Due on Friday, 5/3/2019 @ 11:59PM

EECS 130: Lab 5

Practice with JavaScript conditional statements

Instructions

Please download the lab05 files and save them to your repo folder. Then, open the entire lab05 folder in Atom.

Part 1: Color Mixer [REQUIRED]

1. Open **`01-color-mixer/index.html`** in Atom, and note the HTML tags. Then, open it in your web browser and take a look at it. Try toggling the red and yellow switches to see what happens.

2. Open **`01-color-mixer/js/index.js`** in Atom, and examine the updateColor() function. Note that the function is attached to the onchange event listener for each menu.

3. Complete the tasks below to finish the color mixer exercise. You will need to use the "strict comparison" operator (===) as well as the and operator (&&) to complete this assignment:

    a. If red is turned on, make the background red.
    b. If yellow is turned on, make the background yellow.
    c. If blue is turned on, make the background blue.
    d. If red and yellow are both turned on, make the background orange.
    e. If red and blue are turned on, make the background purple.
    f. If yellow and blue are turned on, make the background green.
    g. If everything is turned on, then make the background black.

Part 2: Rock Paper Scissors [REQUIRED]

Create a game called "Rock, Paper, Scissors." The user will pick one of the three options, and the computer will randomly pick one of the 3 options. Here are the rules:

- If the user picked "Rock" and the computer picked "Paper": Computer Wins
- If the user picked "Rock" and the computer picked "Scissors": User Wins
- If the user picked "Paper" and the computer picked "Scissors": Computer Wins
- If the user picked "Paper" and the computer picked "Rock": User Wins
- If the user picked "Scissors" and the computer picked "Rock": Computer Wins

- If the user picked “Scissors” and the computer picked “Paper”: User Wins
- Otherwise: Tie

I have already started the code for you, located in the **`02-rock-paper-scissors`** folder. In the js/index.js file, there is already a function, getComputerChoice(), that simulates the computer “picking” one of the 3 options. In addition, there is already functionality in the playGame() function that outputs the computer and user’s respective choices to the screen.

Your job: Using a series of if / else statements, implement the remaining logic of the playGame() function, to figure out if the user or the computer won the match.

Optional (if you’re interested)

Use icons and images, instead of text to represent the game. I’ve included a reference to a popular external stylesheet called “FontAwesome” that makes it easy to incorporate icons. The icon list is here: https://fontawesome.com/icons?d=gallery&m=free. I have also created 3 variables that store some code to output an icon to the DOM.

Part 3: Color Mixer Advanced [OPTIONAL]

If you have time, try to complete the **`03-color-mixer-optional`** color mixer (where the r, g, and b values are used to dynamically update the background.

Turn in your files via GitHub

1. Ensure that your lab05 folder is inside of your repo folder.

2. Check the status of your repo:

   ```
   $ git status
   ```

   It should say that your lab04 files are "untracked"

3. Add all of your lab05 files to the list of files that are tracked by the repo:

   ```
   $ git add lab05
   ```

4. Check the status of your repo again:

   ```
   $ git status
   ```

   It should now say that you lab04 files are being tracked.

5. Now, commit your new lab05 files to the repo:

   ```
   $ git commit -m "Adding my lab05 to the repo"
   ```

6. Finally, "push" your files to GitHub:

   ```
   $ git push origin master
   ```

7. Paste a link to your GitHub repo and to your GitHub pages