Flask Deployment

Name: Mahnoor Farhat
Date: 28th January, 2025
Internship Batch: LISUM41: 30 December, 2024 – 31 March, 2025
URL: https://github.com/mahnoor-farhat/flask-deployment

Deployment Steps:

index.html

templates > index.html > ...

```html
<!DOCTYPE html>
<html>
<head>
    <title>Iris Prediction</title>
</head>
<body>
    <h1>Predict Iris Species</h1>
    <form action="{{url_for('/predict')}}" method="POST">
        <label>Sepal Length:</label>
        <input type="text" name="sepal_length"><br>
        <label>Sepal Width:</label>
        <input type="text" name="sepal_width"><br>
        <label>Petal Length:</label>
        <input type="text" name="petal_length"><br>
        <label>Petal Width:</label>
        <input type="text" name="petal_width"><br>
        <button type="submit">Predict</button>
    </form>
    {% if result %}
        <h2>Prediction: {{ result }}</h2>
    {% endif %}
</body>
</html>

```

iris.py

```python
import pickle
from sklearn.datasets import load_iris
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split

data = load_iris()
X, y = data.data, data.target
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

model = RandomForestClassifier()
model.fit(X_train, y_train)

with open("iris_model.pkl", "wb") as file:
    pickle.dump(model, file)

print("Model saved as iris_model.pkl")
```

app.py

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

app = Flask(__name__)
model = pickle.load(open('iris_model.pkl', 'rb'))

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

@app.route('/predict', methods=['POST'])
def predict():
    try:
        # Log form data
        print("Form Data Received:", request.form)

        sepal_length = float(request.form['sepal_length'])
        sepal_width = float(request.form['sepal_width'])
        petal_length = float(request.form['petal_length'])
        petal_width = float(request.form['petal_width'])

        input_data = np.array([[sepal_length, sepal_width, petal_length, petal_width]])
        print("Input Data for Prediction:", input_data)
```

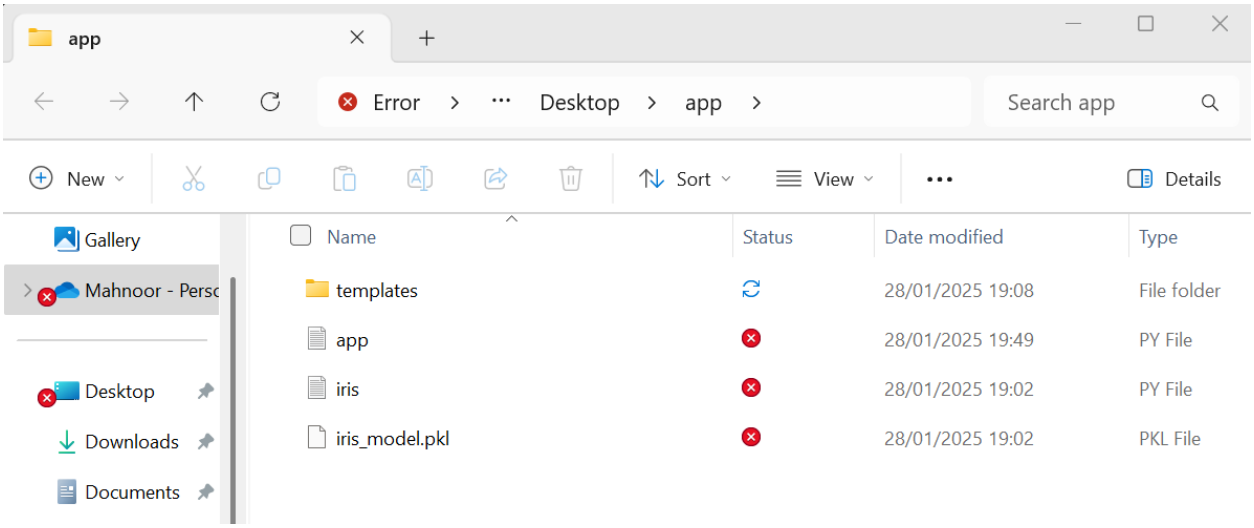
app.py

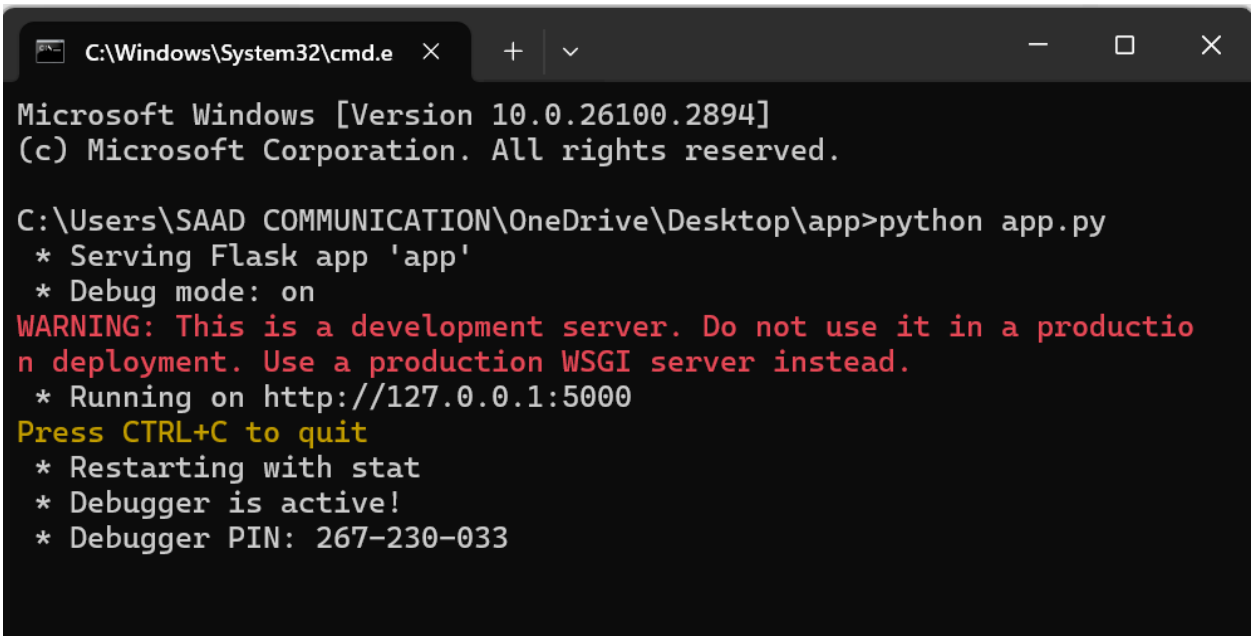
app.py > home

```python
def predict():
        prediction = model.predict(input_data)
        species = ["Setosa", "Versicolor", "Virginica"][prediction[0]]
        print("Prediction Result:", species)

        return render_template("index.html", result=species)
    except Exception as e:
        print("Error in Prediction:", e)
        return "Error: " + str(e)


if __name__ == '__main__':
    app.run(port=5000, debug=True)
```




C:\Windows\System32\cmd.e
Microsoft Windows [Version 10.0.26100.2894]
(c) Microsoft Corporation. All rights reserved.
C:\Users\SAAD COMMUNICATION\OneDrive\Desktop\app>python app.py
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a productio
n deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 267-230-033