



Deploying Rails to Heroku



To prepare

1. **Create a heroku account**

If you don't already have one .. it's free

2. **Install the heroku CLI**

<https://devcenter.heroku.com/articles/heroku-cli#download-and-install>

3. **Login to heroku (from the CLI)**

`heroku login`



You need an app to deploy

1. **Create the app**

Create your basic running application

2. **Git it**

Initialise a git repository (`git init`) and commit your work (`git add . && git commit -m "initial commit"`)

It's also a good idea to push it up to a repo on github. This isn't required for deployment, but it is needed for a healthy development process (more about this later)



Create the heroku app

```
heroku create <app-name>
```

- app-name is optional - heroku will make up a name if you don't give one
- You can update the app name later

After running heroku create, you should see the connection to the repository hosted on heroku with **git remote -v**



Set the master key

Rails uses a master key for decryption of its credentials. You never want to push this up to your repository, but you need to let heroku know the key value for your app to work

The master key is in the config directory of your app in a file called master.key

```
heroku config:set RAILS_MASTER_KEY=<your-key>
```

Now push to heroku master

```
git push heroku master
```

Set up the database



Heroku runs its own postgresql server instance

Just like when you set up the app to run in your local development environment, you need to set up the database to run in the heroku (production) environment, which also has a postgresql server instance deployed for your app to use

For initial setup, some commands don't work in the heroku environment (`db:setup`, `db:create`) because we don't have access to the postgres user password, but we can accomplish what we need with other db commands

We can run rails commands in the heroku environment from the heroku CLI using the **heroku run** command



Heroku rails db commands

We will use:

- `heroku run rails db:schema:load` to initially create our tables from the `db/schema.rb` file
This will clear all data, so only use it for initial setup
- `heroku rails db:migrate` if we make any changes to our db structure (tables, columns) after we deploy to produce
- `heroku rails db:seed` to seed any data we need in production (not always necessary but sometimes makes sense)

We will use seed to create our menu items



Check the database.yml

You should be using postgresql, and if you created your app with `-d postgresql`, the database.yml file will already be properly set up to run in production. You should see something like this for production in your database.yml file:

```
production:
  <<: *default
  database: cafe_production
  username: cafe
  password: <%=
ENV[ 'CAFE_DATABASE_PASSWORD' ] %>
```



Issue db:schema:load and db:seed

Initialise the database tables (create them) from our db/schema.rb

```
heroku run rails db:schema:load
```

Seed the menu items data

```
heroku run rails db:seed
```



Open the app in heroku

You can open the heroku app from heroku, or from the heroku cli in the app directory

```
heroku open
```

If there have been no errors, you should see your app open in a browser window - deployed on heroku!
You will be able to access this deployed version of your app from any network connected device.



Resources

[Deploying Rails5 to Heroku](#)

[Project workflow](#)