Week 4 in class work

Tell me about what you worked on during class during week 4 (html, css and svg).

You can include screenshots of what you created.

Include some code.

Tell me about how things are going- where do you want to learn more, practice more?

Monday activity

We are going to make a UI for a smart fridge in-class. Today, we are just practicing the following with a partner (write your own code to submit for in-class work, but work together to figure out how):

1. Create an html project. Starter template here: basic-template.zip

2. Open the html file in your favorite editor. I like sublime text: https://www.sublimetext.com/.

3. Edit the html file to add text, buttons, controls, svg elements to create some of the functionalities of a smart fridge listed below. These will not be interactive, because we aren't writing javascript code today. These won't be laid out elegantly, because we haven't covered tools that will help with layout. Initially, they may not look very nice or interesting, but you can play around with style settings.

4. As you go, view the result in your browser. You can just double click on the html file to view it in the browser. I suggest Chrome.

5. Inspect your page using the web developer tools in Chrome (View >> Developer >> Developer Tools). Look at the elements and where they are onscreen, and the layout and sizing decisions computed by the layout engine in the Chrome browser.

6. Test out editing the html in the developer console to see immediate changes.

What are we implementing?

Starting assumptions, for a basic UI:

1. This smart fridge has interface that enables the user to see the temperature for different compartments

2. The user will be able to set the overall temperature for:

- the fridge
- the freezer
- the crisper compartments

3. The user will be able to see how full each portion of the fridge is

4. The smart fridge tracks and displays:

- the number of times the fridge someone has opened the fridge that day
- the average daily number of times the fridge is opened
- the median length of time the fridge has been left open that day
- the median length of the time the fridge has been left open over the entire time someone owned the fridge

5. The smart fridge communicates how much water has been dispensed that day, vs the average daily water dispensed

6. The smart fridge has a photo viewer

7. The smart fridge has a place to enter a reminder note

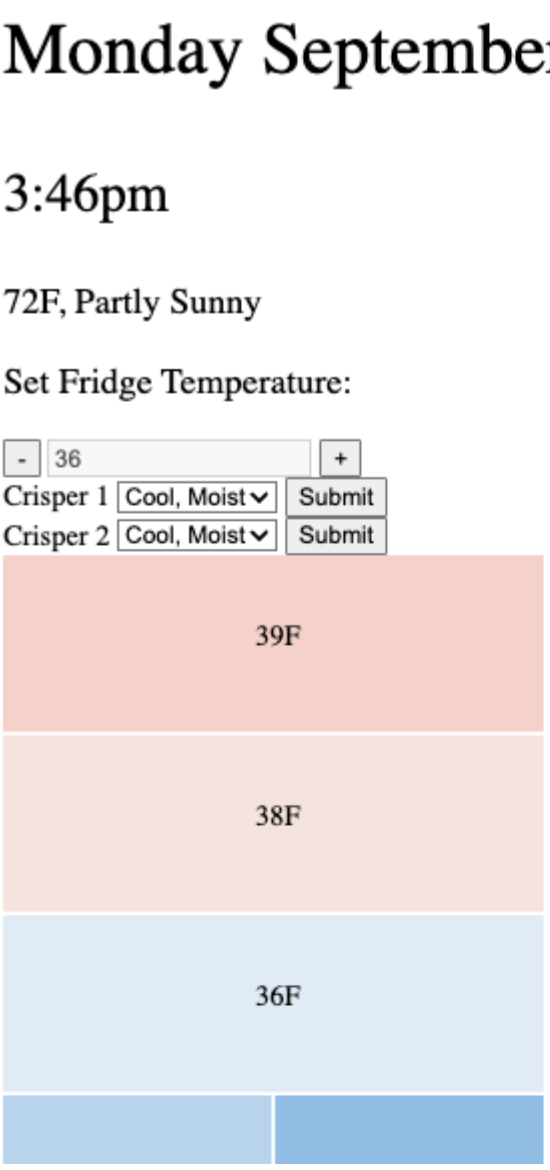
8. The smart fridge lets you set a timer

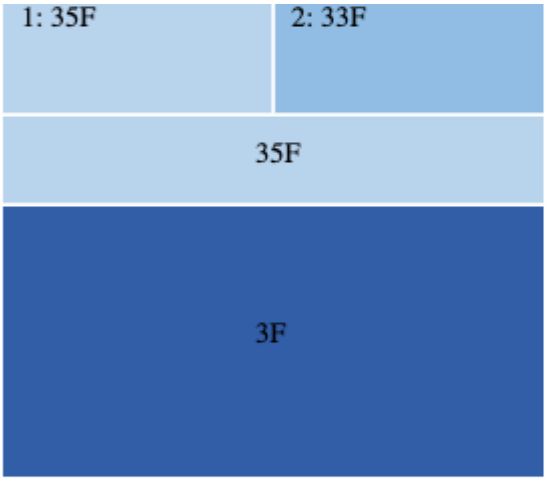
9. Come up with your own ideas ....

Here's what I created (see below). The fridge picture I did using svg rectangles and text elements, crudely positioned manually. The rest is html elements. The only style rules I set were for text size. I downloaded some images to include, and put them in a folder called 'img'.

It doesn't look great. It is not created after needs finding, or with lots of design prototyping. But it will get you to kick the tires for developing a simple webpage with just html, css and svg. Next time, we will work on layout and get into javascript.

Try it yourself, with a partner. Your code will be your own, but see if you can work together to try making a smart fridge UI like the one below. It does NOT have to be identical. Feel free to play around and see how things work.

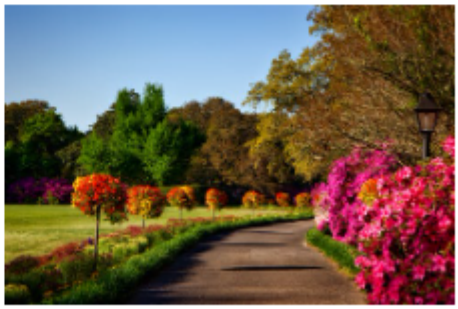




Fridge opened 23 times today (typical 20)

You open your fridge for 38 seconds on average today (typical 14)

Family photos:



Create note:

Wednesday activity

Let's continue with the smart fridge UI we played with last time. As a reminder, this is what we did, but you can take the folder from last time and continue working on it.

1. Open the html file in your favorite editor. I like sublime text: https://www.sublimetext.com/.

AND open the css file in your favorite editor.

2. Edit the html file to add MORE text, buttons, images, controls to create some of the functionalities of a smart fridge listed below.

3. As you go, view the result in your browser. You can just double click on the html file to view it in the browser. I suggest Chrome.

4. Inspect your page using the web developer tools in Chrome (View >> Developer >> Developer Tools). Look at the elements and where they are onscreen, and the layout and sizing decisions computed by the layout engine in the Chrome browser.

5. Test out editing the html in the developer console to see immediate changes.

NOW:

Try

A. Grouping elements into div's. How does it change your layout?

B. Use
 to separate elements.

C. Give elements class names, in your html code. And write style rules for those classes.

- Set their color
- Choose a new font and font size
- Add rounded corners if it is a div
- Try some things!

Below is what I did last time, can you make it look better?

Friday:

1. Add some interactive changes to the style of your elements when hovering over them, using css pseudo classes. (selector:hover).

2. Modify the layout of your elements using either css alone, css flexbox or css grid.

a. If you have time, rather than implementing the colored fridge temperature indicators shown below using svg (Which was what I did), NOW make it using a divs, styling those divs, and positioning with css, css flexbox or css grid. You can also add more compartments and sub-compartments to make it more interesting.