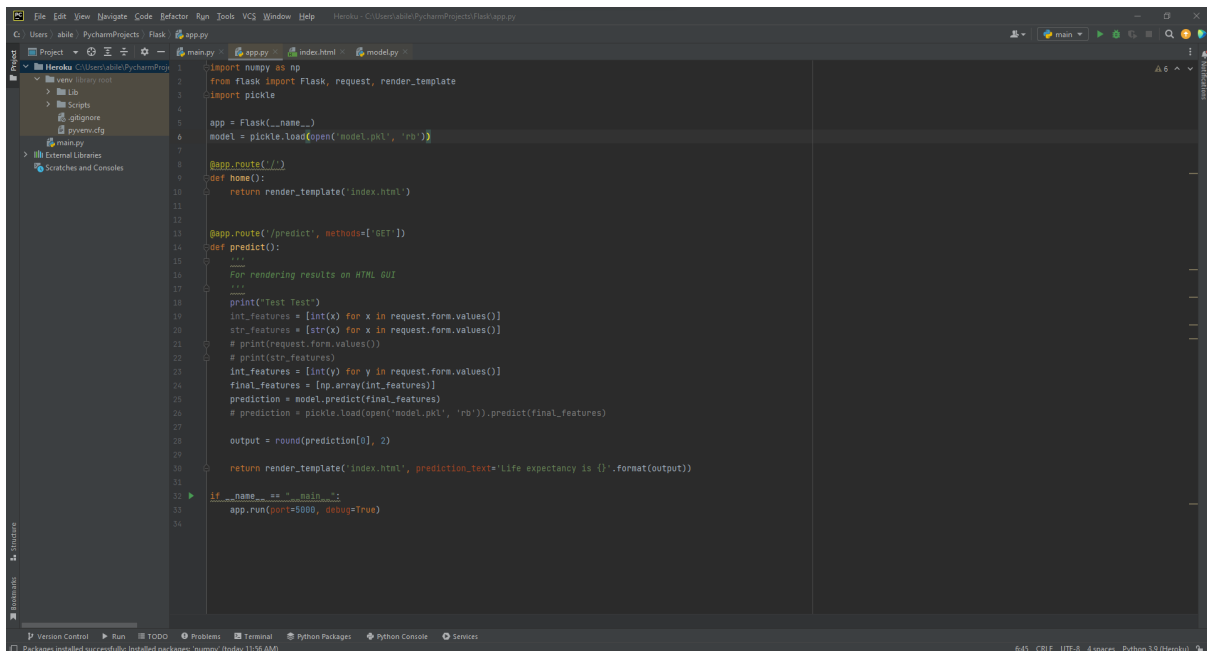
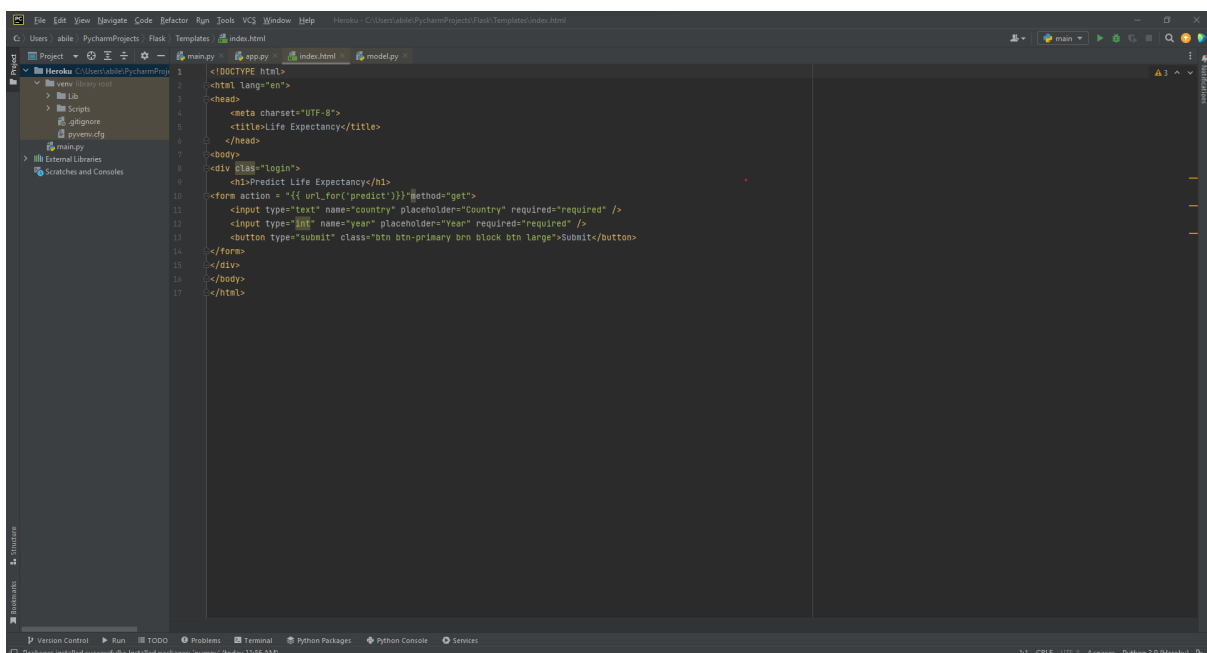


Name: Abilesh Ayyanar
Batch #: LISUM215
Submission Date: 5 December 2022



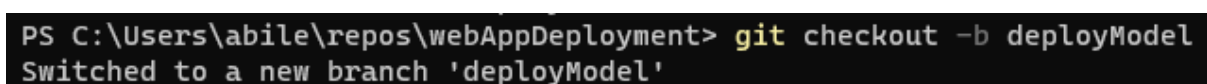
```
1 import numpy as np
2 from flask import Flask, request, render_template
3 import pickle
4
5 app = Flask(__name__)
6 model = pickle.load(open('model.pkl', 'rb'))
7
8 @app.route('/')
9 def home():
10     return render_template('index.html')
11
12
13 @app.route('/predict', methods=['GET'])
14 def predict():
15     """
16     For rendering results on HTML GUI
17     """
18     print("Test Test")
19     int_features = [int(x) for x in request.form.values()]
20     str_features = [str(x) for x in request.form.values()]
21     # print(request.form.values())
22     # print(str_features)
23     int_features = [int(y) for y in request.form.values()]
24     final_features = [np.array(int_features)]
25     prediction = model.predict(final_features)
26     # prediction = pickle.load(open('model.pkl', 'rb')).predict(final_features)
27
28     output = round(prediction[0], 2)
29
30     return render_template('index.html', prediction_text='Life expectancy is {}'.format(output))
31
32 if __name__ == '__main__':
33     app.run(port=5000, debug=True)
```

This app.py file shows the input type as well as the output type.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>Life Expectancy</title>
6 </head>
7 <body>
8     <div class="login">
9         <h1>Predict Life Expectancy</h1>
10         <form action = "{{ url_for('predict')}}" method="get">
11             <input type="text" name="country" placeholder="Country" required="required" />
12             <input type="text" name="year" placeholder="Year" required="required" />
13             <button type="submit" class="btn btn-primary btn-block btn-large">Submit</button>
14         </form>
15     </div>
16 </body>
17 </html>
```

The index.txt file shows how the search bars and buttons were added to the app.



```
PS C:\Users\abile\repos\webAppDeployment> git checkout -b deployModel
Switched to a new branch 'deployModel'
```

New branch is created to deploy on Heroku

Connect to GitHub

Connect for automatic deploys, test runs and review apps on this pipeline

Search for a repository to connect to

abileshayyanar

webDeployment

Search

Missing a GitHub organization? [Ensure Heroku Dashboard has team access.](#)

abileshayyanar/webDeployment

Connect

The app was created on Heroku and connected directly to the GitHub repository

Automatic deploys

Enables a chosen branch to be automatically deployed to this app.

You can now change your main deploy branch from "master" to "main" for both manual and automatic deploys, please follow the instructions [here](#).

Enable automatic deploys from GitHub

Every push to the branch you specify here will deploy a new version of this app. **Deploys happen automatically;** be sure that this branch is always in a deployable state and any tests have passed before you push. [Learn more](#).

Choose a branch to deploy

deployModel

☐ Wait for CI to pass before deploy

Only enable this option if you have a Continuous Integration service configured on your repo.

Enable Automatic Deploys

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

deployModel

Deploy Branch

App was then deployed in the specific branch in the repository

Predict Life Expectancy

Country Year Submit

Heroku then built and deployed the application that looks like this