**Week 3:**

Starting off for week 3, we first started learning about hosting a site. Hosting a site is simply sending your website to a server. The server than publishes the site for the world to see. The server is simply a special computer that can store and send information through the internet. It serves the same purpose as a normal computer just specifically built to be better at what it does.  
  
Most server you have to build yourself or buy access to it. Although, there is one free website you can use to upload your website. This website is called Github and it is simply somewhere you can upload files for other people who enjoy programming to see. Although, github has a feature called Github pages which allows you to upload a website by simply sending your files.  
  
**CSS:**

This week we have also learned quite a bit of CSS. Most notable, the <div> tag. The <div> tag is a tag that simply allows us to group different elements of html together. It is simply a way to select specific parts of HTML or text. The <div> tag is simply to use, you simply put the tag along with its closing tag, maybe give it a class/id, and put everything you want to group in that tag. For example:  
  
<div class=”my-first-div”>

<p> This is a paragraph</p>

<p> This is another paragraph</p>  
<a href=”google.com”>Now this is actually a link to google!</a>

<p>Once again a paragraph. All of these things are contained in the div tag</p>

</div>  
  
<p>This paragraph is not contained in the div! This paragraph stays the same</p>  
  
Now if you select:  
  
.my-first-div {

color: red;

}  
  
You had the entire div being colored red but not the paragraph outside of it!  
  
Something else that is similar is the <span> tag. The span tag works the exact same way except that the <span> tag Is inline instead of a block tag and is usually used to select text inside of an element and not elements themselves. Now, what do I mean by inline instead of block. Inline might be a link for example while a paragraph will be a block. Block tags will be “seperated” from the rest. You can think of it as having <br> tags before and after. In comparison, inline tags stay on the same line. They do not have those imaginary <br> tags. These two tags(<span> and <div>) are called grouping elements, since as the name suggest, they group things together.  
  
  
  
  
  
The next thing we learned about in CSS is the height and width property. Height and Width, as the name suggests, can change the height and/or width of an element. To use it, you simply select the element in CSS and use the height/width property and specify the amount. Those properties can be used to resize anything such as images or the amount of space a text takes(not the font size). For example:

.my-element {

height: 100px;

width: 50px;

}

.my-image {

height: 30px;

}

.my-div {

width: 50%;

}

As you can see in “.my-div”, the width is using “%” instead of “px”. What is the difference? Pretty self-explanatory, one is measured in pixels and the other is measured using the \_\_\_\_ of the page. So for this example, it uses the \_width\_ of the page and uses only 50% of it. There are a lot more “units” that can be used but those are the most common.

One more thing about the height and width properties. When used with an image, it is best to specify only one value. If both values are specified, there is a chance that the image will be squished. It is best to use one value and leave the other for the browser to calculate.  
  
  
  
The next thing we learned about was actually something called, “overflow”. When you change the width/height of text, it can do some pretty weird things. For this reason, there is something called overflow. Overflow is what deals with the… well… overflowing text. To use the “overflow” property, you simply need to specify one of those values.

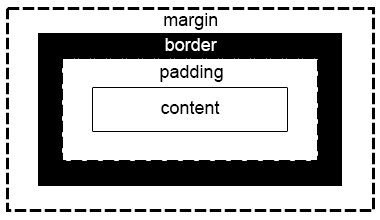
|  |  |
| --- | --- |
| Auto | Adds a scrolling bar to see the overflowing text |
| Hidden | Hides the overflowing text |
| Visible | The default option. Allows you to see the overflowing text at all times. |

So for example:

.my-paragraph {

width: 30%;

overflow: Auto;

}  
  
  
  
The next thing we learned about is the CSS box model.  
  


This model shows the 3 different kinds of “margins” you can have. I will explain each one.

|  |  |
| --- | --- |
| Margin | The invisible space outside of an element. The outside of this table for example. |
| Border | The outline of the element. For example, the outline of this table. |
| padding | The space between the edge of the element and the text. For example… you guessed it. This table. |

For each of these, you simply precise a value and a unit and it will add the space. Feel free to experiment with it. Although, here is an example of how to use it.  
  
.my-paragraph {

margin: 50px;

border: 5px;

padding: 50%;

}

It is a good idea to specify the color as well. You can do that by simply adding the color after the measure. The default is always black.

margin: 50px green;  
  
  
Something that applies only to the border is that you can also change the style of the border. For example, you can have it be dotted or dashed, solid or double lined! There are multiple styles each doing different things, so make sure to find all of them and choose which one you prefer for the specific task. Anyways, the style goes just before the color but after the measure.

border: 5px dotted red;

One more thing about each of these, you can specify what sides they appear on. The way to do this is to simply specify the side with a hyphen before (margin-top:50px for example).  
  
  
  
  
We have also learned how to position elements using CSS.