Unit 3 Learning Diary

Learning CSS

I have started to style my pages with CSS. I began with the home page. I decided to get rid of the "main_header" div in my original HTML pages, because I thought it didn't fit with the logo after I applied some basic CSS. The logo looked great on its own, so I let it stand on its own.

I had some issues with center-ing all the objects on the page, including the navigation bar, the logo, and the other sections on the page. I fixed this by removing the logo from the division that contained the navigation bar and using CSS to center it on its own. I used this to figure out how to center images.

For the navigation bar functionality, I used this resource.

I then moved on to the Products page. I ended up removing the gear logo I had in my original HTML pages because I thought it didn't fit. I wanted something different. So I edited my own image where I overlaid some text on top of a cutting board. I then centered this image and proceeded to the rest of the elements.

I wanted each type of pasta to have a little block of its own, which will have additional functionality when I get to javascript. I used the "display: inline-block" property to format the pasta blocks, along with basic background formatting and borders.

On the Products page, I wanted the logo and the navigation bar to be laid on top of an opaque background. Putting these elements in a parent div and then making that div opaque didn't work. Doing this made EVERYTHING inside the div opaque, and I didn't want that at all. So I decided to create a separate div with no elements inside, make its position absolute (so it can lay on top of other elements), give it a white background, then make that div opaque. To make the div span the width of the page, I used this source as help. However, this was really messy, so I switched to a plain color background for a while, until I accidentally discovered this feature

On the Products page, I tried to increase the space between the logo and the top of the page, but it wasn't working even though I increased the "margin-top" CSS property of the image with the id "board", which is the logo in question. So I put the logo into the div that contained the navigation bar, and tried to increase the margin between the navigation_bar div and the top of the page. This ended up working.

To wrap up this page, I tidied up the pictures of pasta and even replaced some of the original ones with new ones, to improve the aesthetics of the page.

Now, I'm moving on to the Pictures & Videos page of my site. I've had a change of heart about how I want this page to look. In addition to a different look, I want to collapse the Development page and the Pictures/Videos page. It's not productive to have these as separate pages.

To start this page out, I took the big paragraph of text from my "Production" page and pasted it into the Pics/Vids page, with some basic CSS formatting (borders, nice font, gray background.) Then I moved on to the main part of the page – the pictures and text outlining the production process.

Given my new plans for the layout of the page, some change in code was required. For each step of development, I wanted a long horizontal div centred on the page, with three main elements: a big number on the left (I got the idea from here, a picture on the right, and some text in the middle.

I encountered some real problems here. I began by creating a container div and three divs inside of it, and giving each inside div relative position attributes (e.g., left: 10%). It didn't work. Every time I tried to put something in the div, whether it was a picture or some text, it would move further down on the page for no reason. I couldn't fix this problem; I tried giving the divs absolute positions or floating them left/right, but it didn't work. So I went on the web for help.

After some Google searches, I stumbled across this site, which gave instructions for something called "Flexbox". It was supposed to give you greater control over the position of your elements, so I gave it a read. I tested it on a blank HTML page with a couple divs and it worked great; I could position divs horizontally/vertically center them, and space them with ease. So I applied this code to my website immediately. I also referenced this site for any additional tips on Flexbox.

After this was finished, I decided to apply the flexbox style to the navigation bar as well, and immediately formatted the nav bars in all the pages.

I then moved on to the background of the Pics/Vids page. Since the focus of this page is production, and gears symbolize machines and creation, I decided to have a background of rotating pasta. It's supposed to make you think of gears, but also what Turri's is all about – pasta.

I gathered some images of the web (pictures of pasta and a nice blue background). I overlaid the images onto the background and used Final Cut Pro to rotate the pasta. I exported the file, went to ezgif.com and converted to a gif. I then used the image as the background for my page, and it looks great.

Finally, I went about designing the logo for the page. Right now I don't have a good idea of a logo for the Pictures/Videos (now the Development page on my site), so I created a simple one as a placeholder until I get a better idea.

Now, I moved on to the Sanitation page. When designing this page, I really developed the final structure and form of my headers. The header used Flexbox to make it easy to lay out. The hierarchy of the elements is as follows: (1) **header-container**, which contains all the items within the header. Items (class **main-item**) are stacked in a column. First up, we have the logo, then we have the navigation bar directly beneath it. The navigation bar is a Flexbox container of its own, starting with the

nav_container class, which is a Flexbox container with the items assorted in a row. The link buttons have simple styling, but I haven't yet used the **nav-item** class in the CSS files for any formatting – I still target the **< li >** and **< a >** items to format the links. I might change this later.

The Sanitation page has a simple layout. We have 3 styles of container. The biggest one (**main-big-container**) only has one purpose, and that is to center everything in the main body of the page. It contains the next biggest container (**main-medium-container**), which is another Flexbox container with items assorted in a column. The items are the smallest container (**main-container**), which contain two things: a picture, and some text to the right. To produce some variability on the page, the pictures go from left to right as you scroll down the page. I also added a border to the top of each for some visual separation.

After finishing the Sanitation page, I decided to view my pages on a bigger computer (until now, I had been viewing my website on my laptop). I went to a full-screen iMac and I noticed some things which needed fixing. First, the webpages usually spans the whole width of the screen, which is problematic because it looks ugly. Also, some of the formatting which looks fine on my laptop breaks on a bigger screen. I have a solution for all the webpages in the future: to have everything stored in a single container, centered on the page and given a max-width property to stop it from getting any wider and breaking. Also, to make the code more maintainable and adjustable, I planned to re-code stuff on other pages to Flexbox, and to stop relying on CSS properties like **display: inline-box** to lay out my material.

Another long-term goal is to make the sites mobile-friendly as well as big-screen-friendly. I'm already able to shrink the pages to minimum width on my screen and they look mostly fine, but some (like the Sanitation page) break. I need to fix this.

Sanitizer icon from here, I also put in the appropriate buttons as requested by the creators of the icon.

I'm going to start designing the Contact page now. I was unhappy with my initial Contact page design (too simple) and went to this site to get some ideas. After looking around, I came to the contact form for mediaproper.com and it looked nice, so I decided to base my contact forms around those.

Using w3school's tutorial on forms helped me a lot with this. Once I finished formatting the forms, I put some contact information above them and that was pretty much it for the page, except for the header. Ever since I finalized the header design, it's been very easy to design headers for a new page – all I need to do is copy & paste the html and css, then change the background color and icon, then I'm all set.

I've almost completed the design for all the pages. However, I still need to make the site viewable for all devices. I could make the pages dynamic and make the text shrink as the pages shrink, but it's too tricky for me. I've found an alternative in **media queries**. From a brief read, I think you just need to create slightly different CSS sheets for screens of different sizes, and instruct the HTML to use a specific sheet if the screen is a certain size.

Before I go start creating the queries, I want to host my website so I can access it from a mobile device. Turns out, I actually don't need to host my website to access my code from a mobile device. From this helpful page on Stack Overflow, I learned how to set up a Python server on my computer from the terminal and then access my files from mobile. I was able to view my site and gather what needed to change on it.

It's been a couple days since I tried to effectively implement media queries into my site. I failed – I can't get it to work correctly, and I only have about a month left to finish my site, so I'm not going to continue messing around with it. Instead, I fixed any lingering bugs on my pages. I am going to tidy up the code and fully comment/document it here in my diary, and then submit it on the Landing.

I will not comment my HTML code. I've read articles by other coders and many of them say you shouldn't comment HTML because it needlessly clutters code that should be easy to read anyway. Source 1. Source 2.

However, since I'm creating a journal and should be explaining my thought processes as I create this site, I don't feel good not talking at all about the structure of my HTML code, so I'm going to talk a bit about it here.

CODE EXPLANATION

1. General Structure

The following comments on my pages might not line up with what I wrote above, since I wrote this diary as I developed my site, and things changed along the way.

Every one of my pages has some things in common. First, the "**master**" tag which holds everything inside the body of the page. This tag has a maximum width of 1440 pixels so that, if my pages are viewed on a larger screen, the boxes don't stretch the width of the page which looks really ugly. Instead, everything is contained inside a smaller box which makes it look tidier.

Second, the header. Every header on all my pages has the same basic structure, only differing in the background color and icon. First, we have the "**header-container**" which contains everything inside the header. This is a flex container which centers all the items inside it horizontally. The items inside are called "**header-items**", and there are two of them – one for the icon on on top and another for the navigation bar. These are also flex containers which center their content horizontally. The second one contains an unordered list with the id "**nav_container**", which in turn contains all the links to the other pages on the site.

On a couple pages, I have a block of text that does some explaining below the header. I give each of these blocks of text the id "**summary**", and they all have the same look – a simple gray box with some text in the center.

Anytime I used an icon from flaticon (this is where I got some of my header icons from), I did as the authors requested and put some code on my site crediting them for their creation.

Finally, on some sites, I have a “**footer**” div at the bottom on the page which serves simply for visual separation.

2. Home Page

On this page, we have three main sections (not including the header), which are all boxes with a question in them – “What do we make?”, “How do we make it?”, and “Who are our customers?”. Each of these boxes is a different color, one for each color on the Italian flag (given the company’s origins). Underneath these question boxes is another box, with an answer to that question. I plan to use Javascript to make these bottom boxes invisible until you click on the question. Each pair of boxes has the tag “**main-container**” and is contained inside a larger “**main-large-container**”, which is a Flexbox that aligns items vertically.

3. Products Page

This page was trickier to make, despite its simpler appearance. We just have a bunch of pasta flip-cards here (16 to be exact). Each card is a product with some cool info soon to be put on the back (for now, I don’t know what to say, but I will think of something eventually). All the cards are inside a “**main-big-container**”, which is a Flexbox that aligns items by row. The cards are set to go to the next row if the current one doesn’t have anymore room, so if you make the web browser narrower, the cards simply move to the next row to make room for each other.

Originally, I was going to use Javascript to animate the cards, but for the animation I had in mind, the best option was to use CSS animations. I used this video and this video to help me get the animations working.

Each flip-card starts out with a class called “**flip3D**”, and there are two divs inside this called “**front**” and “**back**”, whose names are self-explanatory. I treat the “**front**” div as a Flexbox with two items inside, some text and a picture of pasta. The back will merely contain some text.

Right now, these flip-cards turn when you hover over them with the mouse. I have some concerns about this being finicky on mobile devices, so once I start implementing Javascript, I will enable these to turn on click only.

4. Development page

This Page has a slightly larger header, to accommodate for the special background. It's a gif of some rotating pasta, which I created with a movie-editing application. The pasta are rotating to resemble gears, which represent production, which is what the page is all about.

Below the header is a large block of text, which briefly explains the production process. For those to want to see production in action, below this block of text are the steps employees go through to prepare the pasta. Each step is a box with three elements: a number, some text, and a photo/gif.

To start out with, we have the "**mega-container**". Yes, I know I am inconsistent with the container names, but I just created them as I went along. A little clumsy, I know. This container merely serves to horizontally center the inside container, "**container**", on the page. Inside this container are the three items I was mentioning earlier with the tag "**flex-item**". For more custom CSS formatting, they also have their own class – **number**, **explanation**, and **pic**, respectively. Close to the bottom of the page, some of the production steps are too minor to warrant being in a div of their own with some text to explain, so I just grouped them together in a single box.

Sanitation page

This was a simple and easy page to make. In a small block of text just under the header, I outline some of the policies Turri's have in order to ensure clean, safe products: gloves, soap, and hair nets. I was originally going to include lab samples as well, but after two cross-country visits to the plants in the US, I didn't get a good opportunity to take pictures of them, so I'm just not going to bother.

We have a hierarchy of two containers on this page. First, the "**main-big-container**", which contains all the material below the block of text, then the three "**main-container**"s, which each contain a picture and their accompanying description (both of which have the tag "**main-item**"). Each **main-container** has a border on top for some visual separation.

Due to the flex-box formatting of each **main-container**, it looks presentable and readable even as you narrow the web browser.

Workers page

The most frustrating page to design, despite being the one with the least material on it. All there is on this page is: (1) a small bit of text below the header, (2) a picture of an employee, and (3) a make-shift table with the different positions Turri's offers and their respective roles.

Starting off, we have the "**main-huge-container**", which centers the picture and table on the page, the "**main-big-container**", which is a Flexbox that aligns the picture and table horizontally, and finally, the two items inside this container: "**left**", which has the picture, and "**right**", which has the table. The rest of the formatting is trivial.

Contact page

This page just has some contact information at the top and a couple forms to fill out below (name, email, phone, website, company, and any further commentary). The "**form**" div and all the elements inside it have widths that are a percentage of the page width, so that when the browser window is narrowed, the div narrows as well.

Final notes

I will put comments in my CSS; I've read that CSS comments are advisory although HTML ones are not.

I've tidied up my code, cleaned up my assets folder and am ready to make a post on the Landing.