Name: Laith Adi | Submission Date: Saturday, March 20, 2021 | Batch Code: LISP01

Data Glacier
Week 4 – Deployment on Flask

Objective

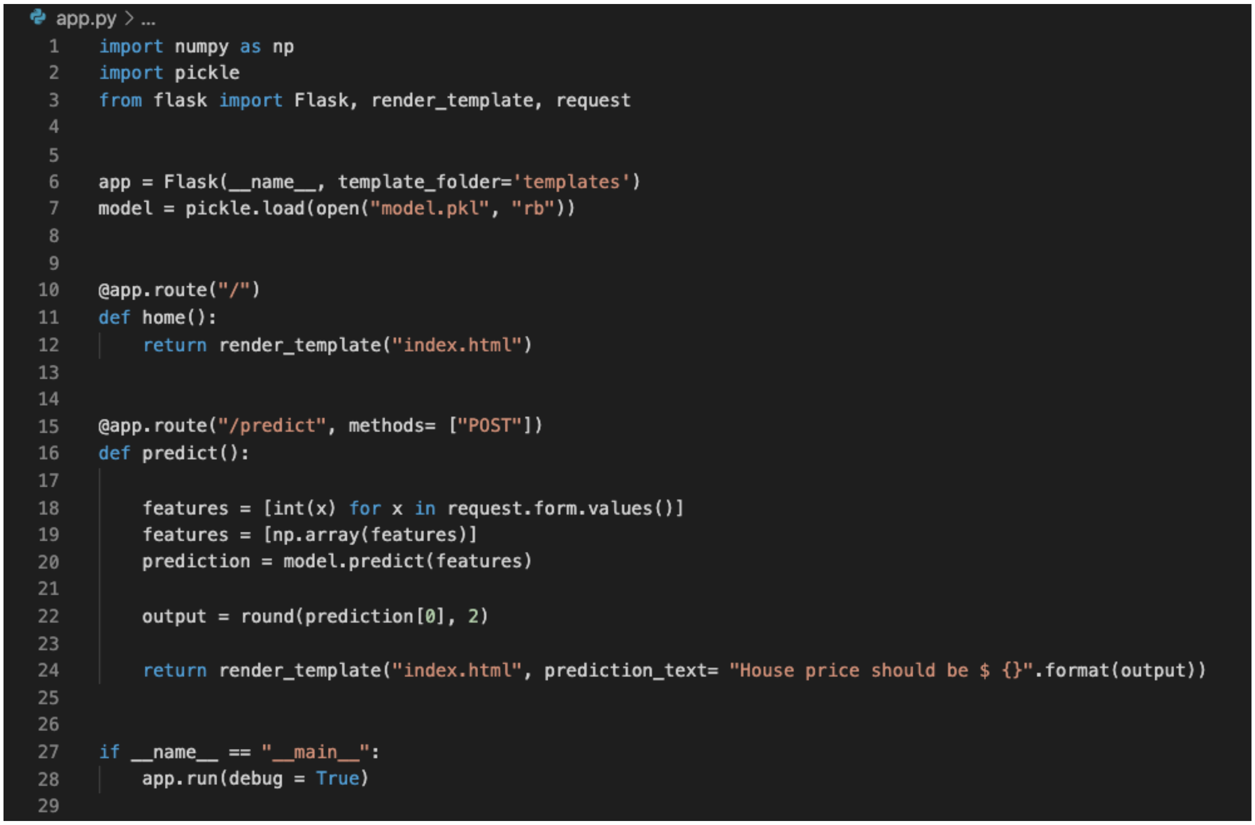
We are going to deploy a simple linear regression ML model on Flask. Using a dataset from Kaggle on house prices, we will train the model to best predict the price of a given house. The webpage will ask for the users' input, in which afterwards the user will click "Predict Price" button that will predict the price.

Data

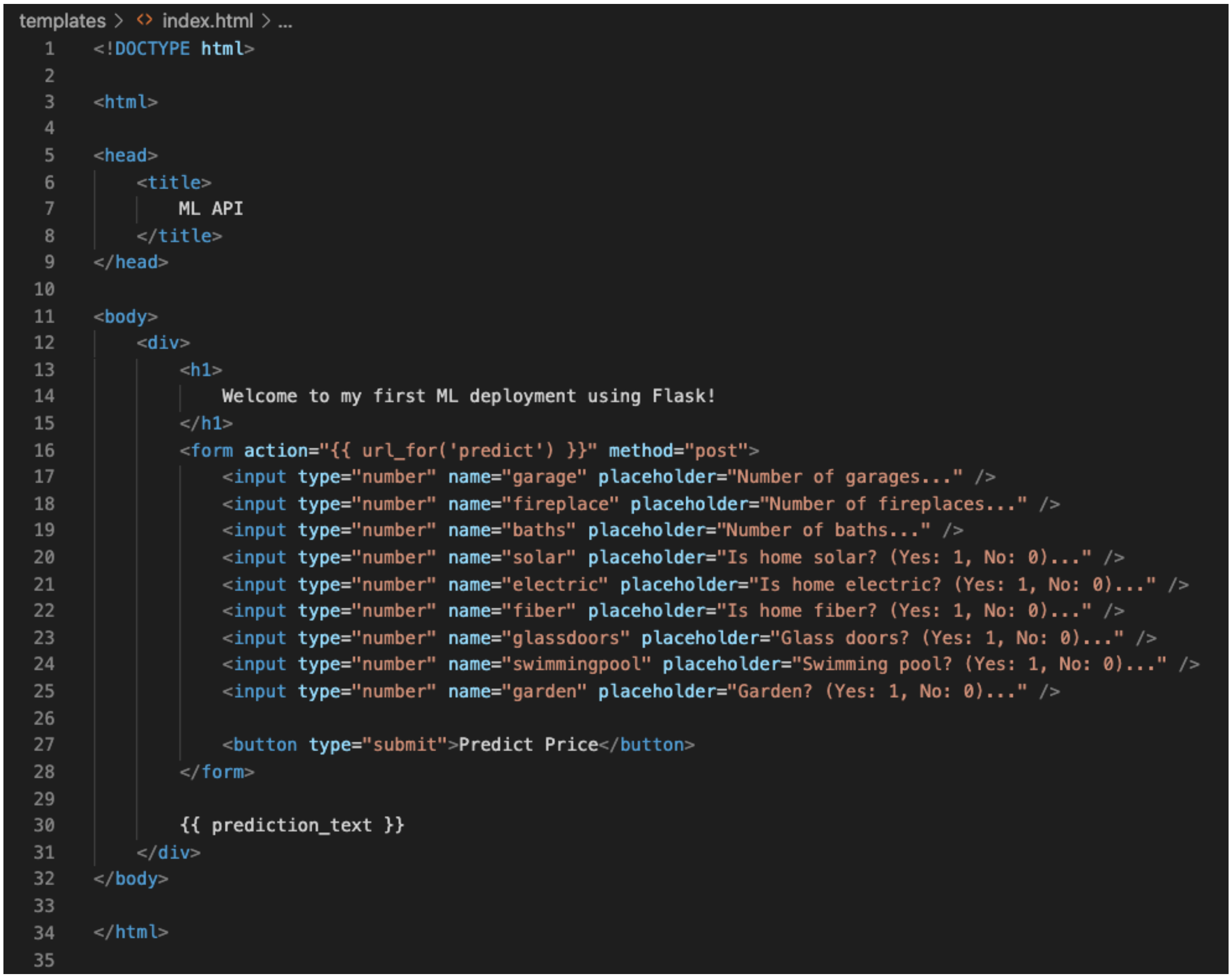
Used a house prices data from Kaggle. Saved the data in the same directory as the app.py file. Did not use all the features in the dataset. You can see exactly which features I used from the dataset to train the model. Same thing goes for the target variable.

Code (in order)

First the app.py and index.html files that store the APIs (using Flask) and the websites template, respectively.

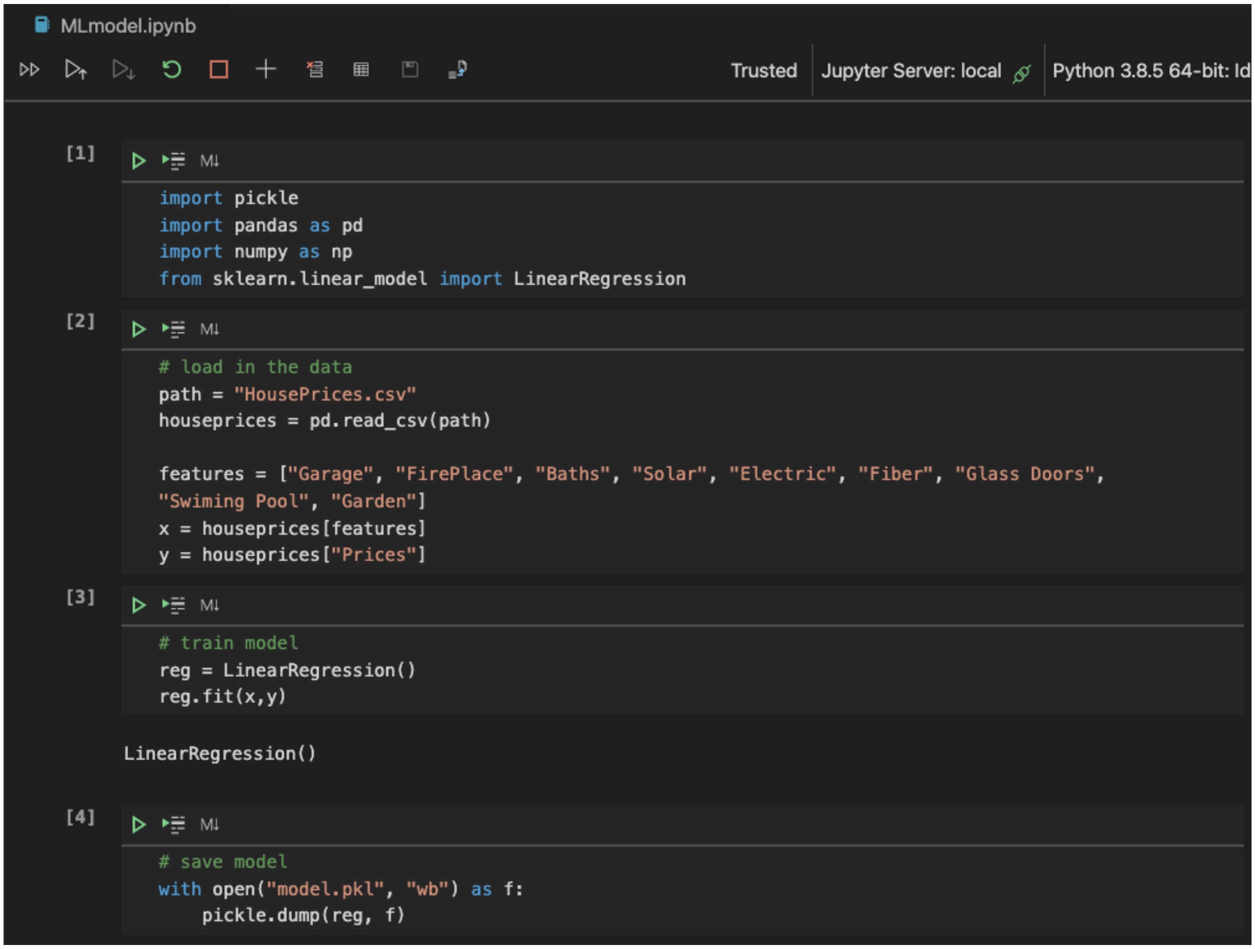


```python
import numpy as np
import pickle
from flask import Flask, render_template, request


app = Flask(__name__, template_folder='templates')
model = pickle.load(open("model.pkl", "rb"))


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


@app.route("/predict", methods= ["POST"])
def predict():

    features = [int(x) for x in request.form.values()]
    features = [np.array(features)]
    prediction = model.predict(features)

    output = round(prediction[0], 2)

    return render_template("index.html", prediction_text= "House price should be $ {}".format(output))


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

```



```html
<!DOCTYPE html>

<html>

<head>
    <title>
        ML API
    </title>
</head>

<body>
    <div>
        <h1>
            Welcome to my first ML deployment using Flask!
        </h1>
        <form action="{{ url_for('predict') }}" method="post">
            <input type="number" name="garage" placeholder="Number of garages..." />
            <input type="number" name="fireplace" placeholder="Number of fireplaces..." />
            <input type="number" name="baths" placeholder="Number of baths..." />
            <input type="number" name="solar" placeholder="Is home solar? (Yes: 1, No: 0)..." />
            <input type="number" name="electric" placeholder="Is home electric? (Yes: 1, No: 0)..." />
            <input type="number" name="fiber" placeholder="Is home fiber? (Yes: 1, No: 0)..." />
            <input type="number" name="glassdoors" placeholder="Glass doors? (Yes: 1, No: 0)..." />
            <input type="number" name="swimmingpool" placeholder="Swimming pool? (Yes: 1, No: 0)..." />
            <input type="number" name="garden" placeholder="Garden? (Yes: 1, No: 0)..." />

            <button type="submit">Predict Price</button>
        </form>

        {{ prediction_text }}
    </div>
</body>

</html>

```

Second, using Pandas, Pickle, and Sklearn, we are going to load our data as a Pandas Dataframe. Then build and train our linear regression model. Then save the model using Pickle. This all will happen in a Jupyter Notebook, MLmodel.ipynb.



```
import pickle
import pandas as pd
import numpy as np
from sklearn.linear_model import LinearRegression
```

```
# load in the data
path = "HousePrices.csv"
houseprices = pd.read_csv(path)

features = ["Garage", "FirePlace", "Baths", "Solar", "Electric", "Fiber", "Glass Doors",
"Swiming Pool", "Garden"]
x = houseprices[features]
y = houseprices["Prices"]
```

```
# train model
reg = LinearRegression()
reg.fit(x,y)
```

```
LinearRegression()
```

```
# save model
with open("model.pkl", "wb") as f:
    pickle.dump(reg, f)
```

Testing

Run the app.py file. It will run locally, so you can test the code there.