Submission Worksheet

Submission Data

Course: IT202-450-M2025
Assignment: IT202 Module 3 HTML, CSS, JS Challenges
Student: Colin R. (ctr26)
Status: Submitted | **Worksheet Progress:** 100%
Potential Grade: 10.00/10.00 (100.00%)
Received Grade: 0.00/10.00 (0.00%)
Started: 6/16/2025 7:42:58 PM
Updated: 6/16/2025 8:04:13 PM
Grading Link: https://learn.ethereallab.app/assignment/v3/IT202-450-M2025/it202-module-3-html-css-js-challenges/grading/ctr26
View Link: https://learn.ethereallab.app/assignment/v3/IT202-450-M2025/it202-module-3-html-css-js-challenges/view/ctr26

Instructions

- Overview Link: https://youtu.be/Dyl6dg1Xybo

1. Ensure you read all instructions and objectives before starting.
2. Create a new branch from `dev` called `M3-Homework`
    1. `git checkout dev` (ensure proper starting branch)
    2. `git pull origin dev` (ensure history is up to date)
    3. `git checkout -b M3-Homework` (create and switch to branch)
3. Copy the template code from here: GitHub Repository - M3 Homework
    - It includes Challenges 1-3, `util.js`, and `styles.css`. Put all into an `M3` folder or similar inside your `public_html`
    - Immediately record to history
        - `git add public_html`
        - `git commit -m "adding M3 HW baseline files"`
        - `git push origin M3-Homework`
        - Create a Pull Request from `M3-Homework` to `dev` and keep it open
4. Fill out the below worksheet
    - Each Problem requires the following as you work
        - Ensure there's a comment with your UCID, date, and brief summary of how the problem was solved
        - Update ucid in header tag
        - Code solution (add/commit periodically as needed) (style and/or script tags)
5. Once finished, click "Submit and Export"
6. Locally add the generated PDF to a folder of your choosing inside your repository folder and move it to Github
    1. `git add .`
    2. `git commit -m "adding PDF"`
    3. `git push origin M3-Homework`
    4. On Github merge the pull request from `M3-Homework` to `dev`
    5. On Github create a pull request from `dev` to `prod` and immediately merge. (This will trigger the `prod`

deploy to make the heroku prod links work)

7. Upload the same PDF to Canvas
8. Sync Local
    1. `git checkout dev`
    2. `git pull origin dev`

Section #1: (3 pts.) Challenge 1 - Fixed Header, Content, Footer

Progress: 100%

☰ Task #1 (3 pts.) - Edit the `style` and `script` tags to solve the challenge requirements

Progress: 100%

Details:

- Only make edits where noted via provided comments
- Update your ucid in the header tag
- #1 The header and footer should remain FIXED in place (top and bottom of page respectively)
- #2 The content area should SCROLL independently (nothing should be pushed off screen and the browser WINDOW scrollbar shouldn't appear)
- #3 The entire page should always take up the full viewport height
- #4 The borders around header,main,footer should remain intact and visible, this will help show that the challenges were solved correctly
- Add code to solve the problem (add/commit as needed)

🖼 Part 1:

Progress: 100%

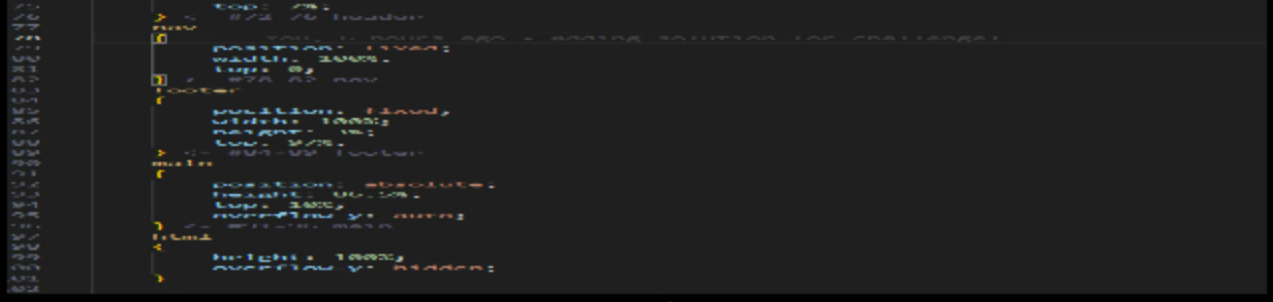
Details:
Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program (visit the proper file on Heroku dev after a manual deploy) 1. Ensure url is visible in the browser's address bar

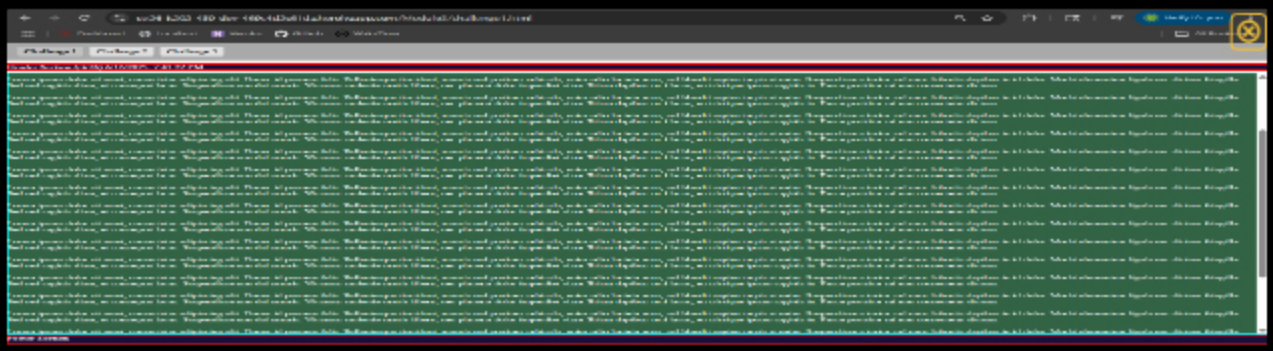
```
61    /*
62    ctr26 06-16-2025
63
64    Step 1: Create new selectors for each of the elements I want to change
65    Step 2: Change their position value to being fixed
66    Step 3: Adjust their top values to display in the right position
67    Step 4: Change their width to take up the entire screen
68    Step 5: Change the height of the scroll bar to not overlap the other elements
69    */
```

C1 outline

```
    header {
        position: fixed;
        width: 100%;
```



C1 code



C1 output

Saved: 6/16/2025 7:47:59 PM

Part 2:

Progress: 100%

Details:

- Direct link to the file in the homework-related branch from GitHub (should end in `.html`)
- Direct link to the file on Heroku Prod (Just grab the base prod url and manually enter the path to the file)

URL #1
https://github.com/ColinRafferty7/ctr26-IT202-450-M3-Homework/public_html/Module3/challenge1.html

URL
https://github.com/ColinRafferty7

URL #2
https://ctr26-it202-450-prod-2ae7f60cad14.herokuapp.com/Module3/challenge1.html

URL
https://ctr26-it202-450-prod-2ae7

Saved: 6/16/2025 7:47:59 PM

Part 3:

Progress: 100%

Details:
Briefly explain `how` the code solves the challenge(s) (note: this isn't the same as `what` the code does)

Your Response:

The code solves the problem by changing the position value of each of the neccesary elements to fixed. Then it moves each one to its desired location on the screen. It also changes the content

element to have its own scroll bar if the text overflows the screen.

Saved: 6/16/2025 7:47:59 PM

Section #2: (3 pts.) Challenge 2 - Header, Content, And Sidebars

Progress: 100%

Task #1 (3 pts.) - Edit the `style` and `script` tags to solve the challenge requirements

Progress: 100%

Details:

- Only make edits where noted via provided comments
- Update your ucid in the header tag
- Using CSS, adjust the layout per the following
  - #1: Header is at the top
  - #2: Content takes up the rest of the height
  - #3: Both sidebars docked to the respective side and take up 15% width; Content area should utilize the remaining width
- Using JavaScript complete the following
  - #4: Attach the appropriate event listener to the buttons
  - #5: Individual toggle the respective panel between collapsed and uncollapsed (Don't lose the button in the process)
  - #6: The content should adjust based on the status of the respective sidebars (i.e., take up more left, right, or both space)
- Add code to solve the problem (add/commit as needed)

Part 1:

Progress: 100%

Details:
Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program (visit the proper file on Heroku dev after a manual deploy) 1. Ensure url is visible in the browser's address bar

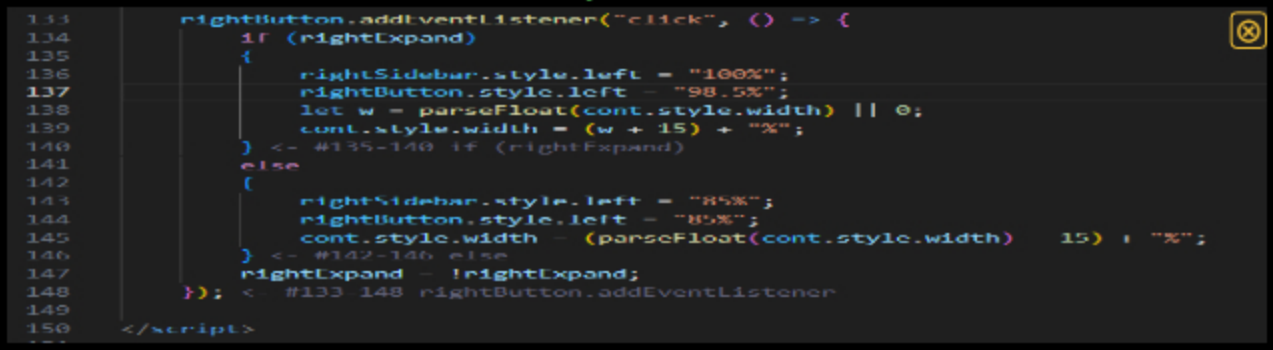
```
/*
ctr26 06-16-2025

Step 1: Change the position of the header to be fixed and adjust the width
Step 2: Adjust the margins for the content to be at the correct height
```

```
Step 3: Change width and height of sidebars to match their respective side
Step 4: Change width of content to fit between sidebars
Step 5: Use the query selector function to add event listeners to each button individually
Step 6: Create variables for each element that will change upon closing a panel
Step 7: Inside the event listener for each button, adjust the values of the respective elements
*/
```

C2 outline

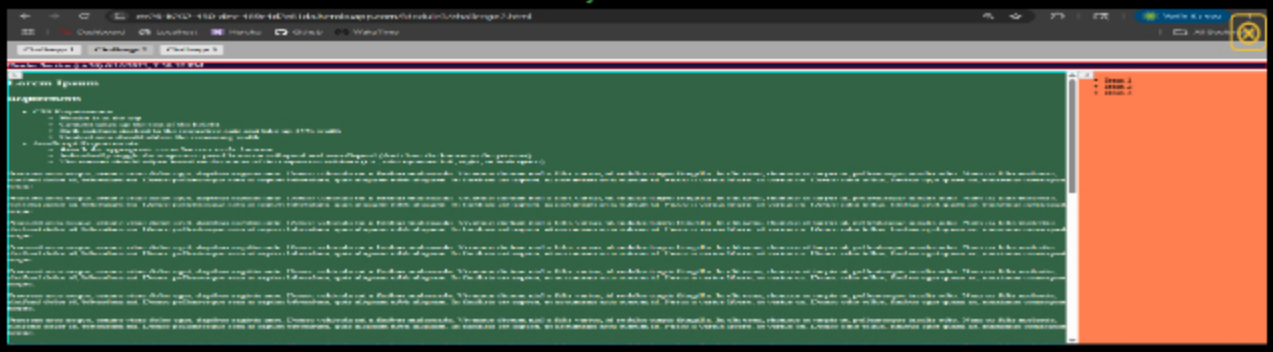
C2 css code

C2 js code 1/2



```
    rightButton.addEventListener("click", () => {
        if (rightExpand)
        {
            rightSidebar.style.left = "100%";
            rightButton.style.left = "98.5%";
            let w = parseFloat(cont.style.width) || 0;
            cont.style.width = (w + 15) + "%";
        } <- #135-140 if (rightExpand)
        else
        {
            rightSidebar.style.left = "85%";
            rightButton.style.left = "83%";
            cont.style.width = (parseFloat(cont.style.width) - 15) + "%";
        } <- #142-146 else
        rightExpand = !rightExpand;
    }); <- #133 148 rightButton.addEventListener

</script>
```

C2 js code 2/2



C2 output

Saved: 6/16/2025 7:53:44 PM

Part 2:

Progress: 100%

Details:

- Direct link to the file in the homework-related branch from GitHub (should end in `.html`)
- Direct link to the file on Heroku Prod (Just grab the base prod url and manually enter the

path to the file)

URL #1
https://github.com/ColinRafferty7/ctr26-IT202450M3-Homework/public_html/Module3/challenge2.html

URL
https://github.com/ColinRafferty7

URL #2
https://ctr26-it202-450-prod-2ae7f60cad14.herokuapp.com/Module3/challenge2.html

URL
https://ctr26-it202-450-prod-2ae7

Saved: 6/16/2025 7:53:44 PM

Part 3:

Progress: 100%

Details:
Briefly explain `how` the code solves the challenge(s) (note: this isn't the same as `what` the code does)

Your Response:

The code solves the problem by giving each button its own function when pressed. They move their respective panels off screen and dynamically adust the size of the content element.

Saved: 6/16/2025 7:53:44 PM

Section #3: (3 pts.) Challenge 3 - Carousel Layout With Swiping

Progress: 50%

Task #1 (3 pts.) - Edit the `style` and `script` tags to solve the challenge requirements

Progress: 100%

Details:

- Only make edits where noted via provided comments
- Update your ucid in the header tag
- Using CSS, adjust the layout per the following
  - #1 Header should be at the top
  - #2 Carousel should take up full width
  - #3 Buttons should be centered
  - #4 Carousel panels should fill the full height of the carousel
  - #5 Carousel content should be centered (vertical and horizontal)
- Using JavaScript complete the following
  - #4 Attach appropriate event listeners to each button

- #4 Attach appropriate event listeners to each button
- #5 Cycle through each panel, showing only 1 at a time
- #6 Ensure that the panels loop when reaching the last or first one

- Add code to solve the problem (add/commit as needed)

Part 1:

Progress: 100%

Details:

Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program (visit the proper file on Heroku dev after a manual deploy) 1. Ensure url is visible in the browser's address bar

```
<style>
    /* You're free to make whatever edits you need here, just don't override anything in the styles.css file */

    /*
    clr26 06-16-2025

    Step 1: Follow same steps as previous challenges for header
    Step 2: Change position values of carousel to be center
    Step 3: Do the same thing for the buttons
    */

    .carousel
    {
        text-align: center;        You, 1 hour ago • Adding css solution for challenge1 ...
    }
    nav
    {
        text-align: center;
    }
</style>
```

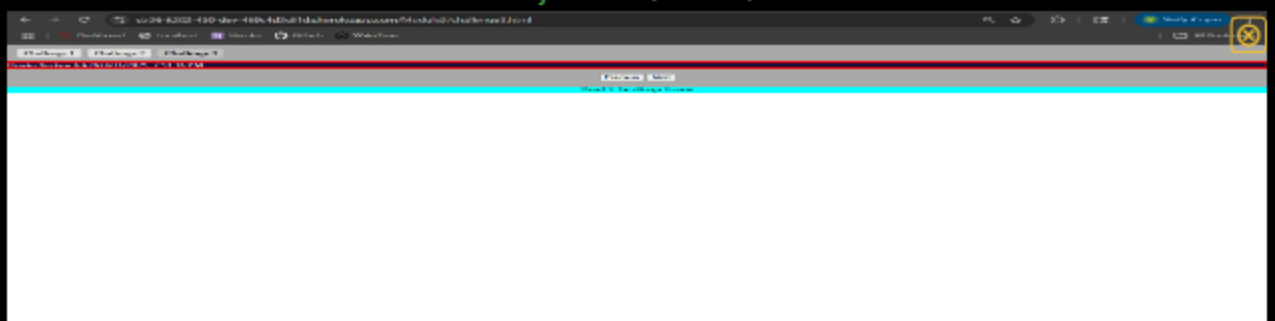
C3 css outline/code

```
    <script>
        /* Solve the button logic here by finding and attaching an appropriate event listener and logic */

        /*
        clr26 06-16-2025

        Step 1: Add event listeners to both buttons
        Step 2: Use indexing for each panel item to cycle them in order
        Step 3: When the index goes above 4 or below 0, switch to the opposite end
        */

        const buttons = document.querySelectorAll("nav button");
        const prevButton = buttons[0];
        const nextButton = buttons[1];

        const panels = document.querySelectorAll(".carousel .panel");

        let currentPanel = 0;

        panels.forEach((panel, i) => {
            if (i != currentPanel)
            {
                panel.style.display = "none";
            }
        });
```

C3 js outline/code 1/2

```
        prevButton.addEventListener("click", () => {
            panels[currentPanel].style.display = "none"
            currentPanel = currentPanel - 1;
            if (currentPanel < 0)
            {
                currentPanel = 4;
            }
            panels[currentPanel].style.display = "block";
        });

        nextButton.addEventListener("click", () => {
            panels[currentPanel].style.display = "none"
            currentPanel = currentPanel + 1;
            if (currentPanel > 4)
            {
                currentPanel = 0;
            }
            panels[currentPanel].style.display = "block";
        });

    </script>
```

C3 js outline/code 2/2



C3 output

Saved: 6/16/2025 7:57:56 PM

Part 2:

Progress: 100%

Details:

- Direct link to the file in the homework-related branch from GitHub (should end in `.html`)
- Direct link to the file on Heroku Prod (Just grab the base prod url and manually enter the path to the file)

URL #1
https://github.com/ColinRafferty7/ctr26-IT202-450/M3-Homework/public_html/Module3/challenge3.html

URL
https://github.com/ColinRafferty7

URL #2
https://ctr26-it202-450-prod-2ae7f60cad14.herokuapp.com/Module3/challenge3.html

URL
https://ctr26-it202-450-prod-2ae7

Saved: 6/16/2025 7:57:56 PM

Part 3:

Progress: 100%

Details:
Briefly explain `how` the code solves the challenge(s) (note: this isn't the same as `what` the code does)

Your Response:

The code works by convert the container into an array and giving each panel its own index. Then, by giving the buttons functions, I can change what the current index is and update the screen accordingly.

Saved: 6/16/2025 7:57:56 PM

Task #2 (+ 1.01 pts.) - Extra Credit - Challenge 7

Progress: 0%

Details:

- Allow mouse swipe on the carousel to cycle through the panels, similar to how the buttons would work

Part 1:

Progress: 0%

Details:

1. Snippet of relevant code showing solution (with ucid/date comment)

Missing Caption

Saved: 6/13/2025 12:57:34 AM

Part 2:

Progress: 0%

Details:
Briefly explain `how` the code solves the challenge(s) (note: this isn't the same as `what` the code does)

Your Response:

Missing Response

Saved: 6/13/2025 12:57:34 AM

Section #4: (1 pt.) Misc

Progress: 100%

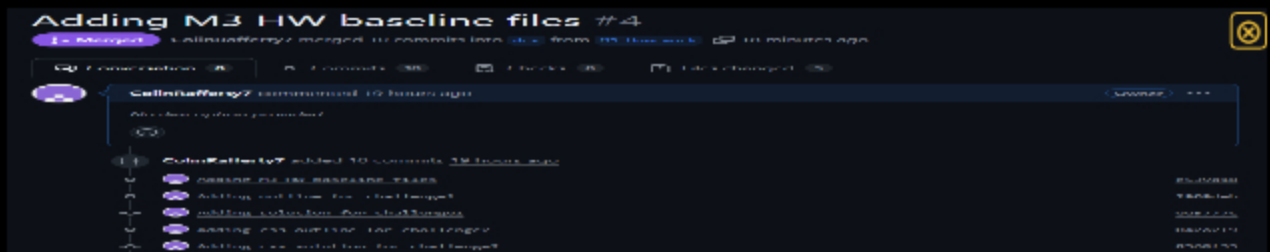
Task #1 (0.33 pts.) - Github Details

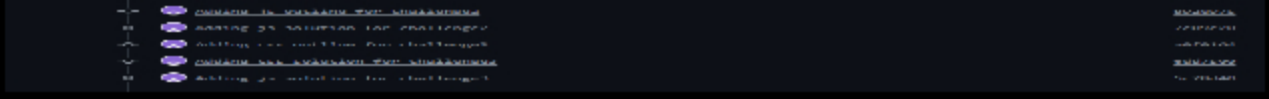
Progress: 100%

Part 1:

Progress: 100%

Details:
From the Commits tab of the Pull Request screenshot the commit history





Pull request

Saved: 6/16/2025 7:59:11 PM

Part 2:

Progress: 100%

Details:
Include the link to the Pull Request (should end in `/pull/#`)

URL #1
https://github.com/ColinRafferty7/ctr26-IT202-450/

URL
https://github.com/ColinRafferty7

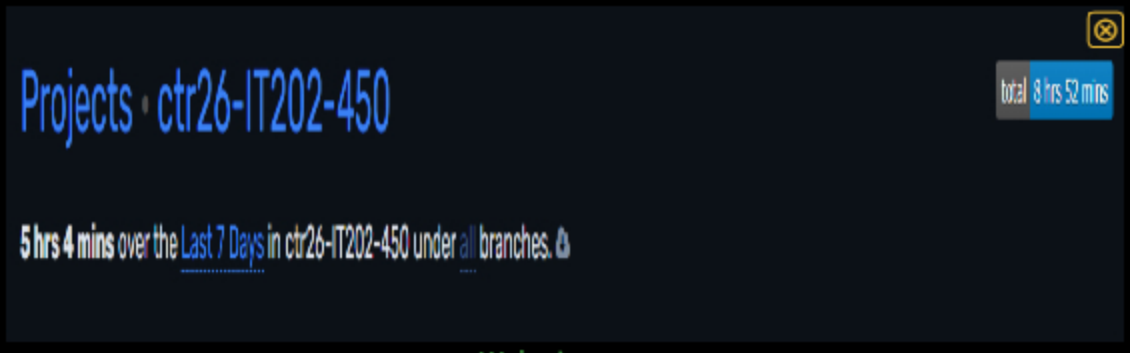
Saved: 6/16/2025 7:59:11 PM

Task #2 (0.33 pts.) - WakaTime - Activity

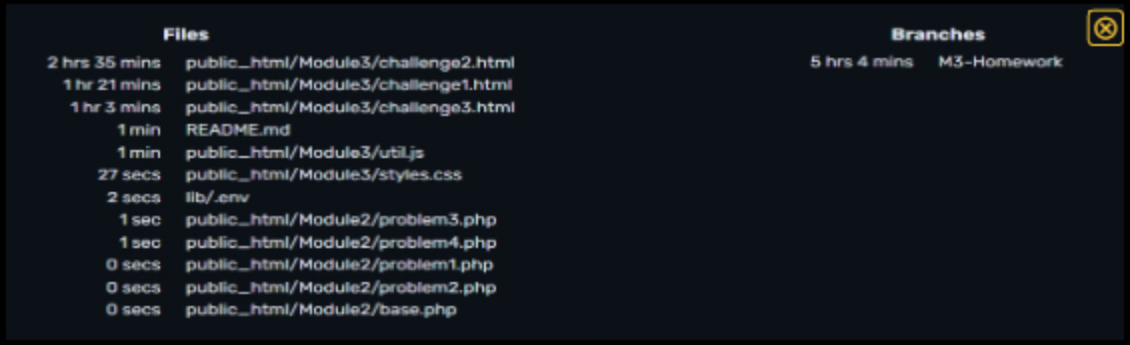
Progress: 100%

Details:

- Visit the WakaTime.com Dashboard
- Click `Projects` and find your repository
- Capture the overall time at the top that includes the repository name
- Capture the individual time at the bottom that includes the file time
- Note: The duration isn't relevant for the grade and the visual graphs aren't necessary



Wakatime top



Wakatime bottom

Saved: 6/16/2025 8:00:34 PM

Task #3 (0.33 pts.) - Reflection

Progress: 100%

Task #1 (0.33 pts.) - What did you learn?

Progress: 100%

Details:
Briefly answer the question (at least a few decent sentences)

Your Response:

In this assignment, I learned how to use css and javascript to edit webpages. This is something that I have never done, but have been looking forward to learning for a while. I knew the very basics of html, but it was a nice experience learning all of the other parts from scratch.

Saved: 6/16/2025 8:01:56 PM

Task #2 (0.33 pts.) - What was the easiest part of the assignment?

Progress: 100%

Details:
Briefly answer the question (at least a few decent sentences)

Your Response:

The easiest part of this assignment was the css elements. The language seems very simple and I like that everything has its own variable and states. It makes it a lot less manual than many other languages I have worked with.

Saved: 6/16/2025 8:02:58 PM

Task #3 (0.33 pts.) - What was the hardest part of the assignment?

Progress: 100%

Details:
Briefly answer the question (at least a few decent sentences)

Your Response:

The hardest part of this assignment was working with the javascript. I have experience in many similar languages, but never in this context. Getting everything I wanted to work was a struggle and I spent a lot of time researching the different solutions.

Saved: 6/16/2025 8:04:13 PM