Flask Intro

Folder Structure

flask-app/

app.py

model.joblib

Procfile

requirements.txt

templates/

index.html

results.html

Heroku Intro:

Heroku is what is known as a Platform as a Service or (PaaS) and at a high level it could be described as a service that allows businesses to quickly build, deploy, manage, and scale applications while removing the hassle that normally comes from the infrastructure required for this type of activity.

In this course, we’re only really going to scratch the surface of what can be done on Heroku…but in saying that, this is going to be really great experience to get your hands dirty within their system…and when I say get your hands dirty, I actually feel like that’s doing Heroku a disservice…as you’ll see everything about how they operate is so clean and intuitive.

Heroku Tutorial

* Create new github repo ….. purchase-prediction
* Add files from flask app folder (add templates folder separately) - commit
* Sign up to Heroku
* “New App”
* Name it, don’t worry too much about region, leave as is
* “Github Connect”
* Allow it to connect to your github account
* Once connected, find your purchase-prediction repo and connect to it
* Auto deploy to always be updating live or manual deploy to deploy only upon request
* Manually deploy
* It will load the required packages from our requirements.txt file as well as anything else it needs
* Then it will load our app to a herokuapp site
* Top right “more” – view logs…if you encounter errors this is a good place to understand what the error message was. Often it is down to a requirement not being satisfied, or even a package version containing a different interpretation. If you can get it to run locally, it’s not a bad idea to use the exact same package versions to ensure it running here