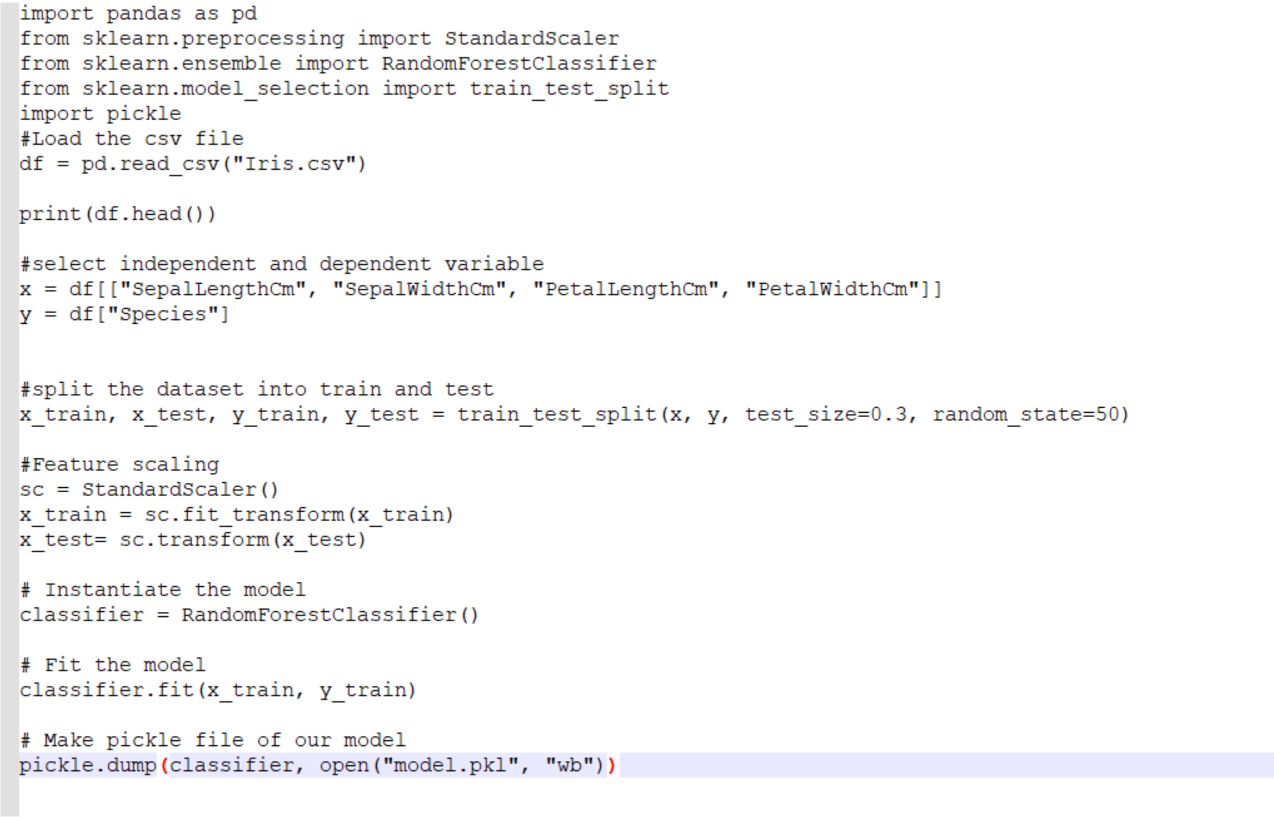
Name: Deborah Adeyemi

Submission Date: 10/08/2022

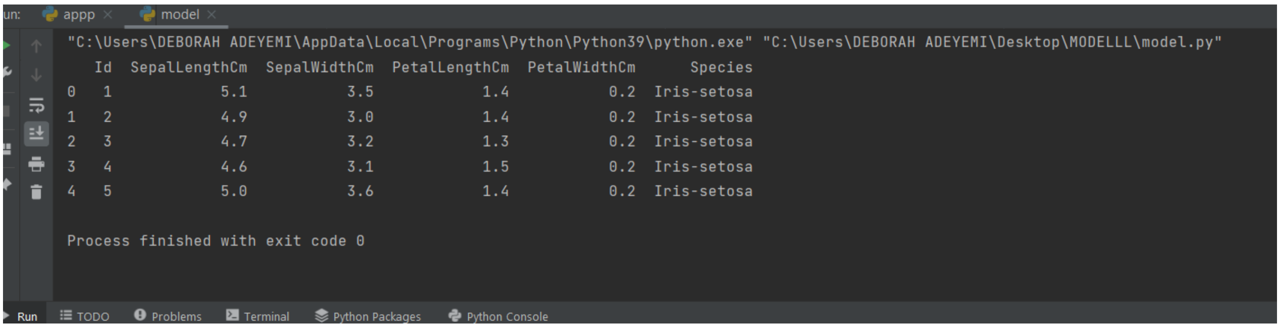
Submitted to GitHub:https://github.com/queendebra92/Week-4

1. Iris model was created and saved. A folder was created on pycharm called model.py



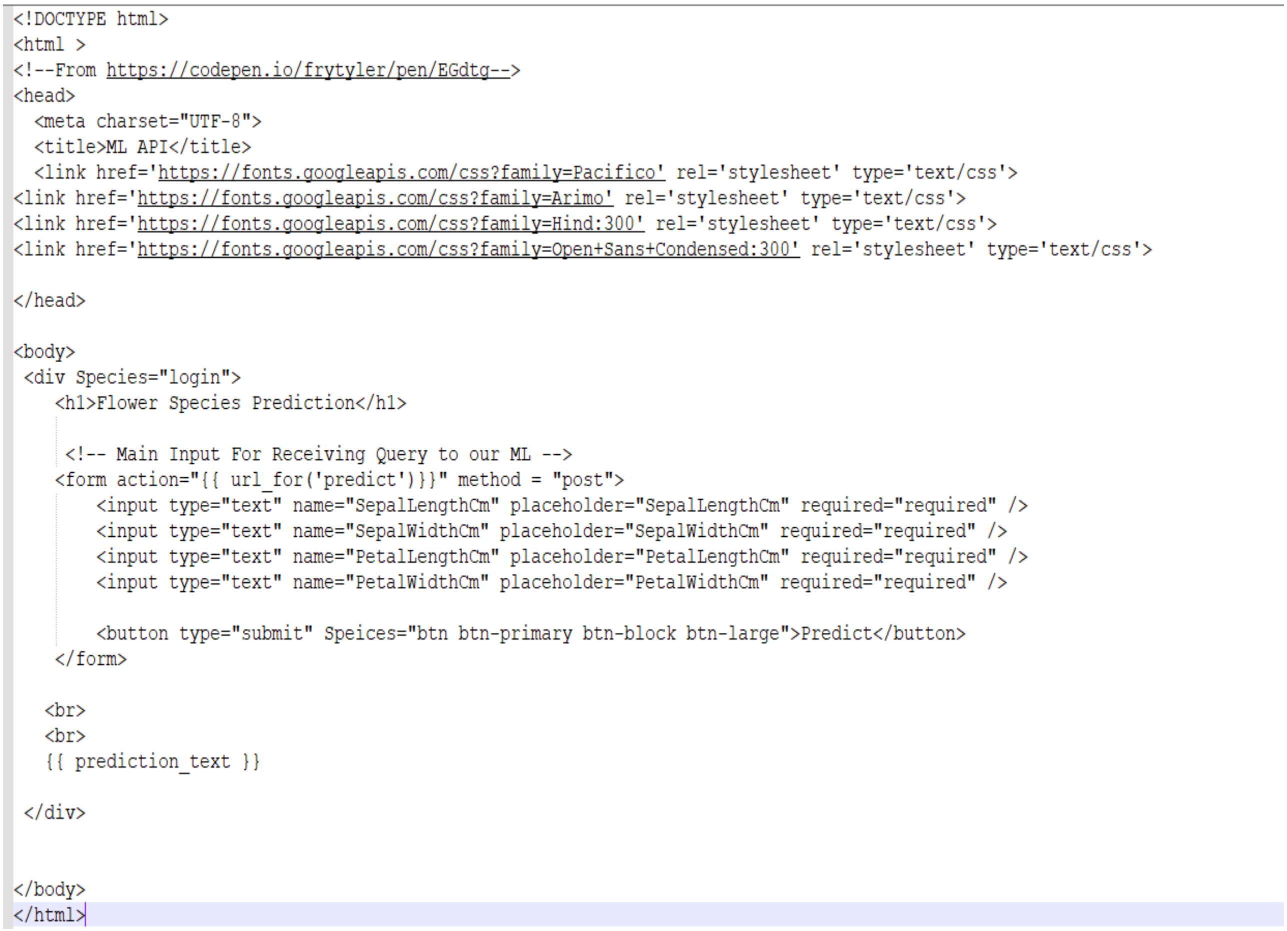
```
import pandas as pd
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
import pickle
#Load the csv file
df = pd.read_csv("Iris.csv")

print(df.head())

#select independent and dependent variable
x = df[["SepalLengthCm", "SepalWidthCm", "PetalLengthCm", "PetalWidthCm"]]
y = df["Species"]


#split the dataset into train and test
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3, random_state=50)

#Feature scaling
sc = StandardScaler()
x_train = sc.fit_transform(x_train)
x_test= sc.transform(x_test)

# Instantiate the model
classifier = RandomForestClassifier()

# Fit the model
classifier.fit(x_train, y_train)

# Make pickle file of our model
pickle.dump(classifier, open("model.pkl", "wb"))
```

Result



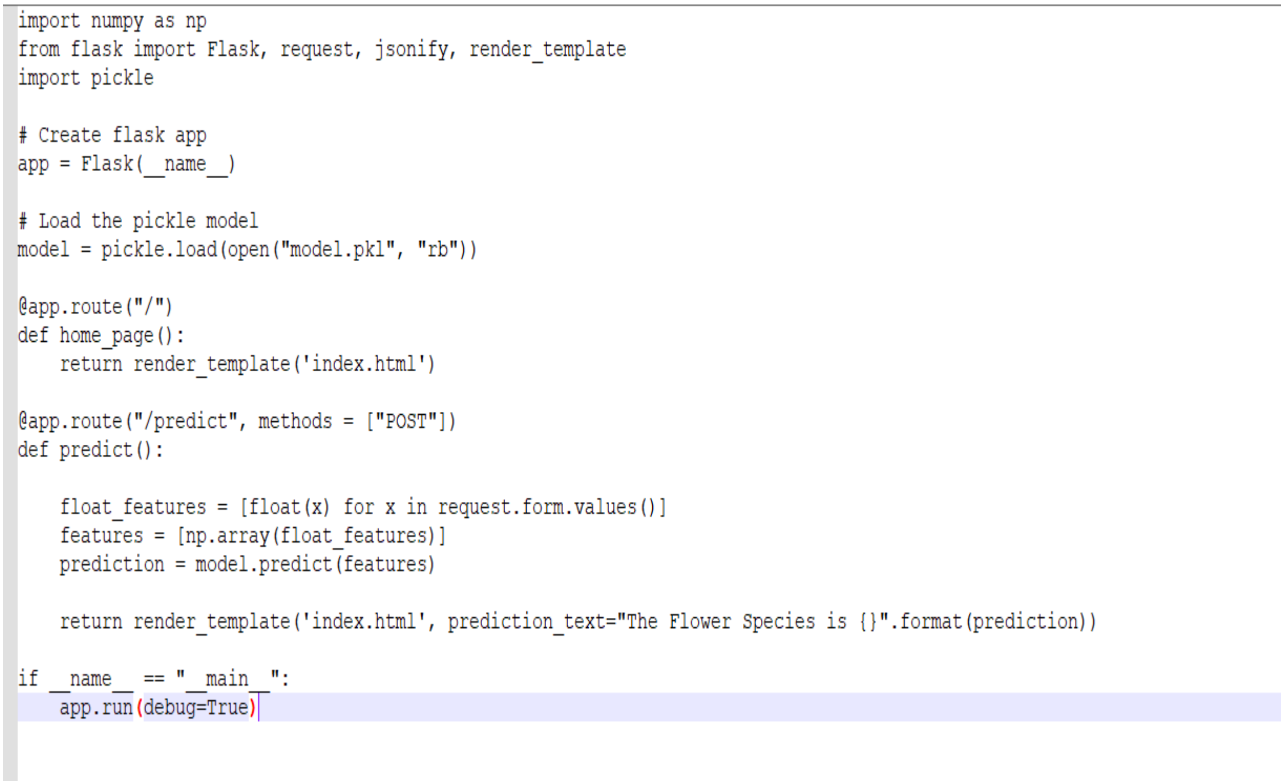
```
"C:\Users\DEBORAH ADEYEMI\AppData\Local\Programs\Python\Python39\python.exe" "C:\Users\DEBORAH ADEYEMI\Desktop\MODELLL\model.py"
   Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm      Species
0   1            5.1           3.5            1.4           0.2  Iris-setosa
1   2            4.9           3.0            1.4           0.2  Iris-setosa
2   3            4.7           3.2            1.3           0.2  Iris-setosa
3   4            4.6           3.1            1.5           0.2  Iris-setosa
4   5            5.0           3.6            1.4           0.2  Iris-setosa

Process finished with exit code 0
```

2. Created a HTML file



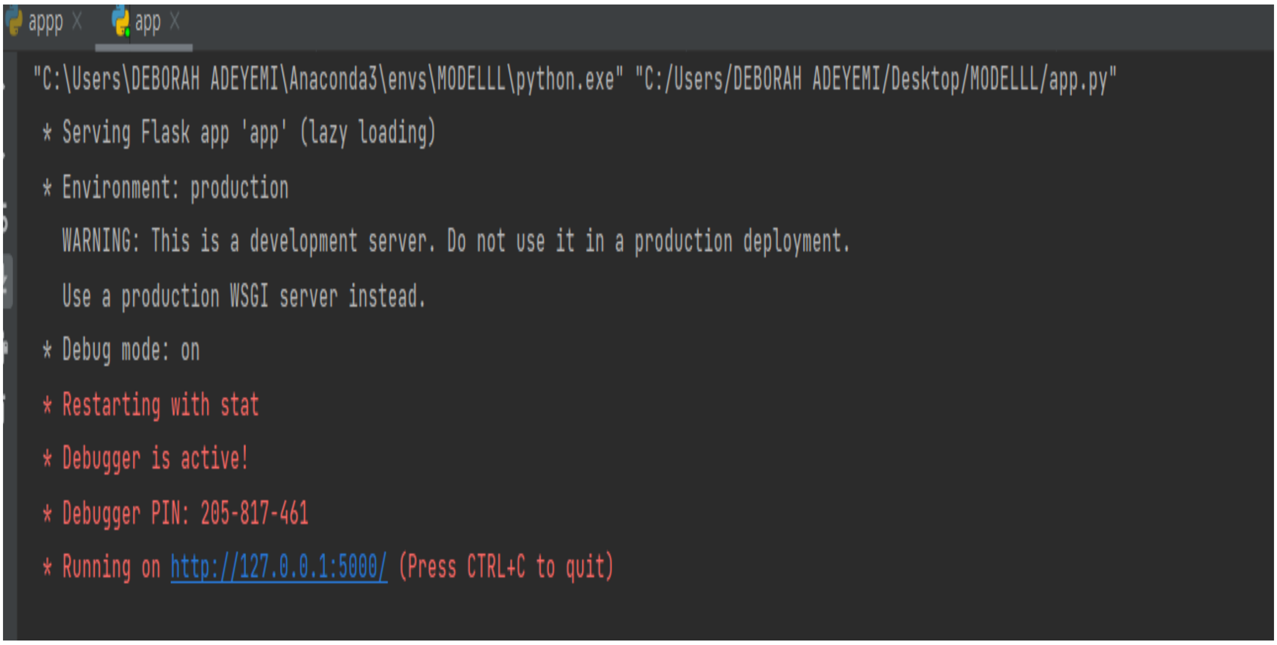
```
<!DOCTYPE html>
<html >
<!--From https://codepen.io/frytyler/pen/EGdtg-->
<head>
  <meta charset="UTF-8">
  <title>ML API</title>
  <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet' type='text/css'>

</head>

<body>
 <div Species="login">
    <h1>Flower Species Prediction</h1>

     <!-- Main Input For Receiving Query to our ML -->
    <form action="{{ url_for('predict')}}" method = "post">
        <input type="text" name="SepalLengthCm" placeholder="SepalLengthCm" required="required" />
        <input type="text" name="SepalWidthCm" placeholder="SepalWidthCm" required="required" />
        <input type="text" name="PetalLengthCm" placeholder="PetalLengthCm" required="required" />
        <input type="text" name="PetalWidthCm" placeholder="PetalWidthCm" required="required" />

        <button type="submit" Speices="btn btn-primary btn-block btn-large">Predict</button>
    </form>

   <br>
   <br>
   {{ prediction_text }}

 </div>


</body>
</html>
```

3. APP.PY file was created.



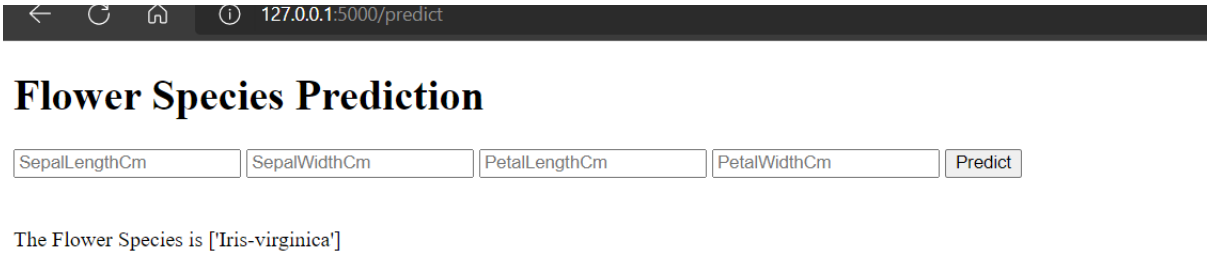
```
import numpy as np
from flask import Flask, request, jsonify, render_template
import pickle

# Create flask app
app = Flask(__name__)

# Load the pickle model
model = pickle.load(open("model.pkl", "rb"))

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

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

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

    return render_template('index.html', prediction_text="The Flower Species is {}".format(prediction))

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

Result



```
"C:\Users\DEBORAH ADEYEMI\Anaconda3\envs\MODELLL\python.exe" "C:/Users/DEBORAH ADEYEMI/Desktop/MODELLL/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 stat
 * Debugger is active!
 * Debugger PIN: 205-817-461
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

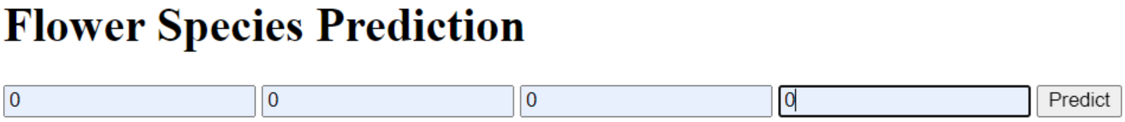
4. Redirected to the web page.



Flower Species Prediction

The Flower Species is ['Iris-virginica']

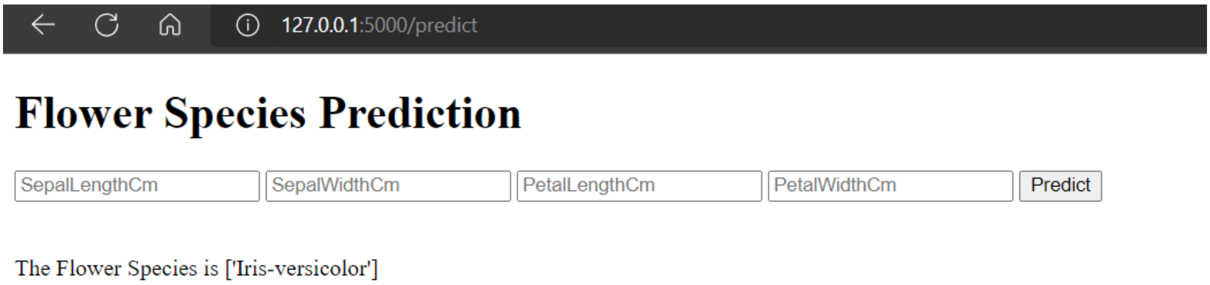
Predicted flower species



Flower Species Prediction

0 0 0 0

Result



Flower Species Prediction

The Flower Species is ['Iris-versicolor']

5. Submitted to github.com
   https://github.com/queendebra92/Week-4