

COMP 2537 Web Development 2

Lab 2

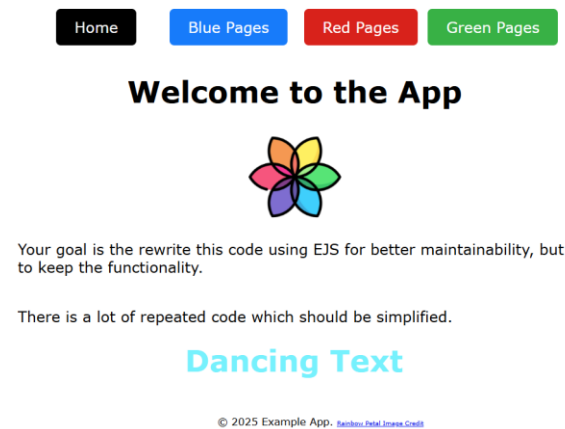
COMP 2537 Web Development 2

Lab 2 Exercise

Introduction:

In the sample code provided there is a simple website with several pages. These pages are very similar and share a lot of common code. This code base, although very small at this point, is hard to maintain due to how much code is repeated. The goal of this lab is to use the EJS templating engine to simplify the code and, when possible, only write common code in a single place.

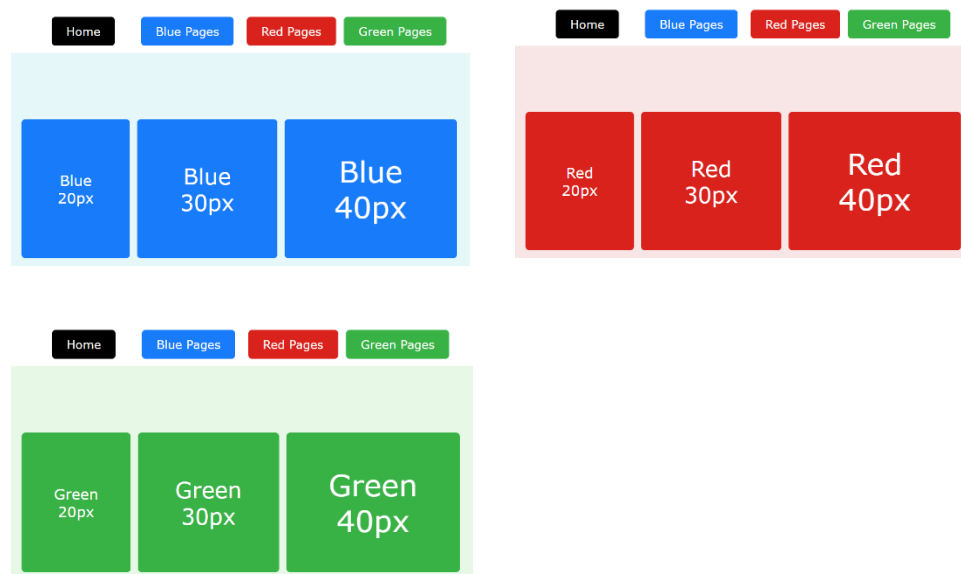
The home page looks like this:



(if it is working properly, it should be styled like this and the text "Dancing Text" should spin around in a circle, while changing color - rotating through the colors of the rainbow)

The navbar at the top, links to 3 "color index" pages: blue, red and green (as well as linking back to the home page).

The color index pages (/blue, /red, /green) look like this:



Notice that the color of the background matches the color in the URL (ex: the blue index has a light blue color; red is red and green is green).

Clicking on the 20px, 30px and 40px buttons on the "color index" pages will take you to "color size" pages where the font size matches that size
(ex: on the 40px page, the font size is 40 pixels):

Page: /blue/20



Page: /green/40



The "color size" pages each have a button and some associated javascript which changes the background color of the inner container div to the color that matches the color of the page (ex: green background for a green page).

COMP 2537 Web Development 2
Lab 2 Exercise

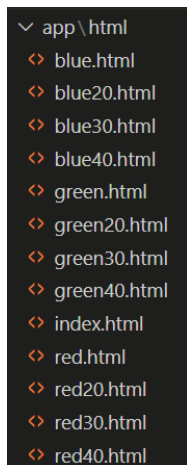
Below is the `/green/40` page after pressing the button:



Notice how every page shares the same navbar at the top and the same footer at the bottom.

Here is the current file structure of the project:

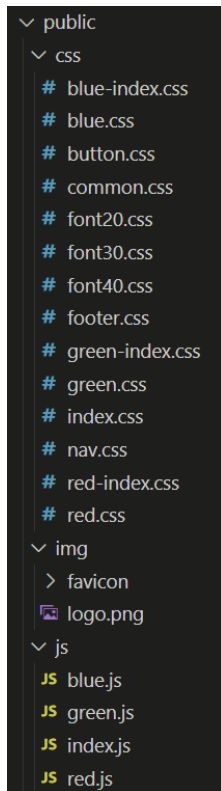
The HTML files that are loaded are in `/app/html`. The home page is `index.html`; the "color index" pages are: `blue.html`, `red.html` and `green.html`; and the "color size" pages are `blue20.html`, `blue30.html`, `blue40.html`, `red20.html`, etc. (one for each color and font size combination).



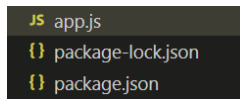
COMP 2537 Web Development 2

Lab 2 Exercise

The `/public` folder has several different folders: `css`, `img` and `js`. The `img` folder is fairly self-explanatory, there is a logo and a folder for various favicon images. The `js` files are specific for specific pages and don't need to (and shouldn't be) included on all pages. The `index.js` file is used ONLY on the home page and `blue.js` is used on the blue "color size" pages (`blue20.html`, `blue30.html`, `blue40.html`) ONLY. Similarly there are several CSS files. Some of the CSS files are used on all pages (such as `common.css`, `footer.css` and `nav.css`) while others are ONLY used on some pages (`blue.css` is used on all blue pages - `blue.html`, `blue20.html`, `blue30.html`, and `blue40.html`; and `font20.css` is used only on font size 20 pages - `blue20.html`, `red20.html` and `green20.html`)



The main server-side js file responds to the browser requests is `app.js`:



COMP 2537 Web Development 2

Lab 2 Exercise

Redundant Code:

There are a number of places where code is duplicated.

1. HTML:

All pages have the very similar headers:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <!-- Common CSS -->
  <link rel="stylesheet" href="/css/nav.css">
  <link rel="stylesheet" href="/css/common.css">
  <link rel="stylesheet" href="/css/footer.css">
  <!-- CSS for Index -->
  <link rel="stylesheet" href="/css/index.css">
  <!-- Favicon -->
  <link rel="apple-touch-icon" sizes="180x180" href="/img/favicon/apple-touch-icon.png">
  <link rel="icon" type="image/png" sizes="32x32" href="/img/favicon/favicon-32x32.png">
  <link rel="icon" type="image/png" sizes="16x16" href="/img/favicon/favicon-16x16.png">
  <link rel="manifest" href="/img/favicon/site.webmanifest">

  <title>Example App</title>
</head>
```

the same navbar:

```
<div class="nav-background">
  <nav class="navbar">
    <ul class="nav-links">
      <li><a href="/">Home</a></li>
      <li><a class="blue" href="/blue">Blue Pages</a></li>
      <li><a class="red" href="/red">Red Pages</a></li>
      <li><a class="green" href="/green">Green Pages</a></li>
    </ul>
  </nav>
</div>
```

and the very similar footers:

```
<footer class="footer">
  <span>&copy; 2025 Example App.</span>
  <span class="img-credit">
    <a href="https://www.flaticon.com/free-icon/lotus-flower_3810196">
      Rainbow Petal Image Credit
    </a>
  </span>
</footer>
<script src="/js/index.js"></script>
</body>
</html>
```

The headers and footers on each of the pages only differ by which CSS files (in the header) and JS files (in the footer) are included.

2. Express routes:

The "color index" pages all have similar routes:

```
app.get('/blue', (req, res) => {
  let html = fs.readFileSync(__dirname + '/app/html/blue.html', 'utf8');
  res.send(html);
});
app.get('/red', (req, res) => {
  let html = fs.readFileSync(__dirname + '/app/html/red.html', 'utf8');
  res.send(html);
});
app.get('/green', (req, res) => {
  let html = fs.readFileSync(__dirname + '/app/html/green.html', 'utf8');
  res.send(html);
});
```

COMP 2537 Web Development 2
Lab 2 Exercise

And so do the "color size" pages:

```
app.get('/blue/20', (req, res) => {  
  let html = fs.readFileSync(__dirname + '/app/html/blue20.html', 'utf8');  
  res.send(html);  
});  
app.get('/blue/30', (req, res) => {  
  let html = fs.readFileSync(__dirname + '/app/html/blue30.html', 'utf8');  
  res.send(html);  
});  
app.get('/blue/40', (req, res) => {  
  let html = fs.readFileSync(__dirname + '/app/html/blue40.html', 'utf8');  
  res.send(html);  
});
```

(these same routes are repeated again, but for red and green color size pages).

COMP 2537 Web Development 2
Lab 2 Exercise

Instructions:

Redesign this website using EJS to reduce code duplication.

1. Use EJS for all HTML content.
 - Convert all `.html` files to `.ejs` files and move them to the `/views` folder
 - Convert all `res.send` to `res.render`.
 - You should only need 3 `ejs` files:
 - o `index.ejs`
 - o `color.ejs`
 - o `color-size.ejs`
 - Pass in any applicable arguments into the `res.render` to provide the slight differences in the pages (ex: the `/blue`, `/red` and `/green` color index pages only differ slightly but will share the same `color.ejs` file.
2. Create a `header.ejs` which will contain the common header code.
 - Create a `/views/partials` folder that will contain common code.
 - Move all common header code out of each of the individual `.ejs` files
 - Make the header dynamic
 - o The `header.ejs` file should expect an array of CSS files as a parameter.
 - o Loop through the CSS array to include CSS files specific to each page
 - o Modify all pages to include the new `header.ejs` which ONLY the CSS files that are required for that page.
3. Create a `footer.ejs` which will contain the common footer code.
 - Create `footer.ejs` in the `/views/partials` folder.
 - Move all common header code out of each of the individual `.ejs` files
 - Make the footer dynamic
 - o The `footer.ejs` file should expect an array of CSS files as a parameter.
 - o Loop through the JS array to include JS files specific to each page
 - o Modify all pages to include the new `footer.ejs` which ONLY the JS files that are required for that page.

COMP 2537 Web Development 2
Lab 2 Exercise

Marking Guide:

Criteria	Marks
Convert all pages to use EJS (no more <code>res.send()</code>). The website after the conversion to EJS should work the same as before. All CSS styles should apply as before. (ex: blue pages are blue, size 40 pages have font of 40px). All JS should work (ex: dancing text on the home page, button presses on the color size pages). Demo the home page, all index pages (blue, red, green) and demo the button presses on at least color size pages (ex: blue 20, red 40 and green 30).	5 marks
Put all the common header and footer code into <code>header.ejs</code> and <code>footer.ejs</code> . Show your <code>header.ejs</code> and <code>footer.ejs</code> . Show your home page <code>ejs</code> file and one of the color size pages. The CSS and JS files must be dynamic (ONLY include the files required for that page and no more - do NOT hard code each page to include all the CSS and JS files).	5 marks
You should need no more than 3 routes: 1. homepage 2. color index pages 3. color size pages Show your routes in your <code>app.js</code> file.	5 marks
Total:	15 marks

Submission Requirements:

Submission:
Show your lab instructor during lab time. No submission needed (unless you can't make it to class).