Cloning From This Point On

If you do choose to skip this section and just fork and clone this repo, or any repo from this point on in the course, remember to add a file called .env to the root folder! In that .env file remember to add a STRIPE\_SECRET\_KEY value equal to your own secret key from your stripe dashboard. You can find it in the same place where you found your publishable key in the developers tab under api keys. You will have to enter the password in to reveal it!

A screenshot of a cell phone

Description automatically generated

You will also need to connect your existing Heroku app to this new forked and cloned repo, or you have to create a new Heroku app and push to it. A quick refresher on how to do either of these:

**Set to an existing Heroku app**

To set to an existing Heroku app you already have deployed, you need to know the name of the app you want to deploy to. To see a list of all the apps you currently have on Heroku:

1. heroku apps

Copy the name of the app you want to connect the project to, then run:

1. heroku git:remote -a <PASTE\_YOUR\_APP\_NAME\_HERE>

And now you'll have your repo connected to the heroku app under the git remote name heroku.

If the Heroku app you connected was deploying just a create-react-app project from earlier in the lesson, you will need to remove the mars/create-react-app-buildpack buildpack first. You can check if you have this buildpack by running:

1. heroku buildpacks

Which will list any buildpacks you currently have, if you see mars/create-react-app-buildpack in the list, you can remove it by running:

1. heroku buildpacks:remove mars/create-react-app-buildpack

Then skip to the bottom of this article to see what to do next!

**To create a new Heroku app**

Create a new Heroku project by typing in your terminal:

1. heroku create

This will create a new Heroku project for you. Then run:

1. git remote -v

You should see heroku https://git.heroku.com/<RANDOMLY\_GENERATED\_NAME\_OF\_YOUR\_APP> in the list. This means you have successfully connected your project to the newly created Heroku app under the git remote of heroku.

**Deploying to Heroku**

Before we deploy, you also need to set a config variable of STRIPE\_SECRET\_KEY to the same secret key value from your stripe dashboard, the same one in your .env file. The .env file is only for local development, in order for our heroku production app to have access to this secret key, we add it to our Heroku projects config variables by typing:

1. heroku config:set STRIPE\_SECRET\_KEY=<YOUR\_STRIPE\_SECRET\_KEY>

After that, you can deploy to heroku by running:

1. git push heroku master

**You will see this warning message if you are pushing to an existing app:**

1. ! [rejected] master -> master (fetch first)
2. error: failed to push some refs to 'https://git.heroku.com/<YOUR\_HEROKU\_APP\_NAME>'
3. hint: Updates were rejected because the remote contains work that you do
4. hint: not have locally. This is usually caused by another repository pushing
5. hint: to the same ref. You may want to first integrate the remote changes
6. hint: (e.g., 'git pull ...') before pushing again.
7. hint: See the 'Note about fast-forwards' in 'git push --help' for details.

This is because we are pushing to an existing app that was deploying an entirely different repository from what we have now. Simply run:

1. git push heroku master --force

This will overwrite the existing Heroku app with our new code.

**Open our Heroku project**

After heroku finishes building our project, we can simply run:

1. heroku open

This will open up our browser and take us to our newly deployed Heroku project!