# eCOMMERCE WEB SITE

## Introduction

Today we jump into the wonderful world of HTML and CSS. HTML allows us to mark up our web site content, giving it structure and meaning. We then take that content and position it, style it, and make it dynamic based on the device’s display by using CSS!

Regardless of the technologies you use to make web applications, HTML and CSS will always be used. Having some familiarity with core markup + styling concepts will put you in a good place, no matter the tech stack being used.

For this project, you will create a new web site that will serve as a home page for a business. By referencing a UX designer’s prototype and using the asset team’s images, you’ll have a clear vision of what the finished product should look like.

We always use HTML & CSS for front end.

## Technologies

Adobe XD  
HTML  
CSS

## Learning Objective

Create a professional web site that will introduce you to interacting with product designs while learning the basics of working with HTML and CSS.

💡 This is a SINGLE page web site. The only files you should have (aside from any images) are **index.html** and **styles.css**

## Resources

**Lectures**

* Design and UX
* HTML
* CSS

**Documents**

* eCommerce Web Site User Stories

## Tasks don’t forget GitHub

## HTML (Index.html): entry point, home page,

## CSS (syles.css), README File, gitignore, image folder: assets

1. Spend about half an hour going over the intricacies of the UX prototype to better understand various components and how they all interact
   1. Look over what styling has been applied, specifically what images, fonts, and colors are used (Is the font bold? What is the Hex color code for that item? Is there text overlaying a background image?)
2. Write the HTML (Skeleton) for the web site. Account for every element used in the prototype, ensuring good, semantic markup is used to define the meaning of your content in addition to proving structure (A <div> is helpful as a wrapper, especially when positioning a child element, but not *every* element should be one!).
   1. **SEE SETUP STEPS BELOW FOR PROJECT SETUP**
3. Write the CSS for the web site. Focus on layout and positioning first (CSS Grid will help with this) before moving onto more specific styling. Remember, you should have, at minimum, 10 CSS rule sets in your stylesheet (rule = selector + declaration).
4. Once you have completed the *required* user stories, move on to the bonus stories.
   1. **YOU WILL ONLY RECEIVE BONUS CREDIT FOR RESPONSIVENESS IF YOU UTILIZE MEDIA QUERIES IN YOUR STYLESHEET.**

## Setup Steps

1. Create a repository on GitHub and clone it down as normal.
   1. Don’t forget the **README.md** as well as the **.gitignore** (You can select NODE for this project).
   2. Once you have navigated into your cloned repo locally, create your project files **index.html** and **styles.css**.
   3. Create a new folder inside the project called “assets” to contain your Adobe XD prototype and product images.
2. It is important to name your HTML document **index.html** so that it functions as an entry point for our web site, allowing us to create new pages later from this home page.
3. Begin writing your markup in the HTML file. Account for all sections of the web site and each section’s content.
4. Once you have written most of the HTML, move on to CSS to start positioning your content, starting from the top of the page and working your way down.
   1. NB: We recommend setting a few basic rules in your stylesheet to make your life easier. Write these 2 rules at the top of the file:

Text

Description automatically generated

## End Result

The final product will be a professional-looking web site that will have made you more comfortable converting your team’s designs into code using the front end technologies of HTML and CSS!

Other Resources

We’ve curated a set of links you will find useful in your HTML/CSS journey. There are many resources out there, so be sure to research carefully! AVOID using video tutorials that you don’t understand to accomplish your goals, as this will create potential problems down the road in your learning.

### W3 Schools

This web site is one of the most trusted and popular resources for learning a variety of development topics. It covers almost all topics you will need for HTML and CSS. If you only go one place, go here!

<https://www.w3schools.com/html/default.asp>

### Background Images

Getting background images to work can be tricky. The two resources below make it easier!

<https://css-tricks.com/perfect-full-page-background-image/>

<https://www.w3schools.com/cssref/pr_background-image.asp>

### Centering Content vs. Aligning Content

A common source of frustration for beginners is controlling the position of your content.

<https://moderncss.dev/complete-guide-to-centering-in-css/>

### Colors

When finding your color scheme, these resources are great tools. Coolers in particular is fantastic for flipping through colors until you find ones that look great together.

<https://coolors.co/>

<https://mycolor.space/>

### Semantic HTML

HTML, contrary to popular belief, has nothing to do with the styling of your content. For example, we don’t choose headers to make fonts bigger (that’s what CSS is for). We choose elements in HTML that tell the user what the content MEANS. Read below for more!

<https://www.thoughtco.com/why-use-semantic-html-3468271><https://keithjgrant.com/posts/2018/03/html5-sectioning-and-landmark-elements/>