

Name: Devin Chau

Date: 2024 December 04

Batch Code: LISUM39

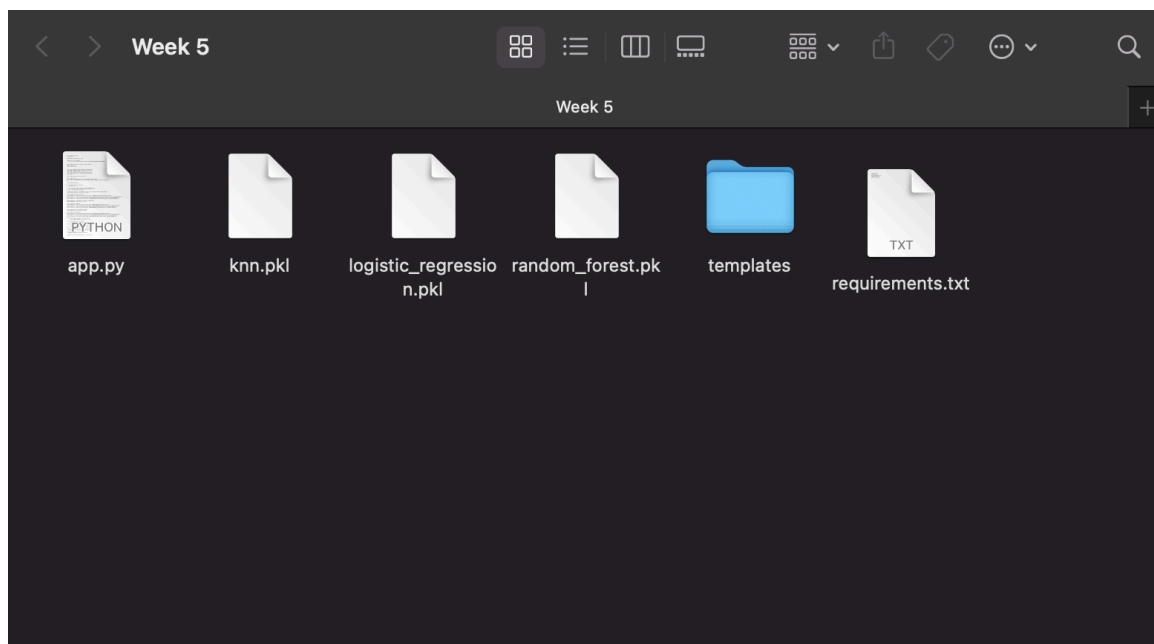
Data intake reviewer: Data Glacier

Submission link:

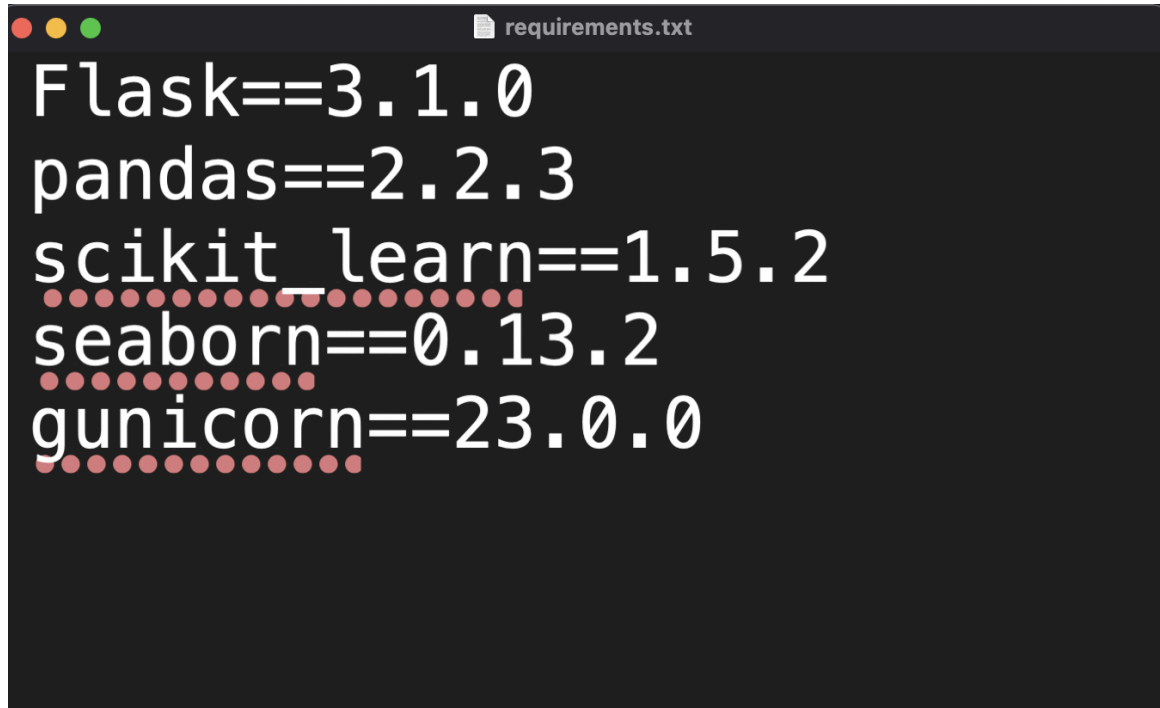
<https://github.com/mynameisdevinchau/Data-Glacier-Internship/tree/main/Week%205>

1.

- Using the model from Week 4, I put all the required values to run Week 4 into the Week 5 folder, with one change: I put a requirements.txt into the folder required for deploying this Web App.



- The requirements.txt lists the pip installation requirements required to run the model.
- These are the values in the requirements.txt that will be run in deployment



```
Flask==3.1.0
pandas==2.2.3
scikit_learn==1.5.2
seaborn==0.13.2
gunicorn==23.0.0
```

- I pip-installed everything on my end to make sure everything worked correctly
- Now that everything is in one folder, I uploaded my Week 5 file to my Data Glacier Internship repository and connected this to my deployment website.

2.

- I used Render to deploy my WebApp by connecting it to my GitHub.
 - <https://render.com/>
 - I chose Render because it was free but slow

Steps of my deployment

- Choosing Data-Glacier-Internship as my Git Provider because it is where I store my Week 5 folder

You are deploying a Web Service

Source Code

Git Provider Public Git Repository Existing Image

Search

mynameisdevinchau / Data-Glacier-Internship 1h ago

- Naming my web service for Web Deployment

Name

A unique name for your web service.

LISUM39-irisapp

- Made sure my language was set properly with the Branch done properly. The Root Directory should be my Week 5 folder. They pip install all the values from my requirements.txt which then uses gunicorn for deployment.

Language

Python 3

Branch

The Git branch to build and deploy.

main

Region

Your services in the same [region](#) can communicate over a [private network](#). You currently have services running in **Oregon**.

Oregon (US West) 1 existing service

Deploy in a new region +

Root Directory Optional

If set, Render runs commands from this directory instead of the repository root. Additionally, code changes outside of this directory do not trigger an auto-deploy. Most commonly used with a [monorepo](#).

Week 5

Build Command

Render runs this command to build your app before each deploy.

Week 5/ \$ pip install -r requirements.txt

Start Command

Render runs this command to start your app with each deploy.

Week 5/ \$ gunicorn app:app

- Picked the Free option to not spend any money but it is much slower when running

Instance Type

For hobby projects

Free \$0 / month	512 MB (RAM) 0.1 CPU	Upgrade to enable more features Free instances spin down after periods of inactivity. They do not support SSH access, scaling, one-off jobs, or persistent disks. Select any paid instance type to enable these features.
----------------------------	-------------------------	---

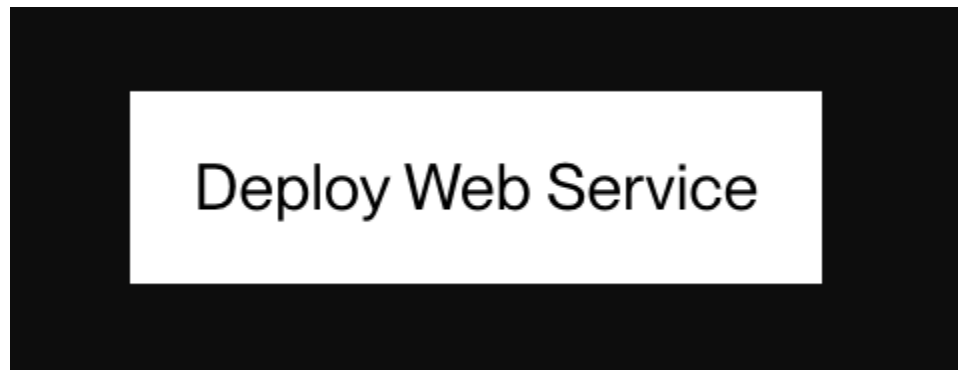
For professional use
For more power and to get the most out of Render, we recommend using one of our paid instance types. All paid instances support:

- Zero Downtime
- SSH Access
- Scaling
- One-off jobs
- Support for persistent disks

Starter \$7 / month	512 MB (RAM) 0.5 CPU	Standard \$25 / month	2 GB (RAM) 1 CPU
Pro \$85 / month	4 GB (RAM) 2 CPU	Pro Plus \$175 / month	8 GB (RAM) 4 CPU
Pro Max \$225 / month	16 GB (RAM) 4 CPU	Pro Ultra \$450 / month	32 GB (RAM) 8 CPU

Need a [custom instance type](#)? We support up to 512 GB RAM and 64 CPUs.

- Next we just click the Deploy Web Service



- After we do that, it will take a couple of minutes to finish but now we can run our web app.

WEB SERVICE

LISUM39-irisapp Python 3 Free Upgrade your instance →

mynameisdevinchau / Data-Glacier-Internship main

<https://lisum39-irisapp.onrender.com>

Events Logs Disks Environment Shell Previews

ⓘ Your free instance will spin down with inactivity, which can delay requests by 50 seconds or more. Upgrade now

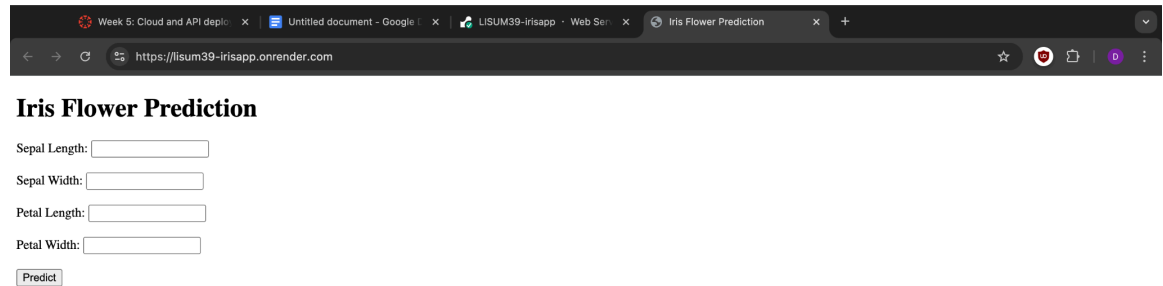
December 4, 2024 at 4:56 PM ✓ Live

8effd1b Add files via upload

All logs Search

Dec 4 04:58:12 PM Your service is live 🎉

- This is what the Web App looks like after deployment using Render



The screenshot shows a web browser window with the address bar displaying `https://lisum39-irisapp.onrender.com`. The page title is "Iris Flower Prediction". The form contains four input fields for "Sepal Length", "Sepal Width", "Petal Length", and "Petal Width", each followed by a "Predict" button.

Iris Flower Prediction

Sepal Length:

Sepal Width:

Petal Length:

Petal Width: