Hyperiondev

Task

Deploy a React App

Visit our website

Introduction

GETTING YOUR APP OUT THERE

At this point, you've seen multiple ways of deploying your apps. From containerisation with Docker to static site hosting with GitHub Pages. Today we'll look at another way of deploying your app — Heroku.

Get in touch

Connect for support

Remember that with our courses, you're not alone! You can contact an expert code reviewer to get support on any aspect of your course.

The best way to get help is to login to Discord at **https://discord.com/invite/hyperdev** where our specialist team is ready to support you.

Our expert code reviewers are happy to offer you support that is tailored to your individual career or education needs. Do not hesitate to ask a question or for additional support!

WHAT IS HEROKU?

Much like GitHub Pages, Heroku is a Platform-as-a-Service (PaaS) provider that allows developers to host their applications online with minimal manual setup. One remarkable difference between the two is that Heroku is able to host much more than static sites. It can also be used to host databases, runtimes for many languages, and integrations for many third-party services like SendGrid, Papertrail, and MongoDB.

So, in a nutshell, you can use Heroku to deploy just about any web-based app you want. One snag however is that to use most of these features you have to build your app from the ground up with Heroku in mind (using their services and backends). So, switching between PaaS providers is a massive hassle, and becomes nearly impossible the bigger your project gets. This is why large scale projects often use their own servers and environments. Nevertheless, Heroku is still great for small projects (like this task), and is substantially cheaper than many of the alternatives.

There are many PaaS alternatives to Heroku that work in much the same way, and are priced similarly too, including **AWS**, **Azure**, and **Firebase**. These providers typically allow a certain amount of free platform usage, and only start charging when you need complex features, or have many users.

OUTLINE FOR USING HEROKU

In order to use Heroku to run your web applications, you have to put a copy of your web application on Heroku. Heroku can then build your application. Building your code results in a slug. **Heroku** defines a slug as “a bundle of your source, fetched dependencies, the language runtime, and compiled/generated output of the build system - ready for execution.” Your package.json file includes a lot of important information that is needed for this build, including the dependencies and the information that specifies which file to use to run the app that is specified in the main field. Your app can then be run on virtual Unix-based web servers called dynos.

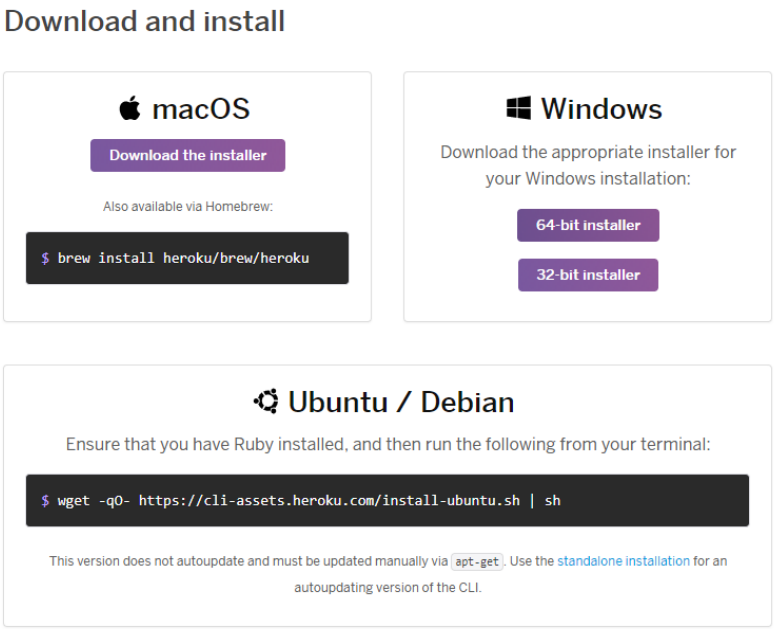
SIGN UP FOR A HEROKU ACCOUNT

You are going to have to sign up for a Heroku account in order to be able to deploy your app to Heroku. To do this, fill in the form **here**.

DEPLOYING YOUR APP TO HEROKU

The simplest way to deploy your app to Heroku is to do so using git and the Heroku command line interface (CLI). You can do this in a few simple steps as shown below.

1. Download and install the Heroku CLI as instructed **here**. Once you have followed the appropriate instructions to install the Heroku CLI, test that it has been properly installed by typing `heroku -v` in the command line interface of your PC. You should see which version of Heroku CLI has been installed if the installation ran correctly.



2. Open your command line interface and navigate to the project directory of the last web app you created using Create React App.
3. Make this directory a git repository if it isn't yet. This git repository will be on your local machine. You will later push this to a Heroku server. Do this by typing `git init`. You won't have to do this if you have used Create React App to create your app since a git repository would have been created by default for your project.
4. Login to Heroku by typing `heroku login` in the CLI. You will be prompted for your password. Use the password you created when you created your Heroku account to login.
5. Create an empty application on Heroku using the official buildpack provided by Heroku to build apps created using Create React Apps. Do this by typing:

```
heroku create -b https://github.com/mars/create-react-app-buildpack.git
```

More details **here**.

6. Commit and push your app to Heroku by typing the following instructions:
    - `git add .`
    - `git commit -m "react-create-app on Heroku"`
    - `git push heroku master`
7. Test your app by typing `heroku open` in the CLI. This should open your app in the browser if everything worked correctly. The address you see in the browser's top bar is accessible to anyone on Earth.

Compulsory Task 1

Follow these steps:

- In your previous Project you created a basic App using Create React App. For this Task, you are required to deploy this app to Heroku. Do this by following the instructions in this document.
- Create a text document called **link.txt** in which you provide the expert code reviewer with the URL they can use to visit your deployed app.

Optional Bonus Task

For all future (and perhaps past) capstones that consist of no more than a website, create a Heroku dyno and host it online. Add the links to the appropriate GitHub repos, so visitors can see your code in action.

Note that using Heroku for dynamic websites that rely on state, e.g. with databases and backends, is a little more complicated than this task. It will require extra configuration and dependencies in your projects. If you're interested, check Heroku's **website**.

Rate us

Share your thoughts

HyperionDev strives to provide internationally-excellent course content that helps you achieve your learning outcomes.

Think the content of this task, or this course as a whole, can be improved or think we've done a good job?

Click here to share your thoughts anonymously.

REFERENCE

React.js. (2020). Getting Started – React. Retrieved 6 August 2020, from **https://reactjs.org/docs/getting-started.html**