

### **Build Your HTML**

1. Create your html boilerplate in the index.htmlfile.
2. Add a new file called style.cssand link it to the HTML file.
3. In the body of your HTML, use [header](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/header), [main](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/main), and [footer](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/footer)tags to clearly define the different parts of the page.
4. Use an h1tag for the site title and an h2for the subtitle inside the page header.
5. Use [nav](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/nav)tags inside the page header to create an accessible navigation. Follow the example in the [W3C Web Accessibility Initiative (WAI) Guidelines](https://www.w3.org/WAI/tips/developing/#provide-meaning-for-non-standard-interactive-elements) to format your navigation:
   1. Wrap each link in an anchor tag
   2. Wrap each anchor tag in a list item
   3. Wrap all of the list item tags in an unordered list tag
   4. Place the unordered list inside the the nav tag and give it attributes for aria-label="Main Navigation"and role="navigation"
6. Mark up both of the blog posts using the [article](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/article)tag.
7. Use this [W3C WAI Guideline](https://www.w3.org/WAI/tips/developing/#use-mark-up-to-convey-meaning-and-structure) example, to help you structure your blog post content. Follow the pattern! Make sure there is a ptag that contains some lorem ipsum text (Remember you can create this with Emmet by typing p>loremand pressing tab)!
8. Make sure your images all have altattributes so that they are accessible.
9. Inside the page footer, use the same technique you used earlier to create a semantic and accessible navigation, using a wrapping navtag, unordered list and the aria attributes.
10. Add a copyright in the footer. For the © symbol, use an [html entity](https://www.w3schools.com/html/html_entities.asp).

*Make sure to add and commit your changes when you've got the html done!*

### **Style Your Page**

1. Use a [border](https://www.w3schools.com/css/css_border.asp)only on the left side of the page's htmlelement. It should have a width of 5pxand a color of lightgray.
2. Set the bodyelement's min-heightto be at least 100% of the [viewport height](https://alligator.io/css/viewport-units/). Also, use the font-familyproperty to set the entire page to use sans-seriffonts.
3. Give the bodya max-widthof 1000pxand center it on the page with the margin auto technique.
4. The border is too close to our text and we've got an issue with [margin-collapse](https://medium.com/@joseph0crick/margin-collapse-in-css-what-why-and-how-328c10e37ca0) at the top of the page! Add paddingto the header, mainand footerelements of 1remon the top and bottom and 2remon the left and right to fix these issues.
5. For the h1on the page we want the color to be tomato.
6. Set all of the images to have a widthof 100%.
7. For your site navigation, you'll need to remove all of the padding on the ulelements and then set it's list-style-typeto none. It would be a good idea to use a class to style the navigation so that you can still make a bulleted list on your page. Use Flexbox to style the ulfor the navigation elements. On the top navigation use the space-betweenrule to have the menu items spread across their container evenly. Set the color for the anchors inside the main navigation to be lightgrayand the footer navigation to be tomato.
8. Style the title of each of your blog posts so that they are larger.
9. Style the 'continues...' anchor tags so that they are aligned to the right and are bold and tomato colored.
10. Set the margin on the bottom the articletags to be 4rem. Also, add a 1pxsolid border in lightgrayto the bottom of each one and give each padding on the bottom of 2rem.
11. For the ptag inside your article, use the [pseudo-element ::first-letter](https://developer.mozilla.org/en-US/docs/Web/CSS/::first-letter) to style the drop cap and set it's color to lightgray.
12. Style the copyright so it is lightgray.

*Awesome job, add, commit and push changes to create your pull request!*

### **Part 2**

* REMAKE ENTIRE PROJECT IN REACT and deploy to NETLIFY
* Use App.js, Header.js, Nav.js, Article.js, Footer.js