

Angel Santoyo

CPSC 349

Fall 2022

Exercise 2

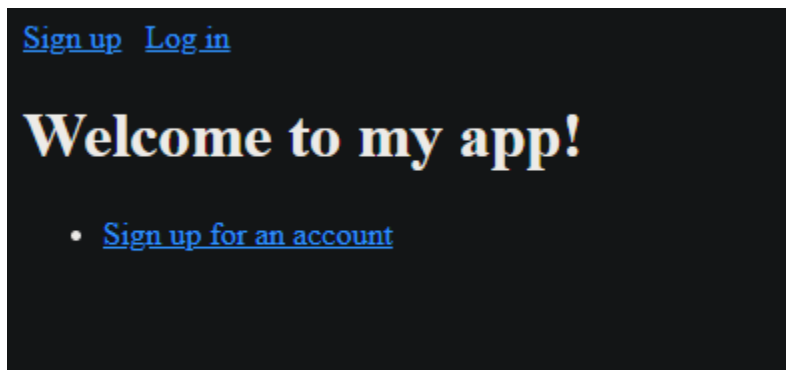
Github repo: <https://github.com/Angus-1/cpsc349-project2.git>

## Steps to complete this exercise

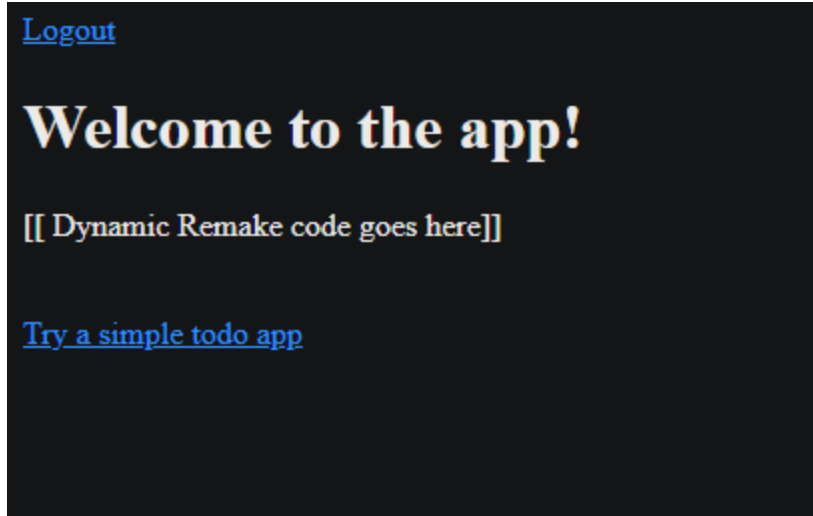
1. In order for the repository to be created correctly in the next step, the Git command-line client needs to be configured with your name and email address. Check that the following commands print those values:

```
PS D:\github\cpsec349-project2> git config --global user.name  
Angel  
PS D:\github\cpsec349-project2> git config --global user.email  
santoyox714@gmail.com
```

- 2) Complete steps 1-3 of the Remake [Getting Started](#) tutorial.



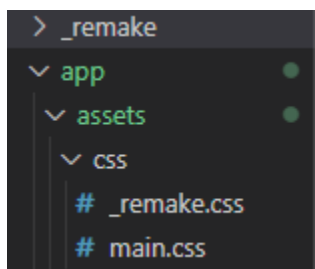
- 3) Click the “Sign up for an account” link, create an account, and verify that you can see the “Welcome to the app!” page.



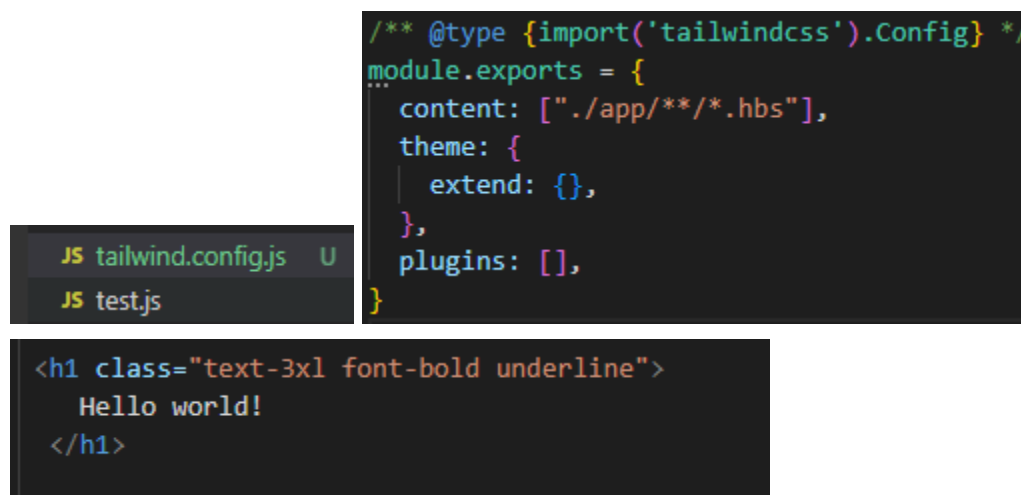
○

4) Check the contents of the file `app/assets/css/main.css`. What do you find?

*Two css files, `_remake` and `main`, `main` is empty*

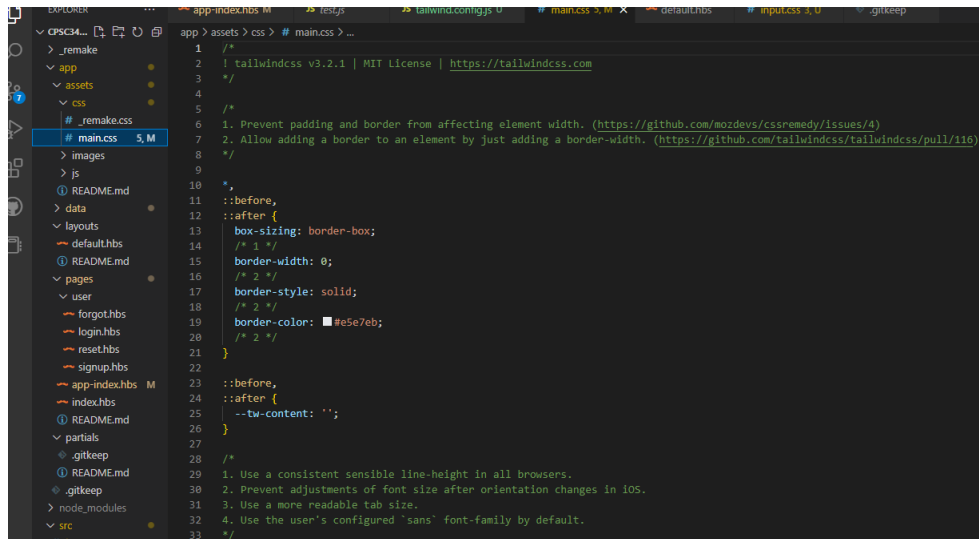


5) In step 5, the HTML to edit is in `./app/pages/app-index.hbs`. You do not need to add a link to the stylesheet (the link is already defined in `./app/layouts/default.hbs`), but you should add the classes shown to the `<h1>` tag.



6) Verify that the `npx tailwindcss ... --watch` command detected the change to `app-index.hbs` and rebuilt `main.css`. How do the contents of the file compare to Step (4)?

What's different is now `main.css` has `tailwind v3.2.1` inside of it and is no longer empty.



1. Open `package.json` and examine [the scripts property](#). You will see that the `npm run dev` command that you ran earlier is defined here.

Define a script named "tailwind" to run the `tailwindcss ... --watch` command you ran earlier, without two changes. (You may omit the `npx` command at the beginning when running inside a `package.json` script.) Verify that it starts correctly when you run the following command:

```
npm run tailwind
```

```
11 "scripts": {
12   "dev": "cross-env NODE_ENV=development nodemon _remake/server.js --watch ./",
13   "remake-build": "npx esbuild _remake/client-side/index.js --bundle --minify --sourcemap",
14   "test": "ava",
15   "tailwind": "npx tailwindcss -i ./src/input.css -o ./app/assets/css/main.css --watch"
16 },
```

PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

PS D:\github\cpssc349-project2> npm run tailwind

> my-app@1.0.0 tailwind  
> npx tailwindcss -i ./src/input.css -o ./app/assets/css/main.css --watch

Rebuilding...

Done in 278ms.

8) Now we have two scripts that need to run at the same time — one to run the server, and one to rebuild main.css when .hbs files are changed. To do this, we'll need another utility, [npm-run-all](#). Add npm-run-all as a development dependency with the following command:

```
npm install -D npm-run-all
```

```
PS D:\github\cpssc349-project2> npm install -D npm-run-all

added 40 packages, changed 6 packages, and audited 1021 packages in 5s

95 packages are looking for funding
  run `npm fund` for details
```

9) Now add a script named "start" to run the following command:

```
npm-run-all --parallel dev tailwind
```

Start both scripts by running `npm start`

```
10  },
    > Debug
11  "scripts": {
12    "dev": "cross-env NODE_ENV=development nodemon _remake/server.js --watch ./",
13    "remake-build": "npx esbuild _remake/client-side/index.js --bundle --minify --sourcemap=external",
14    "test": "ava",
15    "tailwind": "npx tailwindcss -i ./src/input.css -o ./app/assets/css/main.css --watch",
16    "start": "npm-run-all --parallel dev tailwind"
17  },
18  "nodemonConfig": {
19    "ignore": [
20      "node_modules",
21      "public"
22    ],
23    "watch": [
24      "src"
25    ],
26    "exec": "npm start"
27  }
28 }

```

PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

PS D:\github\cpssc349-project2> **npm** start

> my-app@1.0.0 start  
> npm-run-all --parallel dev tailwind

> my-app@1.0.0 dev  
> cross-env NODE\_ENV=development nodemon \_remake/server.js --watch ./

> my-app@1.0.0 tailwind  
> npx tailwindcss -i ./src/input.css -o ./app/assets/css/main.css --watch

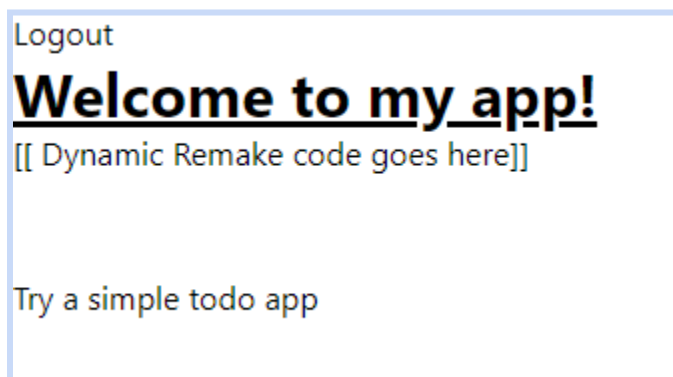
[nodemon] 2.0.13  
[nodemon] to restart at any time, enter `rs`  
[nodemon] watching path(s): \*\*\\*  
[nodemon] watching extensions: js,json,svg,png,gif,jpg,jpeg,ico,mp3,mp4,mpg,mpeg,ogg,otf,ttf,woff,woff2,txt,webp  
[nodemon] starting `node \_remake/server.js`

Rebuilding...

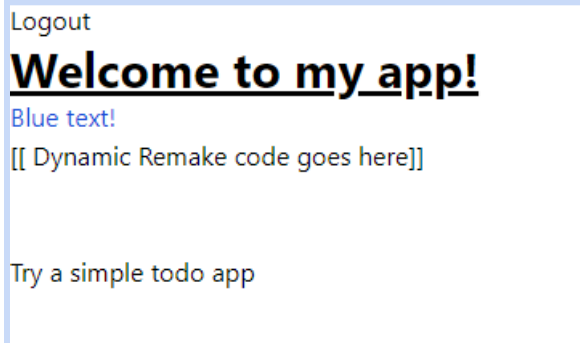
Done in 446ms.

Visit your Remake app: <http://localhost:3100>  
Check this log to see the requests made by the app, as you use it.

10) Return to the “Welcome to the app!” page you opened in Step (3). Verify that the heading now has the styles you added to `app-index.hbs` in Step (5).



Add another class to the <h1> tag (e.g. text-blue-700) and verify that the page is updated in the browser when you save the .hbs file.



```
<h1 class="text-3xl font-bold underline">
  Welcome to my app!
</h1>
<h1 class="text-blue-700"> Blue text! </h1>
<div>[[ Dynamic Remake code goes here]]</div>
<br><br>
<a href="https://docs.remaketheweb.com/a-simple-example-app/">Try a simple todo app</a>
```

11) Submit your results as a single PDF file uploaded to Canvas by the deadline. If you work with another student, only one submission is required as long as it includes the names of both students.