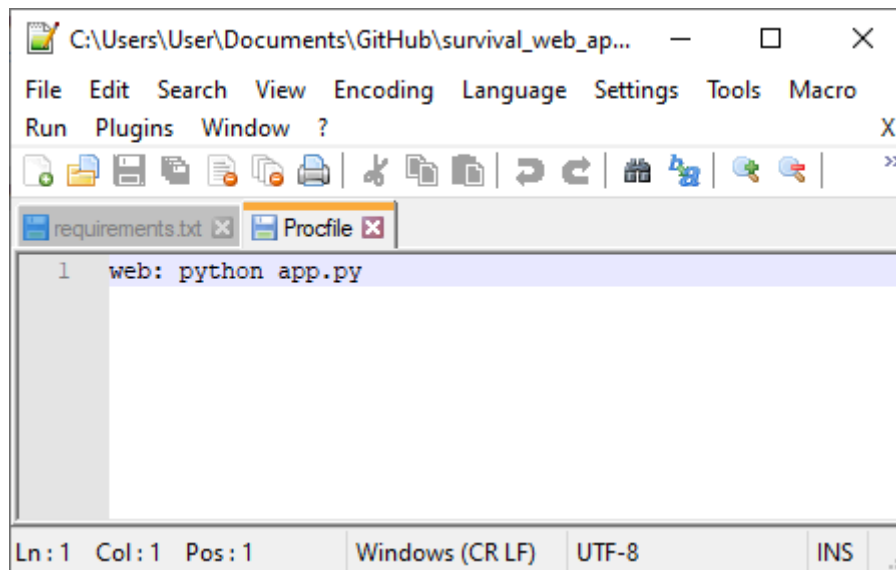


Name: Giorgos Moysiadis
Internship Batch Code: LISUM02
Submission Date: 21/8/2021
Submitted to: Data Glacier

Continuing with the data and the model from the previous week's assignment, I will be creating a separate repository to store only the saved pickle model, the template index.html page and the app.py module that has the app running.

After that I will add two files, Procfile and requirements.txt, that are required in order to deploy a web app on Heroku. The following screenshots show their content and functionality.

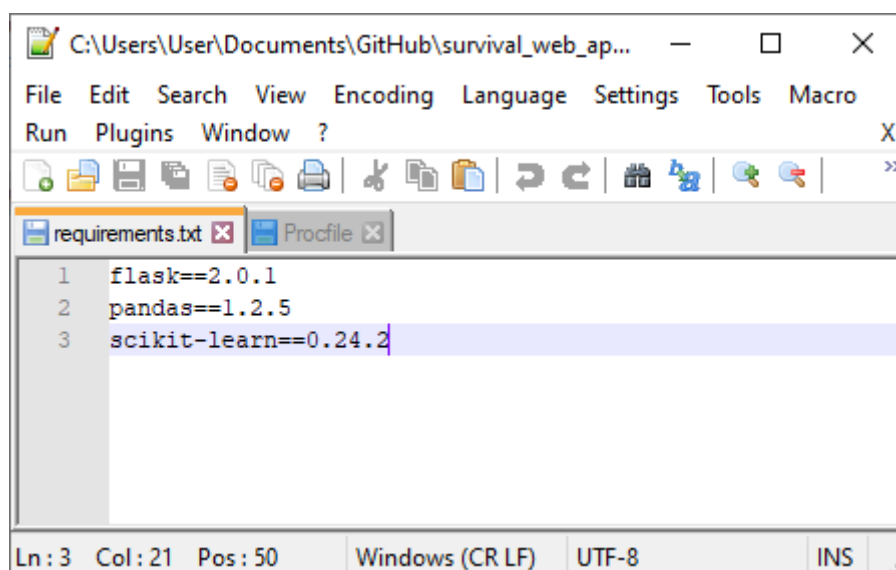


A screenshot of a text editor window titled 'C:\Users\User\Documents\GitHub\survival_web_ap...'. The window has a menu bar with 'File', 'Edit', 'Search', 'View', 'Encoding', 'Language', 'Settings', 'Tools', and 'Macro'. Below the menu bar is a toolbar with various icons. The editor shows two tabs: 'requirements.txt' and 'Procfile'. The 'Procfile' tab is active, displaying a single line of code: '1 web: python app.py'. The status bar at the bottom indicates 'Ln: 1 Col: 1 Pos: 1', 'Windows (CR LF)', 'UTF-8', and 'INS'.

```
1 web: python app.py
```

Procfile specifies the commands that are to be executed by the app on startup and is formatted as: <process type>: <command>.

When we ran our web app locally on the command line we simply wrote "python app.py" and it launched our app on our local machine. On Heroku we declare that our "web" app will run with "python app.py".

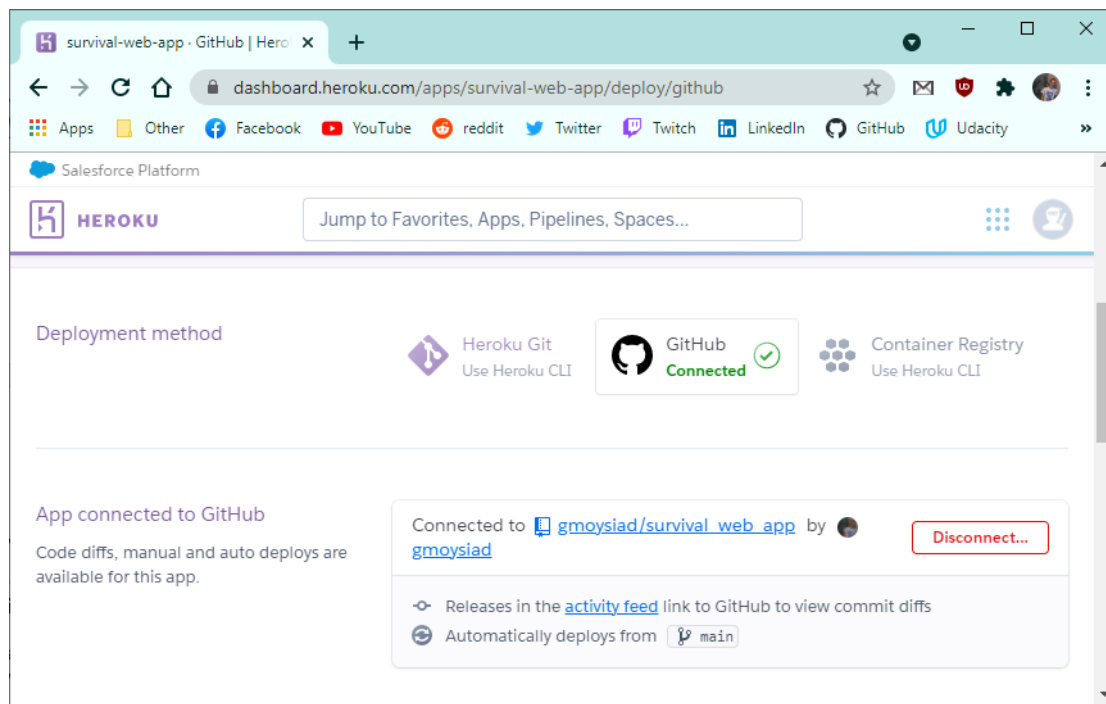


A screenshot of a text editor window titled 'C:\Users\User\Documents\GitHub\survival_web_ap...'. The window has a menu bar with 'File', 'Edit', 'Search', 'View', 'Encoding', 'Language', 'Settings', 'Tools', and 'Macro'. Below the menu bar is a toolbar with various icons. The editor shows two tabs: 'requirements.txt' and 'Procfile'. The 'requirements.txt' tab is active, displaying three lines of code: '1 flask==2.0.1', '2 pandas==1.2.5', and '3 scikit-learn==0.24.2'. The status bar at the bottom indicates 'Ln: 3 Col: 21 Pos: 50', 'Windows (CR LF)', 'UTF-8', and 'INS'.

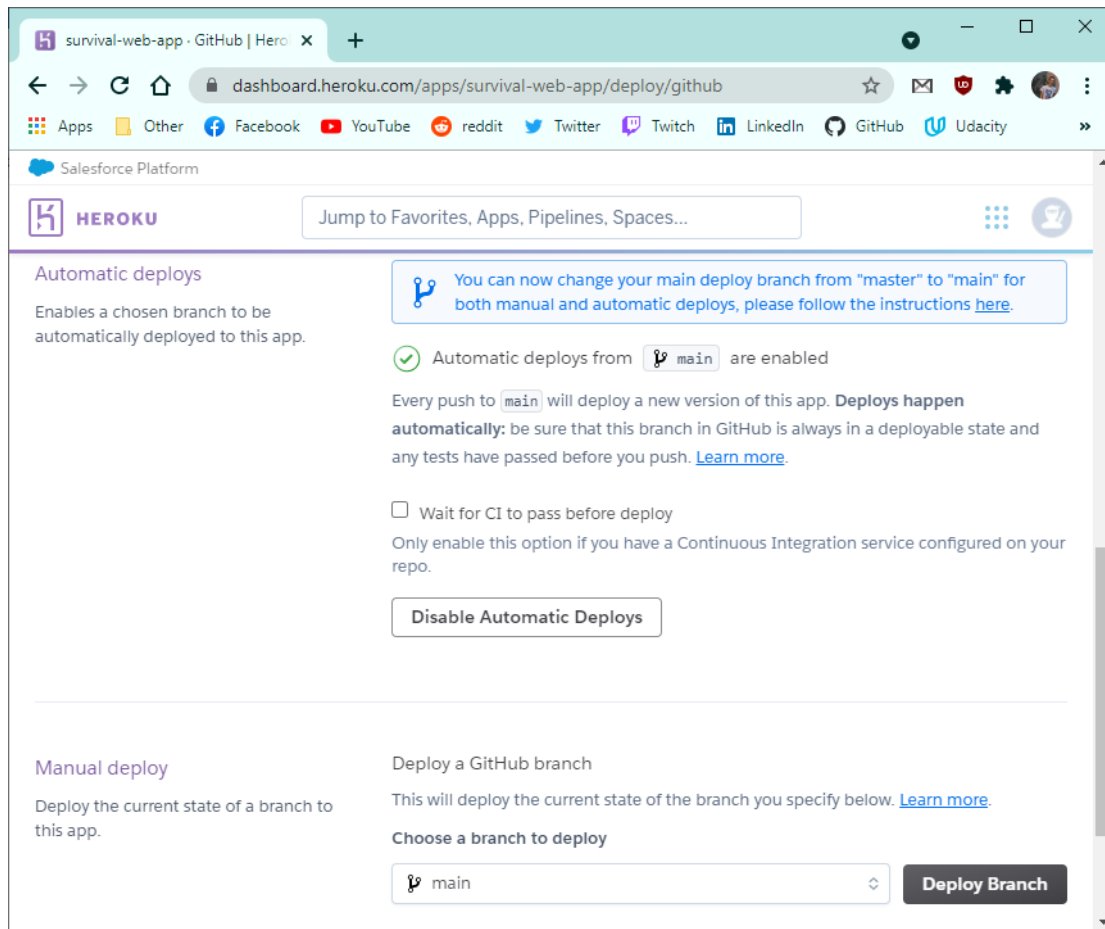
```
1 flask==2.0.1
2 pandas==1.2.5
3 scikit-learn==0.24.2
```

Heroku requires a “requirements.txt” file that contains the dependencies for our Python application and are installed via “pip”. From the above screenshot we can see that web app requires flask, pandas and scikit-learn in order to work correctly.

After creating the GitHub repository, I am going to login to Heroku and create a new app and I am going to call it “survival-web-app”.



After selecting my app, I am going to the “Deploy” tab and select to connect a GitHub repository to associate with the app. The screenshot above shows that my GitHub repository is connected with the web app that I created with Heroku.



After connecting the GitHub repository, I am going to select a branch that I want to deploy down on the “Manual deploy” section. After clicking “Deploy Branch” there is a small dialog that shows the steps that are running in order to deploy the web app for the first time. Looking closely one can notice that the console runs a “pip” command for our dependencies that we have provided with the “requirements.txt”.

Once the process is finished we can select “Enable Automatic Deploys” which helps with automatic deploys every time we are making a change in our project repository.

When we open our app we can see that it is ready for use, and since it’s running I’m going to provide with some input (Passenger Class=2, Sex=Male, Fare=70, Cabin Type=5) and see our results that are running correctly.

survival-web-app - GitHub | Heroku x Titanic Web App x +

survival-web-app.herokuapp.com/predict

Apps Other Facebook YouTube reddit Twitter Twitch LinkedIn GitHub Udacity »

Predict survival of Titanic passenger

Passenger Class (1-3)

Sex

Male

Fare

Cabin Type (0-8)

Predict

Passenger did not survive

As it can be seen from the screenshot above, the app is running on “survival-web-app.herokuapp.com” and with the input that I have provided (mentioned above) the model has predicted that our passenger has not survived.