Name: Ameya Damle

Batch Code: LISUM23: 30

Submission Date: 28-July-2023

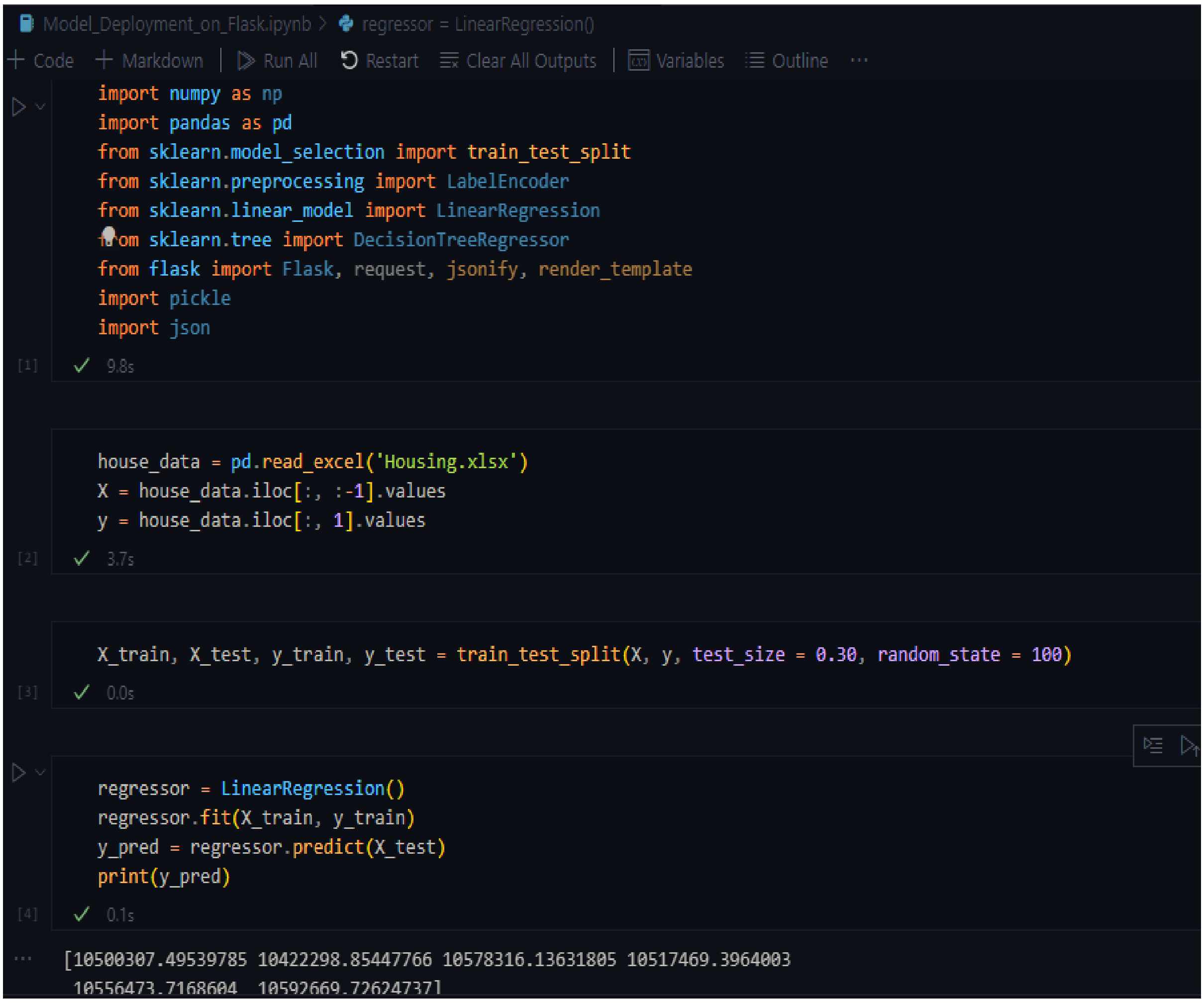
Submitted to: Data Glacier

Deployment of Flask Web App

Step 1:

Developing a model:

Predict the price of a house based on feature 'area' using Linear Regression Model.



```python
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.linear_model import LinearRegression
from sklearn.tree import DecisionTreeRegressor
from flask import Flask, request, jsonify, render_template
import pickle
import json
```

```python
house_data = pd.read_excel('Housing.xlsx')
X = house_data.iloc[:, :-1].values
y = house_data.iloc[:, 1].values
```

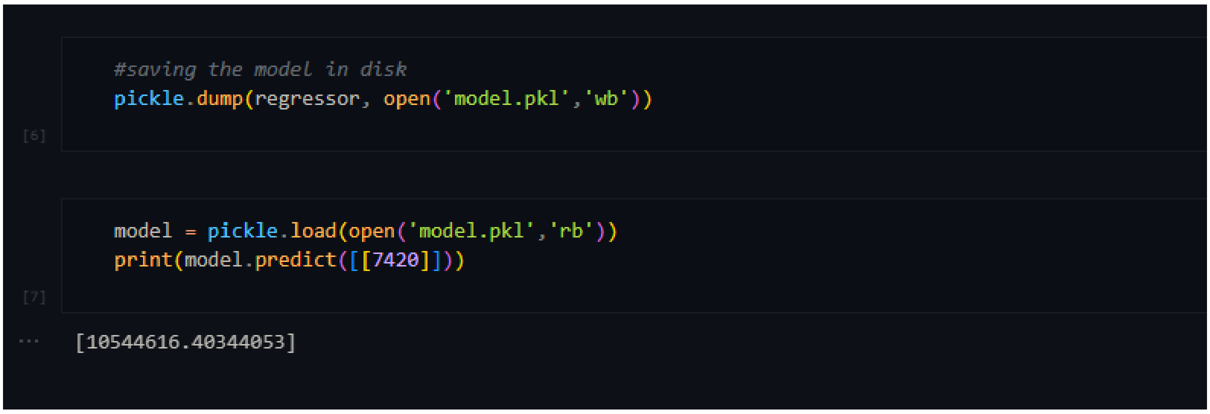
```python
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.30, random_state = 100)
```

```python
regressor = LinearRegression()
regressor.fit(X_train, y_train)
y_pred = regressor.predict(X_test)
print(y_pred)
```

```
[10500307.49539785 10422298.85447766 10578316.13631805 10517469.3964003
 10556473.7168604  10592669.72624737]
```

Step 2:

Saving the trained model as pickle file in the root folder.



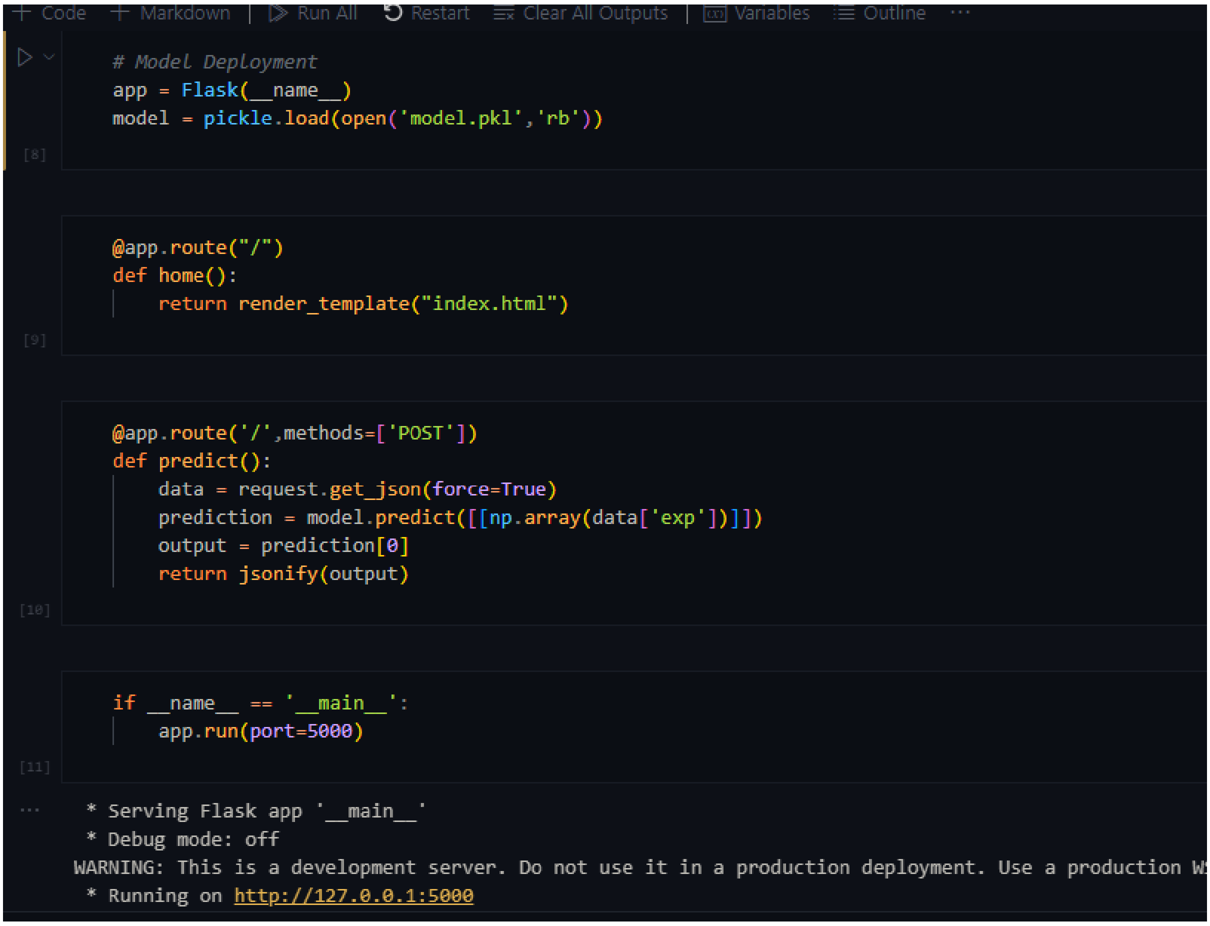
```python
#saving the model in disk
pickle.dump(regressor, open('model.pkl','wb'))
```

```python
model = pickle.load(open('model.pkl','rb'))
print(model.predict([[7420]]))
```

```
[10544616.40344053]
```

Step 3:

Deployment of model using flask.



```python
# Model Deployment
app = Flask(__name__)
model = pickle.load(open('model.pkl','rb'))
```

```python
@app.route("/")
def home():
    return render_template("index.html")
```

```python
@app.route('/',methods=['POST'])
def predict():
    data = request.get_json(force=True)
    prediction = model.predict([[np.array(data['exp'])]])
    output = prediction[0]
    return jsonify(output)
```

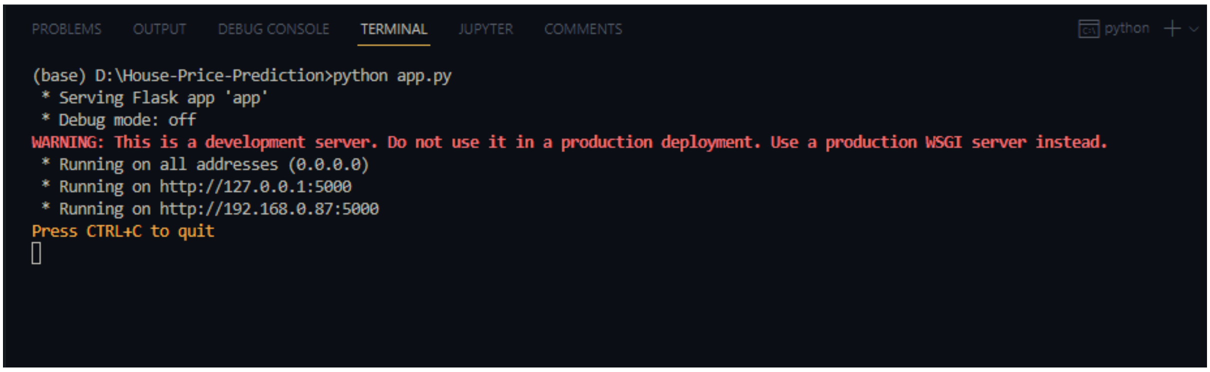
```python
if __name__ == '__main__':
    app.run(port=5000)
```

```
 * Serving Flask app '__main__'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production W
 * Running on http://127.0.0.1:5000
```

The script creates a Flask app and loads a machine learning model from a pickled file (model.pkl). The app has two routes defined: one for the home page ("/") that renders the "index.html" template, and another for handling POST requests sent to the root URL ("/") for making predictions. When the user submits the input data using the form on the home page, a POST request is sent to the server, and the predict() function is executed to make predictions using the model. The result of the prediction is then returned as a JSON response to the client.

Step 4:

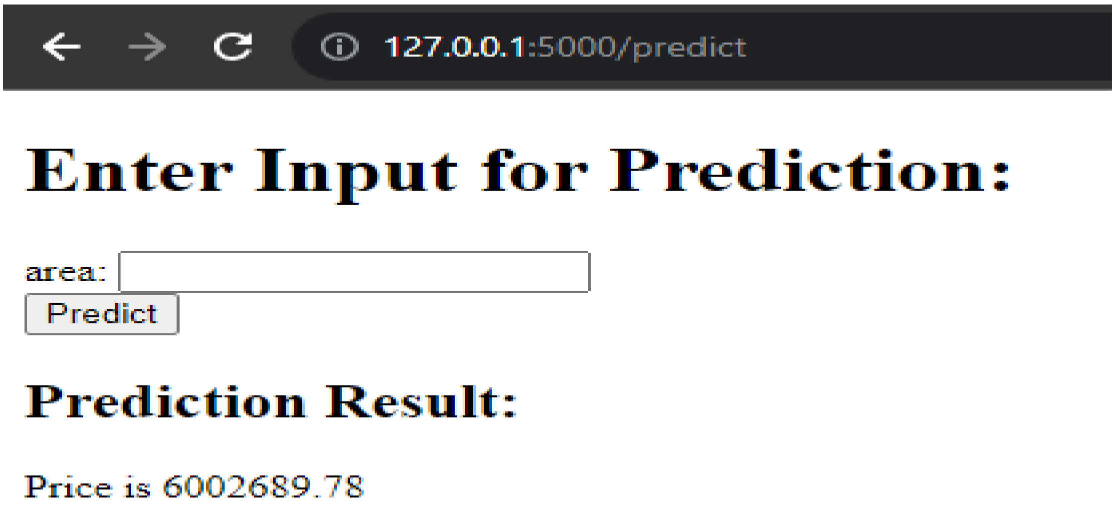
Checking the app.py file in cmd.



```
(base) D:\House-Price-Prediction>python app.py
 * Serving Flask app 'app'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http://192.168.0.87:5000
Press CTRL+C to quit
```

Step 5:

Launching the webapp by typing address “127.0.0.1:5000” in any browser.



Enter Input for Prediction:

area:

Prediction Result:

Price is 6002689.78