CS628 Full-Stack Development Web App

**PE01 – Resume**

School of Technology & Computing (STC)

City University of Seattle (CityU)

**Before You Start**

* You already created a private GitHub repository for all your programming exercises, “cs628-pe-your\_first\_name.”
* You allowed your instructor and the Teaching Assistant to access your GitHub repository for programming assignments.
* The GitHub Codespaces may bill your account according to your usage. Check the price at <https://docs.github.com/en/billing/managing-billing-for-github-codespaces/about-billing-for-github-codespaces>. Please pay attention to the storage and core hours of use free of charge for personal accounts.
* Some steps are not explained in the assignment**.**If you are not sure what to do:
  + Consult the resources listed in your course.
  + If you need help solving the problem after a few tries (~15 minutes), ask a TA for help.

**Learning Outcomes**

Students will be able to:

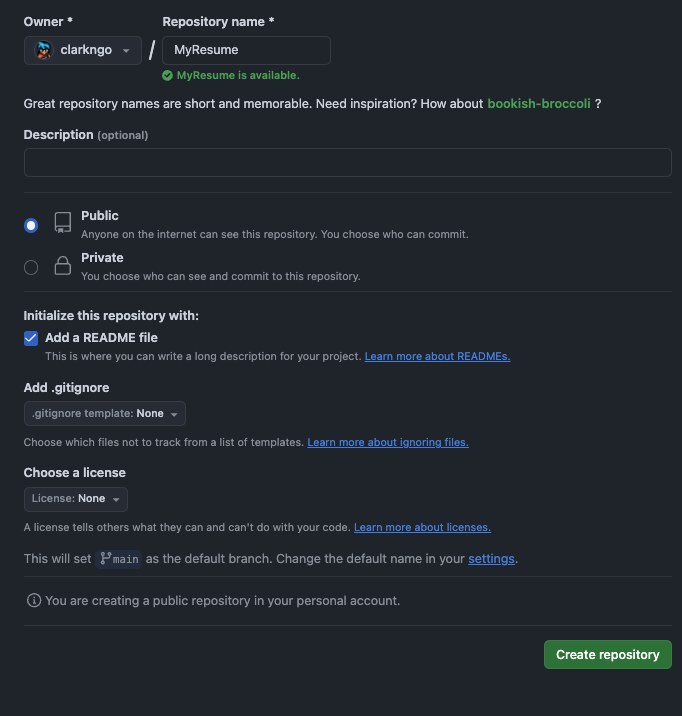
* + Create a client web app using Codespaces and React.

**Create a GitHub Repository**

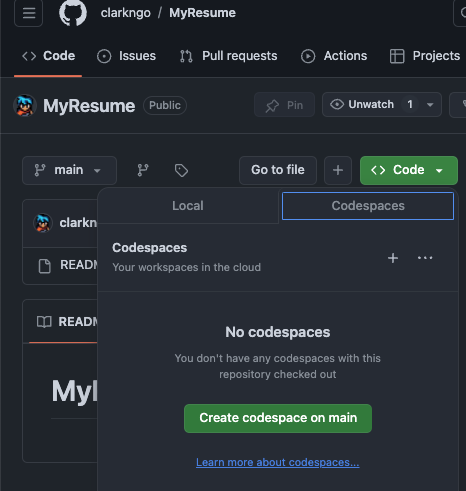
1. Go to your GitHub account and click , then New Repository.



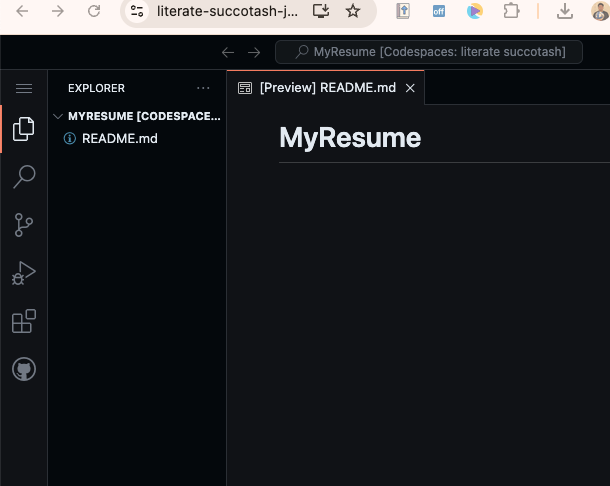
1. In the Create a new repository page
   1. Repository name: **MyResume**
   2. Tick the checkbox for **Add a README file**
   3. Click **Create repository**



1. Open GitHub Codespaces
   1. Click **Code**
   2. Click **Codespaces**
   3. Click **Create codespace on main**



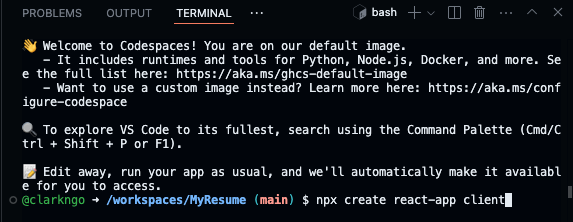
**Sample Output**



**Setup create-react-app**

To create a React app, you can use Create React App. It's a tool developed by the React team that sets up a new React project with all the necessary configuration and build tools.

Run the following command in the Terminal to create a new React app:



**>> npx create-react-app** *client*

This command will create a new directory called **client** and set up the initial React project.

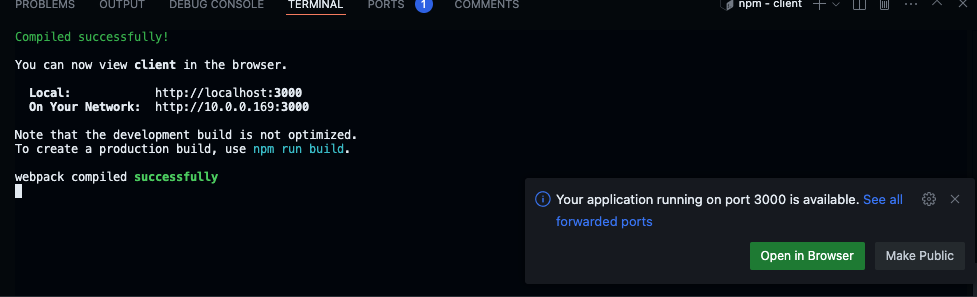
After the command completes, navigate into the newly created app directory:

**>> cd client**

Finally, you can start the development server and open the app in your browser by running the following command:

**>> npm start**

Click **Open in Browser** if it does not automatically open a new tab.



Alternatively, you can access the URL in the Ports tab



This command will start the development server and automatically open the app in your default browser. Now you can build your React app by editing the source files in the '****src****' directory.

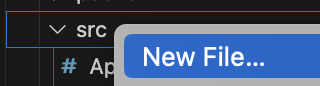
If the command does not work, install the dependencies to the local ‘**node\_modules**’ folder by typing ‘**npm install**.’

**Problem**

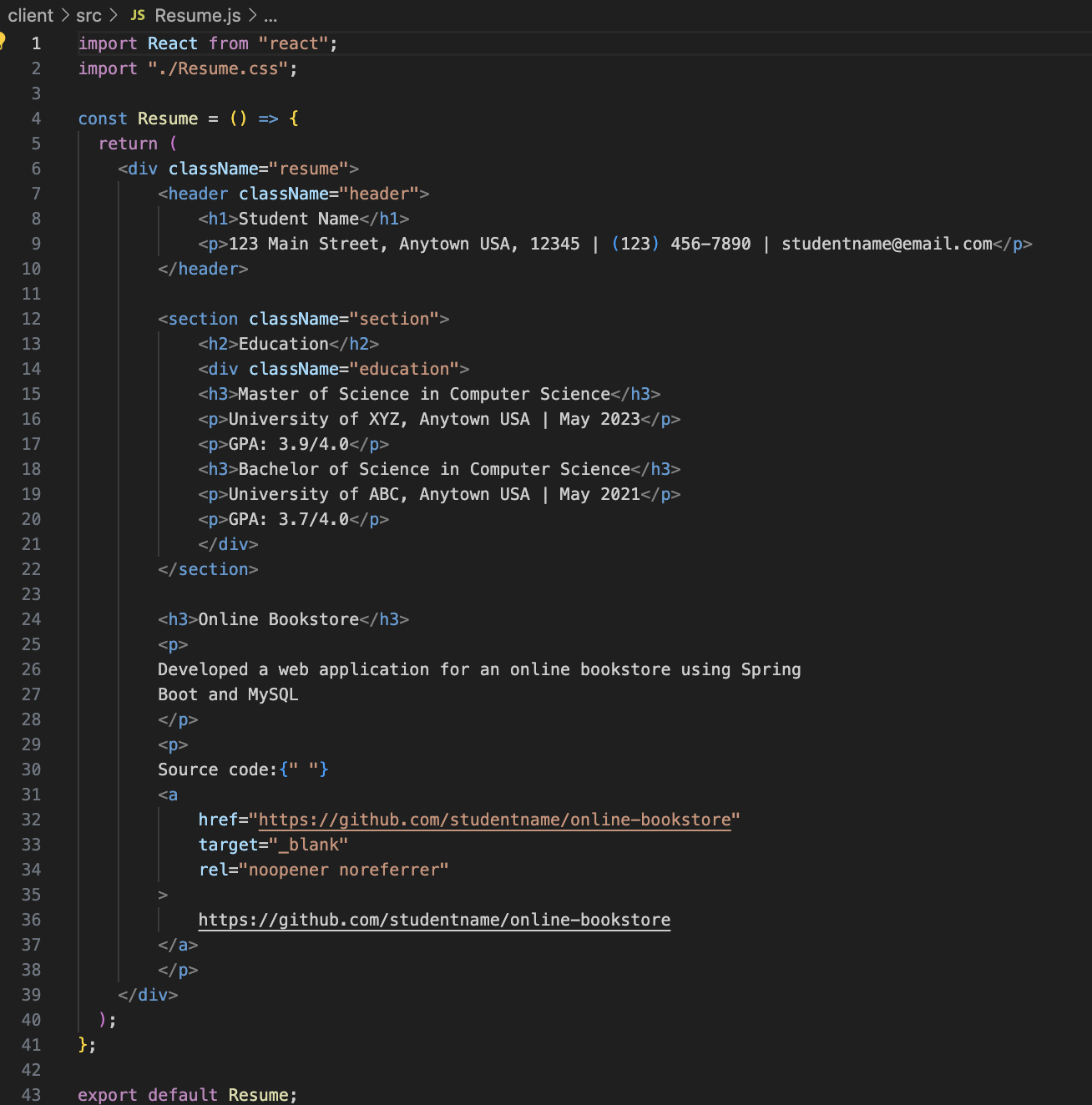
Create a “MyResume” web app that will display your Resume using react. The resume includes the name and contact information of the person, their education, skills, work experience, and projects.

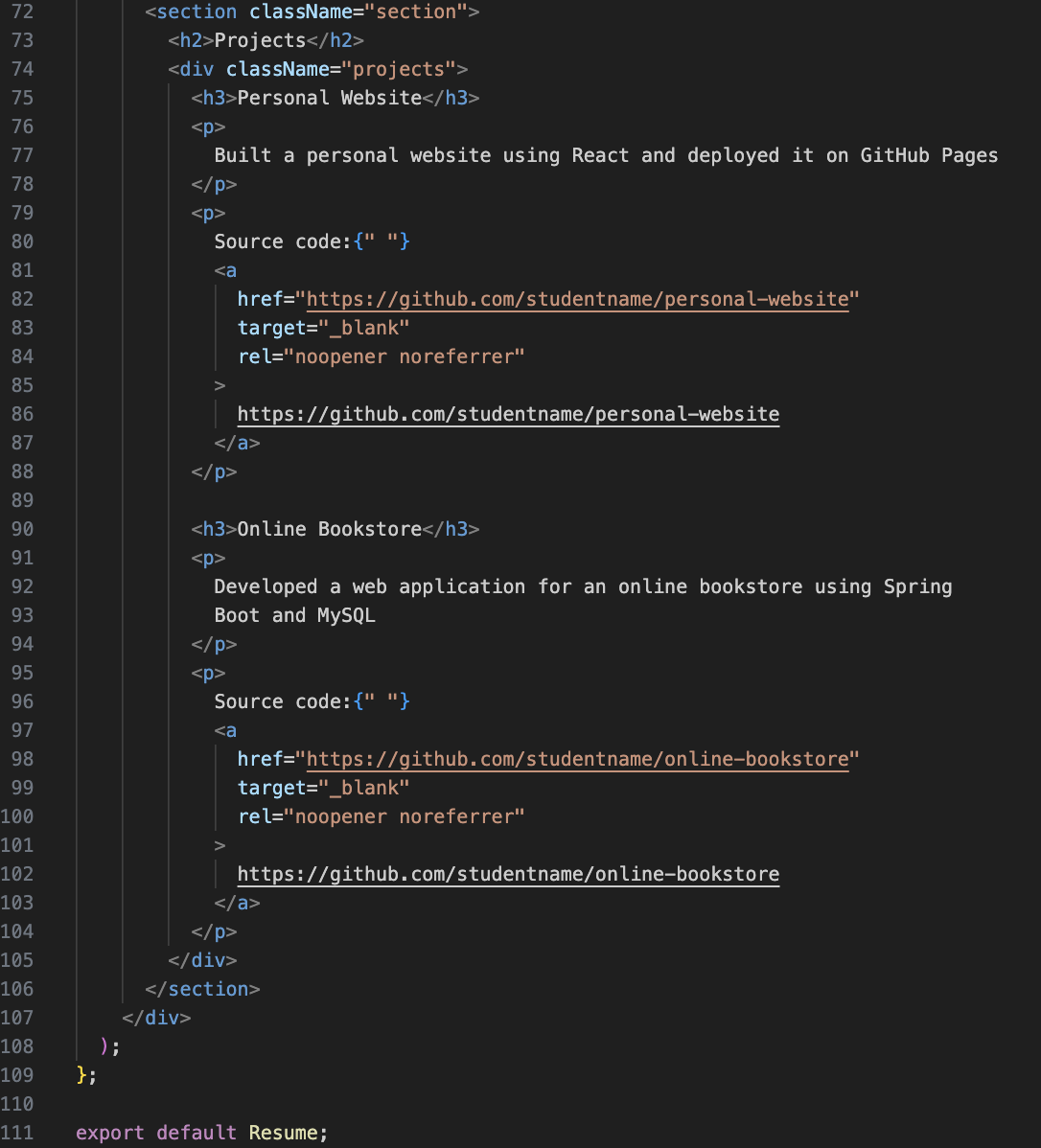
* User Requirements
  + The background color of the resume page should be rgba(#bdfb04)
  + All the text should be defaulting color black.
  + Adjust the margins and fonts to match the sample image shown below.
* System Requirements
  + You need to develop Resume.js (Resume Component) and Resume.css
  + Import these two files to your App.js

In your src folder, create Resume.js and Resume.css.



In Resume.js, you can reuse this sample code and extend it.





In Resume.css, you can reuse this sample code.

|  |
| --- |
| .resume {  font-family: Arial, sans-serif;  margin: 0 auto;  max-width: 800px;  line-height: 1.6;  color: #333;  padding: 20px;  background-color: #f5ffef;  }  .header {  text-align: center;  margin-bottom: 30px;  }  .header h1 {  margin: 0;  font-size: 2.5rem;  color: #1a4d2e;  }  .header p {  font-size: 1rem;  color: #555;  }  .section {  margin-bottom: 20px;  }  .section h2 {  font-size: 1.5rem;  color: #1a4d2e;  margin-bottom: 10px;  }  .education h3,  .experience h3,  .projects h3 {  margin: 0;  font-size: 1.2rem;  color: #2e8b57;  }  .education p,  .projects p {  margin: 5px 0;  }  .skills ul {  list-style-type: none;  margin-left: 20px;  }  .skills li {  margin-bottom: 5px;  list-style-type: none;  }  .experience ul {  list-style-type: none;  margin-left: 20px;  }  .experience li {  margin-bottom: 5px;  }  .projects a {  color: #007acc;  text-decoration: none;  }  .projects a:hover {  text-decoration: underline;  } |

Update App.js like below:

A picture containing text

Description automatically generated

**Sample output:**

Text, letter

Description automatically generated

**Submission**

1. Create a GitHub repository for your programming exercises. The repository name will be “cs628-pe-*your\_first\_name*.”

Graphical user interface, application

Description automatically generated

1

1. Click the Settings menu. Invite your instructor and TA to collaborators.

Graphical user interface, application

Description automatically generated

1

1. Under the repository, create a directory for the programming exercise 1, “PE01-Resume.” For example, the screen below shows the directory created for programming exercise 01.

Graphical user interface, application

Description automatically generated

1. Finish your programming exercise under the PE01 directory.
2. Write a 150-word analysis report to explain how the program works in [README.md](https://www.markdownguide.org/basic-syntax/) in terms of the [input-process-output model](https://press.rebus.community/programmingfundamentals/chapter/input-process-output-model/). The README.md has three level-1 headings – Input, Process, and Output.
3. Please upload the screenshots of your output to your GitHub repository to demonstrate that you have completed the requirements.
4. Submit the link of your GitHub repository to your course shell through your assignment submission.

