# WEB222 Assignment 5

# Overview

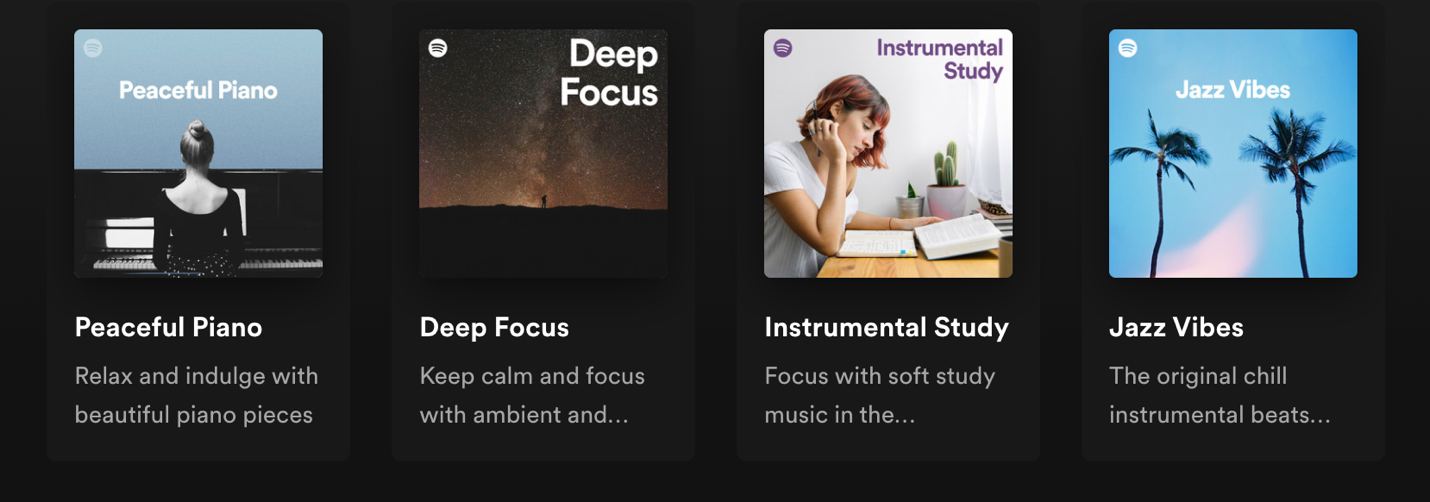
This assignment is designed to have you practice building more complex HTML and CSS layouts.

You must do all the work for this assignment on your own. You may consult your notes, use the web for inspiration, but you should not copy code directly from other sites, or other students.

You are to design and develop a website (3 mandatory pages at least – file names are described later) with necessary css. The home page (index.html) of the website will display images of at least 10 similar products with related information such as title, description, price etc. Be aware of copyright for the images you use. Refrain from using abusive text or images. You may refer to amazon, ebay, Walmart for design guidance. However, you are not allowed to direct copy any code from any website. You will be graded based on creativity.

Cards

Cards on the web, much like trading or playing cards, are rectangular areas that allow you to visually present a lot of related data. We often see them used in online stores, social media, and anywhere that we want to mix images, titles, and text in a rectangle. Here are some real-world examples from Spotify, Amazon, SoundCloud, and airbnb:



Graphical user interface, website

Description automatically generated

Graphical user interface, website

Description automatically generated

A picture containing text, outdoor

Description automatically generated

There are lots of resources you can use to learn more about creating a card, for example:

* <https://developer.mozilla.org/en-US/docs/Web/CSS/Layout_cookbook/Card>
* <https://www.w3schools.com/howto/howto_css_cards.asp>
* <https://www.freecodecamp.org/news/learn-css-basics-by-building-a-card-component/>

Update Your Store (index.html) to Use Cards

Use CSS classes on your card’s elements in order to apply colours, fonts, margins, padding, borders, etc. until you have something that you think looks good.

1. Create rows of cards. Use **Flexbox** or **CSS Grid** to create rows that repeat your cards across and down. For now, you can copy and paste your card from step 1 over and over in order to repeat it. Make your page look good with rows of 3 or 4 cards. Adjust the spacing, size, etc. until you’re happy with how it looks.

Card 1

Card 2

Card 4

Card 3

1. Make sure you optimize the images so they are not too big to download (i.e., don’t use a 5000x6000 image in a card that uses 400x200).

You can use <https://squoosh.app/> for images that you download. Or you can also use a trick with <https://unsplash.com/> images to resize them automatically via the URL. For example, this bike image <https://unsplash.com/photos/tG36rvCeqng>. Here’s the full-sized image <https://images.unsplash.com/photo-1485965120184-e220f721d03e> (it’s 3.8M in size, and 4440x2960). We can reduce that image by adding some parameters to the image URL: **?auto=format&fit=crop&w=750&q=80** to crop and resize it to 750 pixels wide, and reduce the quality a bit to 80%, like this: <https://images.unsplash.com/photo-1485965120184-e220f721d03e?auto=format&fit=crop&w=750&q=80> See <https://unsplash.com/documentation#dynamically-resizable-images> for more details.

1. Add A Basic Stylesheet: Y**ou are encouraged to use one of the various “class-less” CSS stylesheets** described here: <https://css-tricks.com/no-class-css-frameworks/> These stylesheets can be included in the <head></head> of your document, for example:

<head>

<link rel="stylesheet" href="https://unpkg.com/mvp.css">

Try experimenting with some of these stylesheets to find one that makes your page look good to you.

## Coding:

Use the website starter project in the assignment ZIP file. Install all dependencies by running the following command in the root of the assignment (e.g., in the same directory as package.json):

**npm install**

Your code should all be placed in the src/ directory. You will find 3 HTML files there now, which should be updated by you as follows:

1. **src/index.html** – Your main web page should go here. Your cards i.e., the products go in this page. Be sure to create a proper HTML5 document and also include hyperlinks to the about.html and honesty.html pages.
2. **src/about.html –** Include information about yourself (the author). Be sure to create a proper HTML5 document as well as **links to index.html and honesty.html**.
3. **src/honesty.html** – Include the standard text for student submissions, as well as credits for any and all resources you used in your site (e.g., citations for images, videos, etc). Be sure to create a proper HTML5 document, as well as **links to index.html and about.html.**

NOTE: you are welcome to create other pages if you need them. Just remember to link all of your pages together and to have a consistent navigation in between pages.

You will learn more about the starter project in later courses.

## Running a Web Server:

You can start a local web server to test your code in a browser by running the following command:

**npm start**

This will start a server on <http://localhost:3000>, which you can open in your web browser

To stop the server, use CTRL + C

## Submission:

When you are finished, run the following command to create your submission ZIP file:

**npm run prepare-submission**

This will generate submission.zip, which you can hand in on Blackboard. Do not wait for the last moment to upload. Late penalty is 100%.