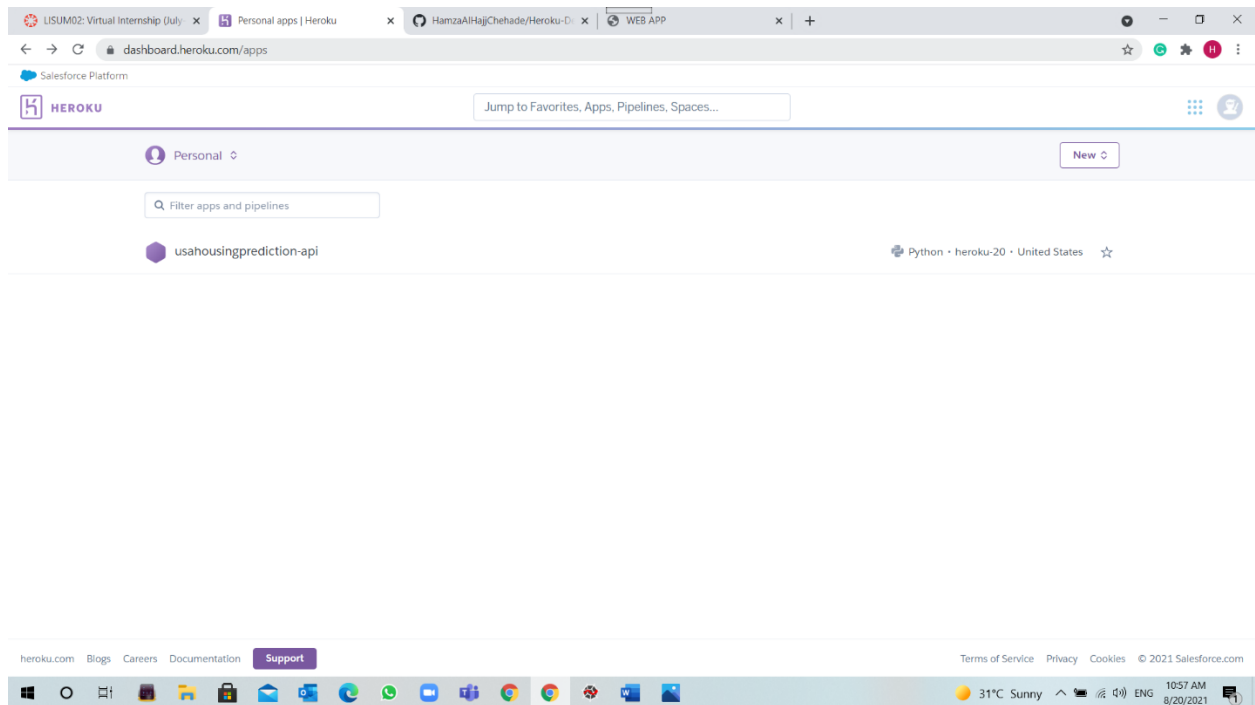
**Create a repository in GitHub and Commit the code.**

**Link: <https://github.com/HamzaAlHajjChehade/Heroku-Demo>**

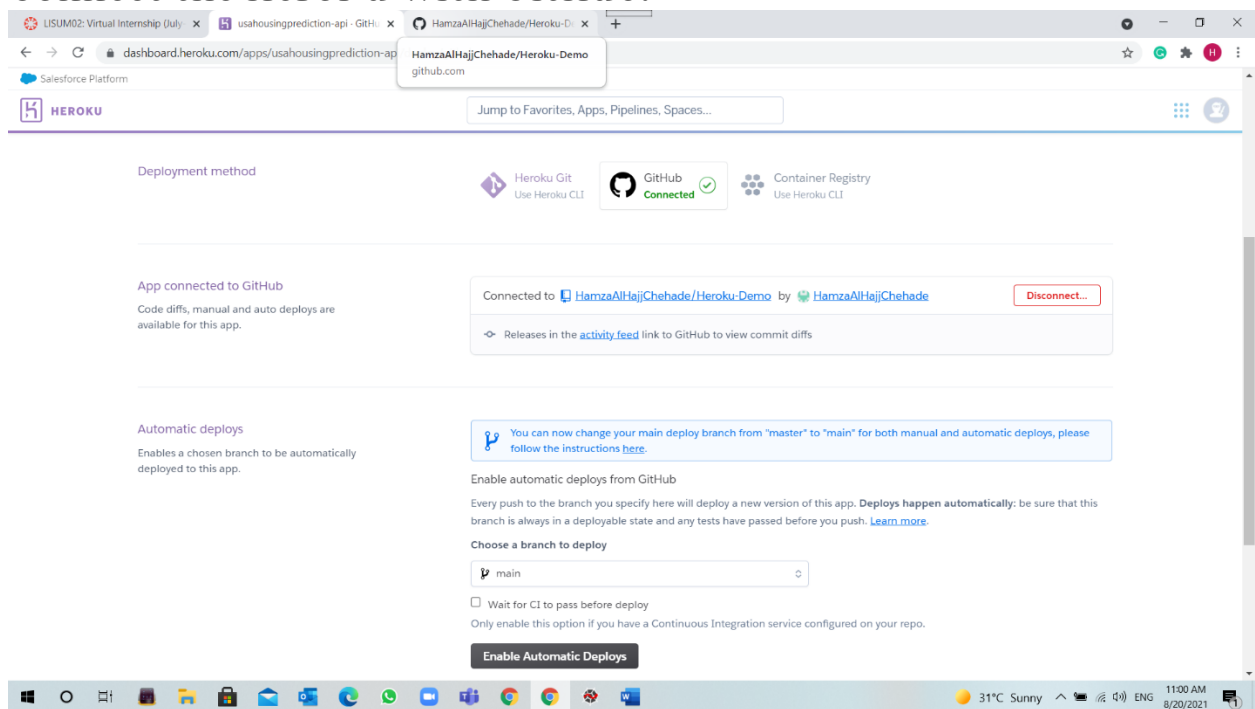


## ➤ **Step8:**

**Create an account in Heroku and the create an app.**



## ➤ Step9: Connect the Heroku with GitHub.



## ➤ Step10:

# Deploy the model in Heroku.

The screenshot shows the Heroku dashboard for the application 'usahousingprediction-api'. The deployment status is 'Deploy to Heroku' with a green checkmark. The deployment log shows the following steps:

- Manual deploy: Deploy the current state of a branch to this app.
- Deploy a GitHub branch: This will deploy the current state of the branch you specify below. [Learn more.](#)
- Choose a branch to deploy: main
- Receive code from GitHub: [checkmark]
- Build main 8ccf106c: [checkmark]
- Release phase: [checkmark]
- Deploy to Heroku: [checkmark]

Your app was successfully deployed. [View](#)

## ➤ Step11:

**App successfully deployed**

<https://usahousingprediction-api.herokuapp.com/>