


Data Glacier Assignment

Iris Model Deployment on Heroku




1. I made a webpage using HTML that could be used to input the values for the Iris flower class prediction.








```
1 <!DOCTYPE html>
2
3 <html lang="en">
4
5 <head>
6   <title>Predict Iris Flower Class</title>
7
8   <link href="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css"
9 </head>
10
11 <body>
12   <br>
13   <h1>Iris Flower Class Prediction</h1>
14   <div>
15     <br>
16     <p>Enter the following measurements and hit 'Predict' to get the class of Iris
17
18     <form action="{ url_for('predict') }}" method="post">
19       <input type="text" name = "sepal_len" placeholder="Sepal Length" required=
20       <input type="text" name = "sepal_wid" placeholder="Sepal Width" required=
21       <input type="text" name = "petal_len" placeholder="Petal Length" required=
22       <input type="text" name = "petal_wid" placeholder="Petal Width" required=
23
24       <button class="btn btn-primary" type="submit">Predict</button>
25     <br><br>
26     {{ prediction_text }}
27   </div>
28 </body>
29
30
31
32
```

2. The files were pushed to a Github repository, including a Procfile and requirements.txt file which are required by Heroku. I also

 **MuadhF** / iris-heroku Unwatch


[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)


 main  1 branch  0 tags Go to file Add file Code


 MuadhF Add files via upload 685005b 1 hour ago 6 commits
 templates Add files via upload 3 hours ago
 Procfile Add files via upload 1 hour ago
 README.md Initial commit 3 hours ago
 model.py Add files via upload 3 hours ago
 requirements.txt Update requirements.txt 2 hours ago
 simple_model.pkl Add files via upload 3 hours ago

3. Then on Heroku, I connected to my Github account and then to the repository.

Deployment method


 Heroku Git
Use Heroku CLI


 GitHub
Connected

 Container Registry
Use Heroku CLI

App connected to GitHub


Code diffs, manual and auto deploys are available for this app.

Connected to [MuadhF/iris-heroku](#) by  **MuadhF** Disconnect...

 Releases in the [activity feed](#) link to GitHub to view commit diffs

4. Once Heroku connected to the repository, I deployed the model

Choose a branch to deploy

 main Deploy Branch

Receive code from GitHub ✓

Build **main** 685005b8 ...

```
-----> Building on the Heroku-20 stack
-----> Using buildpack: heroku/python
-----> Python app detected
```

☒ Autoscroll with output [View build log](#)


Release phase

Deploy to Heroku

Deploy a GitHub branch

This will deploy the current state of the branch you specify below. [Learn more](#).

Choose a branch to deploy


 main

Deploy Branch

Receive code from GitHub 

Build **main** 685005b8 

Release phase 

Deploy to Heroku 

Your app was successfully deployed.

 View

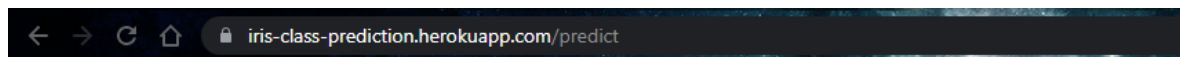
5. Then the link could be opened and the model was ready to use.



Iris Flower Class Prediction

Enter the following measurements and hit 'Predict' to get the class of Iris flower!

Predict



Iris Flower Class Prediction

Enter the following measurements and hit 'Predict' to get the class of Iris flower!

Sepal Length

Sepal Width

Petal Length

Petal Width

Predict

Iris flower class is 2