* Explain the importance of responsive design and its impact on the target audience. (minimum 50 words).
  + **Responsive Design is critical in Web Design. By implementing responsive design into any website can help users to access the site from any mobile device at any given time. Responsive Design is also a huge factor with Search Engine Optimization because Google’s search algorithm prioritizes visibility and accessibility. A percentage of users now use mobile devices to access the internet. Responsive design improves user experience because of the content layout and easy navigation. This benefits users so they have an easy browsing experience across multiple devices. Users can have more of a stable experience by using any device no matter where they are at.**
* Google Analytics provides a wealth of data for free to help assess the success of websites. Describe three different kinds of reports that will tell you what information your target audience is accessing (minimum 100 words).
  + **Acquisition Reports are one way that will tell us how users are accessing the site. Whether that be through paid advertising or organically by search results, different types of browsers, mobile or desktop, and more.**
  + **Engagement Reports are another option to see what content drives the engagement or conversion on the site. These reports can be used to track downloads, form submissions, purchases and more. One might be able to track what pages are converting users and what channels they may be coming from whether that be from social media, organic traffic, or paid search.**
  + **Monetization Reports allows one to see how much revenue the website is generating from products, downloads, ads, and subscriptions.**

**Preprocessing**

* Explain the ways to implement preprocessing (client-side vs server-side) (minimum 50 words).
  + **A few ways that we can implement preprocessing is to write code in a less or scss file, download a compiler such as node.js which would be the server side and in return will convert the less styles to CSS styles that will be interpreted by our HTML and web browsers.**
* Define and describes the mechanisms used in either LESS or SASS: mixins, variables, functions, operators, nested variables (minimum 100 words)
  + **Mixins – allow one to store a few styles with one selector. It’s a way to condense a group of styles and apply them to multiple elements to create unique components**
  + **Variables – values that can be stored for later use. These values can be placed anywhere.**
  + **Functions – used to create more dynamic CSS options that are built in. These include options to transform colors, do math, and manipulate strings.**
  + **Operators – typically the + - \* / mathematical operations that take units and convert numbers before adding, subtracting, or comparing them. These can be used on any number, color, or variable.**
  + **Nested Variables – takes child styles and adds them to a parent style. This helps keeps code more simple and concise.**
* Implement LESS or SASS preprocessing.
  + Link and comment out LESS or SASS file in the HTML file including explanation of how it was executed.
    - How I implemented LESS: I used the terminal from the macOS to download the files that I needed including homebrew, yarn, less and finally the less-watch-compiler to execute the code to compile the less to css. This allowed me to use less seamlessly.
  + Example :  
    <!-- How I implemented LESS: I used a macOS software called CodeKit to compile my .less files into .css files. This application allows me to set up my project in such a way where I can have a source folder, which includes all my source files such as .less files, and it automatically creates a build folder containing all the compiled files such as the .css files, every time I make a change to the source files and save it. I took advantage of LESS variables for setting up the colors. I also used mixins for creating the bottom borders. -->