

# Creating a Site Plan

---

## Activity Instructions

Estimated Time: 90 minutes

### What is a Site Plan?

## Overview

- **Task:** Create a site plan to lay the groundwork for the site development.
- **Purpose:** Prepare your website with the proper foundational information and guides.

## Background

Developing a site plan is a critical step in any web design process. Site plans help establish the purpose of the site, the voice, and overall look and feel, and then they help you apply those consistently across your site.

Site plans come in many shapes and sizes and in varying levels of detail. Here is a list of common elements in a site plan:

- Statement of purpose and goals
- Definition of audience
- Personas and scenarios (Who will use the site and how will they use it?)
- A style guide:
  - Branding information (how to use the branding for the site correctly)
  - Typography choices and justification
  - Color palette
  - Specific styling of elements on the site (headings, lists, paragraphs, forms, links, buttons, and so on)
  - A site map

- Wireframes showing general layout and content for major pages

We will be making a simplified site plan for our web site. Ours will focus on the sections for purpose, audience and a partial style guide. Since the style guide will be the largest portion, let's take a look at an example to get a better idea about how site plan might be put together.

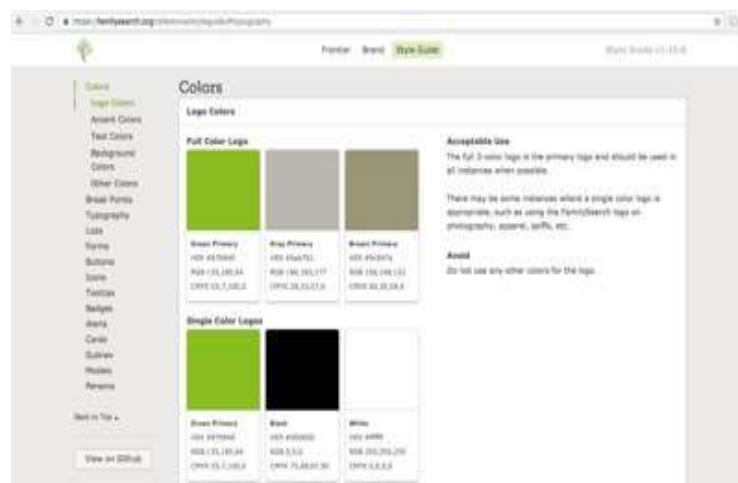
## Example

This may not be a site plan, but this style guide demonstrates similar purposes of our siteplan we will create.

## FamilySearch



[FamilySearch Home](https://www.familysearch.org)



[FamilySearch Style Guide](https://www.familysearch.org/content/familysearch.org/page.nsf/00000000000000000000000000000000~open?open=1)

FamilySearch has a nice style guide that is viewable by anyone. Take a moment and think about the purpose and audience of the site. Does the site accomplish the purpose well and cater to its audience?

This site is much bigger and more complex than what we will do for our site. However, even though our site will be a simple website for a White Water Rafting Company, this planning phase is still important, and you are expected to produce a site plan that is complete and professional-looking.

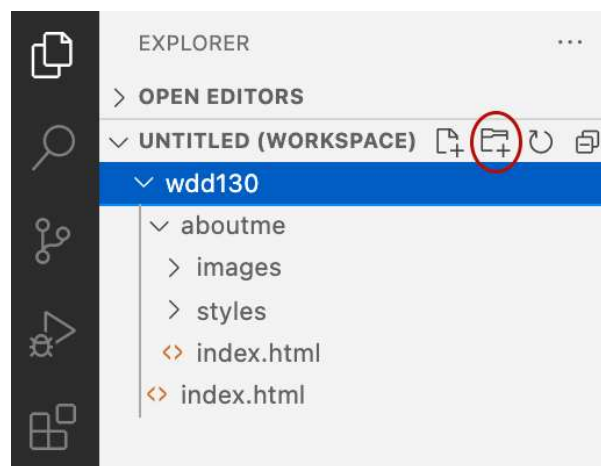
## Instructions

### Prepare a place for our site plan

As we set up GitHub Pages, we created a 'wdd130' repository with an index.html file. Now we will create a folder within the wdd130 folder to hold our White Water Rafting project. The folder will be called 'wwr'. It will contain our site plan that we will work on this week and eventually all other files that make up our White Water Rafting site.

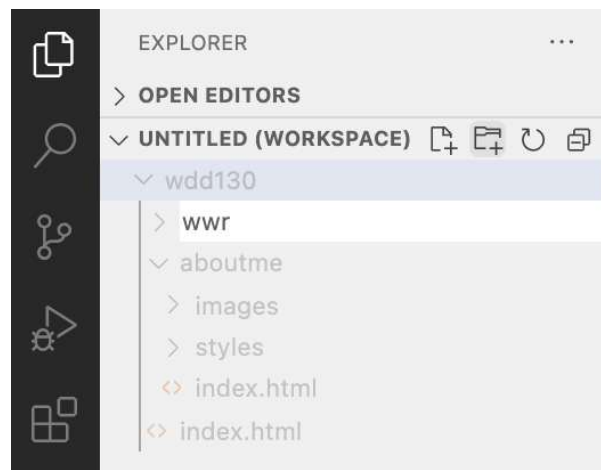
Open Visual Studio Code and make sure your project folder 'wdd130' is added if it is not already there. You should see the project in the Explorer panel.

Click on the 'New Folder' icon



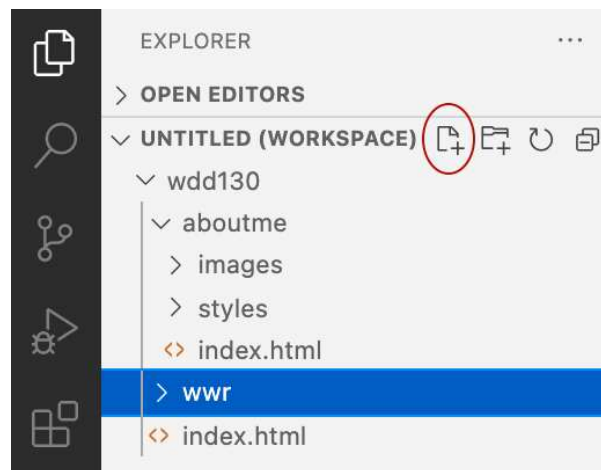
Add a new folder

and name the folder 'wwr' for white water rafting.



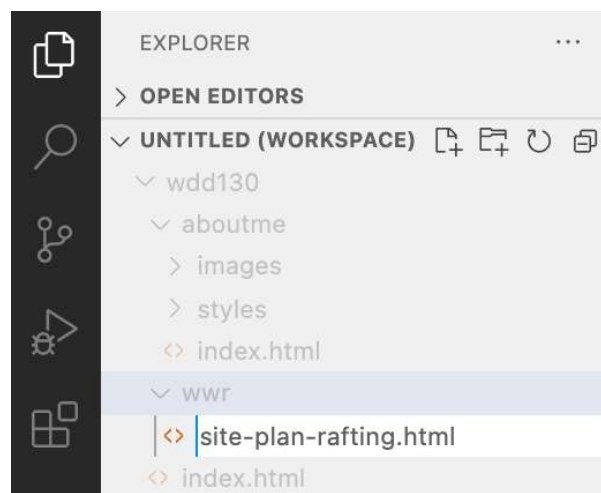
Name the folder wvr

Within the wvr folder, we will create a new file. Select the 'New File' icon.



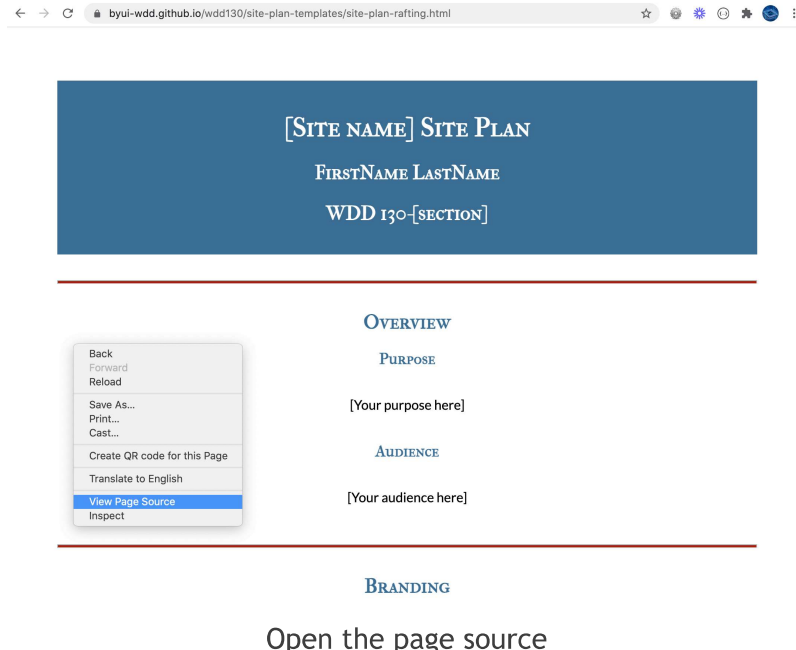
Create a new file

Now add the site-plan-rafting.html page.



Create an HTML file called site-plan-rafting.html

Go to [Site Plan Template](#). Right-click on the page and view the source code. (On Chrome, it's the pop-up menu option called **View Page Source**)



Copy **all** the HTML code. (not all the code shows on the image, but you will still need to copy all the code). You can use Cmd+A or Ctrl+A to select it all quickly, then copy it.

```
view-source:https://byui-wdd: x +
view-source:https://byui-wdd.github.io/wdd130/site-plan-template...
Line wrap
1 <!-- HTML -->
2 <html lang="en-us">
3
4 <head>
5   <meta charset="utf-8">
6   <title>Site Plan</title>
7   <link type="text/css" rel="stylesheet" href="styles/site-plan-rafting.css">
8 </head>
9
10 <body>
11   <header>
12     <h1>[Site name] Site Plan</h1>
13     <h2>First Name LastName</h2>
14     <h2>WDD 130-[section]</h2>
15     <!-- In the header above, add the name of your site, your name and class number. For example if you are a student, you might have: <h1>My Cool Site Plan</h1> <h2>John Doe</h2> <h2>WDD 130-Web Development</h2> -->
16   </header>
17   <main>
18
19     <!-- Steps 2-3 -->
20     <hr>
21     <h2>Overview</h2>
22     <h3>Purpose</h3>
23     <p>[Your purpose here]</p> <!-- change this -->
24
25     <h3>Audience</h3>
26     <p>[Your audience here]</p> <!-- change this -->
27
28     <hr>
29     <h2>Branding</h2>
30     <h3>Website Logo</h3>
31     <!-- Replace this with some sort of logo for your site. A logo can be as simple as the name of your site. -->
32     
33     <hr>
34     <h2>
35       Style Guide
36     </h2>
37
38     <!-- Steps 4-8 -->
39
40     <h3>
41       Color Palette
42     </h3>
43     <!-- The colors you choose for a website are one of the most important decisions you will make. Follow the steps below to create a color palette. -->
44
45     <!-- Copy and paste the URL to your finished palette below. Replace the href value that is there with the URL to your palette. -->
46     <!-- Palette URL: <a href="https://coolers.co/396e94-e7c24f-a43312-381d2a-aabd8c" target="_blank">https://coolers.co/396e94-e7c24f-a43312-381d2a-aabd8c -->
47
48     <table class="colors">
49       <tr>
50         <th>Primary</th>
51         <th>Secondary</th>
52         <th>Accent 1</th>
53         <th>Accent 2</th>
54       </tr>
55     </table>
56   </main>
57 </body>
58 </html>
```

Copy the page source code

Paste the code into your site-plan-rafting.html page in VSCode and save the file.

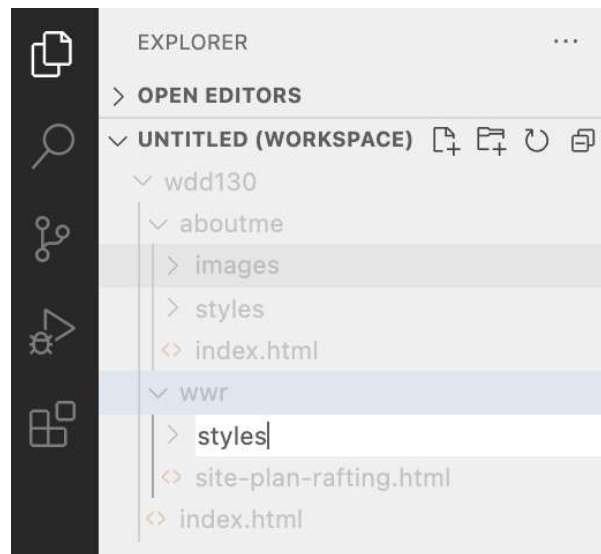
```
site-plan-rafting.html — Untitled (Workspace)
site-plan-rafting.html U X
wvr > site-plan-rafting.html > html
1
2 <!DOCTYPE html>
3 <html lang="en-us">
4
5 <head>
6   <meta charset="utf-8">
7   <title>Site Plan</title>
8   <link type="text/css" rel="stylesheet" href="site-plan-rafting.css">
9 </head>
10
11 <body>
12   <header>
13     <h1>[Site name] Site Plan</h1>
14     <h2>FirstName LastName</h2>
15     <h2>WDD 130-[section]</h2>
16     <!-- In the header above, add the name of your site, your name and class number. For
17         example if you are in section 3 you would put WDD 130-03 -->
18   </header>
19   <main>
20     <!-- Steps 2-3 ----- -->
21     <hr>
22     <h2>Overview</h2>
23     <h3>Purpose</h3>
24     <p>[Your purpose here]</p> <!-- change this -->
25
26     <h3>Audience</h3>
27     <p>[Your audience here]</p> <!-- change this -->
28
29     <hr>
30     <h2>Branding</h2>
31     <h3>Website Logo</h3>
32     <!-- Replace this with some sort of logo for your site. A logo can be as simple as
33         the name of your site in a nice font :) -->
34     
36     <hr>
37     <h2>
```

Paste the code into site-plan-rafting.html

**NOTE:** You should see a new number on the Source Control icon (the little blue circle with a number). Git has noticed that we added a new file. We won't commit or push until we are all done with the activity.

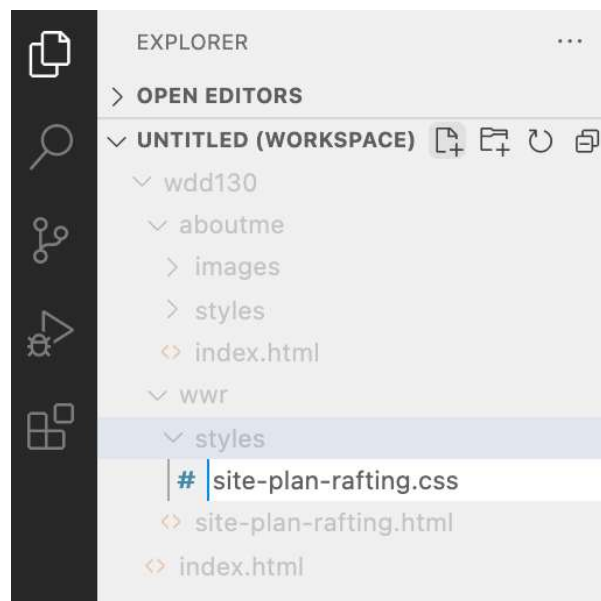
We also need the CSS file that goes with our HTML. Let's make a styles folder to hold the CSS.

With the 'wvr' folder selected, select the **New Folder** icon and call the new folder 'styles'.



Make a folder to hold CSS

With that new styles folder selected, add a new file called site-plan-rafting.css within the styles folder.



Add a new site-plan-rafting.css file

Go to [Site Plan CSS](#). Copy all the code.



```
https://byui-wdd.github.io/wdd130/site-plan-templates/site-plan-rafting.css
byui-wdd.github.io/wdd130/site-plan-templates/site-plan-rafting.css

/* if you are using any Google fonts change the font names below to your fonts.
Any spaces in your font name should be replaced with a +.
Fonts are separated by a | */
@import url('https://fonts.googleapis.com/css?family=IM+Fell+French+Canon+SC|Lato');

:root {
  /* change the values below to your colors from your palette */
  --primary-color: #396E94;
  --secondary-color: #E7C24F;
  --accent1-color: #A42212;
  --accent2-color: white;

  /* change the values below to your chosen font(s) */
  --heading-font: "IM Fell French Canon SC";
  --paragraph-font: Lato, Helvetica, sans-serif;

  /* these colors below should be chosen from among your palette colors above */
  --headline-color-on-white: #396E94; /* headlines on a white background */
  --headline-color-on-color: white; /* headlines on a colored background */
  --paragraph-color-on-white: #396E94; /* paragraph text on a white background */
  --paragraph-color-on-color: white; /* paragraph text on a colored background */
  --paragraph-background-color: #396E94;
  --nav-link-color: #396E94;
  --nav-background-color: #D4AC0D;
  --nav-hover-link-color: white;
  --nav-hover-background-color: #396E94;
}

/* Look around below...but DON'T CHANGE ANYTHING! */

body {
  max-width: 960px;
  margin: 0 auto;
  padding: 4em;
  font-size: 18px;
  text-align: center;
}

img {
  display: block;
  margin: 0 auto;
  max-width: 300px;
}

h1, h2, h3, h4, h5, h6 {
  font-family: var(--heading-font);
  color: var(--headline-color-on-white);
}

h2 {
  text-align: center;
}

hr {
  height: 3px;
}
```

Copy all the CSS code

Paste the code into the site-plan-rafting.css file in VSCode and save it.

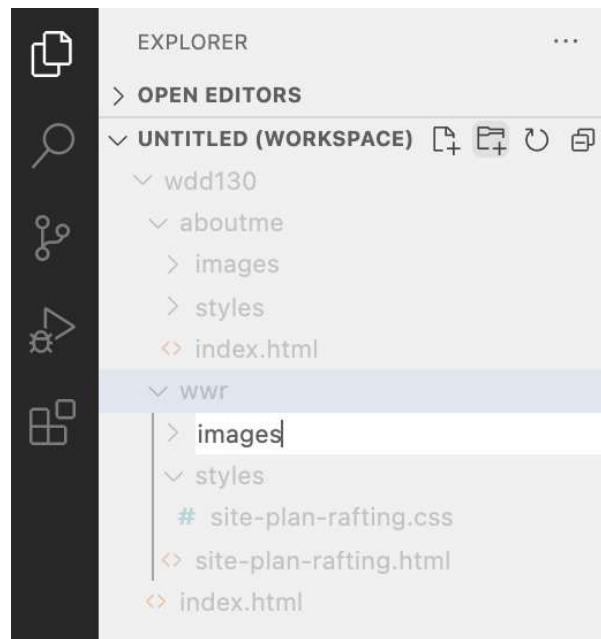
```
site-plan-rafting.css — Untitled (Workspace)
site-plan-rafting.html U # site-plan-rafting.css X
wvr > styles > # site-plan-rafting.css > :root

1 /* if you are using any Google fonts change the font names below to your fonts.
2 Any spaces in your font name should be replaced with a +.
3 Fonts are separated by a | */
4 @import url('https://fonts.googleapis.com/css?family=IM+Fell+French+Canon+SC|Lato');
5
6 :root {
7   /* change the values below to your colors from your palette */
8   --primary-color: #396E94;
9   --secondary-color: #E7C24F;
10  --accent1-color: #A42212;
11  --accent2-color: white;
12
13  /* change the values below to your chosen font(s) */
14  --heading-font: "IM Fell French Canon SC";
15  --paragraph-font: Lato, Helvetica, sans-serif;
16
17  /* these colors below should be chosen from among your palette colors above */
18  --headline-color-on-white: #396E94; /* headlines on a white background */
19  --headline-color-on-color: white; /* headlines on a colored background */
20  --paragraph-color-on-white: #396E94; /* paragraph text on a white background */
21  --paragraph-color-on-color: white; /* paragraph text on a colored background */
22  --paragraph-background-color: #396E94;
23  --nav-link-color: #396E94;
24  --nav-background-color: #D4AC0D;
25  --nav-hover-link-color: white;
26  --nav-hover-background-color: #396E94;
27
28
29
30 /* Look around below...but DON'T CHANGE ANYTHING! */
31
32 body {
33   max-width: 960px;
34   margin: 0 auto;
35   padding: 4em;
36   font-size: 18px;
37   text-align: center;
38 }
39
40 img {
41   display: block;
42   margin: 0 auto;
43   max-width: 300px;
44 }
```

Paste all the CSS code

While we are on file management, let's add an image folder to our project as well. In the site plan, you may choose to use your own logo. The images folder is where you will place that logo.

Add a folder to the project called images. It won't have anything in it for now.



Add an images folder

## Make Additions and/or Changes to the Template

Open the site-plan-rafting.html file in your VSCode editor. In the `<header>` section of the code, add your site name, your name, and section number. (Notice that anywhere there are [ ] square brackets, you will be replacing them with your information.)

```

1  <!DOCTYPE html>
2  <html lang="en-us">
3
4
5  <head>
6    <meta charset="utf-8">
7    <title>Site Plan</title>
8    <link type="text/css" rel="stylesheet" href="site-plan-rafting.css">
9  </head>
10
11 <body>
12   <header>
13     <h1>[Site name] Site Plan</h1>
14     <h2>FirstName LastName</h2>
15     <h2>WDD 130-[section]</h2>
16     <!-- In the header above, add the name of your site, your name and class
17          number. For example if you are in section 3 you would put WDD 130-03 -->
18   </header>
19   <main>
20
21     <!-- -----Steps 2-3----- -->
22     <hr>
23     <h2>Overview</h2>
24     <h3>Purpose</h3>
25     <p>[Your purpose here]</p> <!-- change this -->
26
27     <h3>Audience</h3>
28     <p>[Your audience here]</p> <!-- change this -->
29
30     <hr>
31     <h2>Branding</h2>

```

Replace brackets with your information

In the next few steps, you will also be changing the `<p>` element content under the 'Purpose' `<h3>` element, where it says '[Your purpose here]', and the audience paragraph, where it says '[Your audience here]'. The next steps explain what to type there.

```

4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
<!-- In the header above, add the name of your site, your name and class
number. For example if you are in section 3 you would put WDD 130-03 -->
<!-- -----Steps 2-3----- -->
<h2>Overview</h2>
<h3>Purpose</h3>
<p>This is where I will type my purpose of the website. I will point out
why we need the website; what we want to accomplish with it. And how this
will benefit our rafting company.</p> <!-- change this -->
<h3>Audience</h3>
<p>This is where I will type who I think the type of person who would visit
this site would be. What type of person do we need to keep in mind as we
are building our site. How will our content, colors, fonts, etc. be
something that would be inviting to this kind of person?</p> <!-- change
this -->
<h2>Branding</h2>

```

Replace brackets with your information

## Purpose Statement

From the beginning, websites have always been about displaying content. This is the underlying purpose behind every site on the Internet, or at least it should be. In order to determine which content you should put on the site, it is critical to determine the purpose or goals of the site. The content should support that purpose. At a high level these purposes can take many forms:

- **Build Revenue** : Build revenues for your new/existing business or
- **Share Information** : Share corporate or educational information or
- **Share Opinions** : Share opinions on a subject
- **Share Personal Interests** : Share personal interests with family and friends

All websites should have a purpose, and that purpose should be more than 'My teacher is making me do this.'

We need to do this now for our website. One problem: the purpose really needs to be defined by the site owner and not the site builder. So for this step and the next, imagine that you are the owner of the business and have decided that it is time for you to have an online presence. **Develop a purpose statement that clearly defines your goals for the website.** Find the portion of the [site-plan-rafting.html](#) file for the purpose. It looks like this:

```
<h3>Purpose</h3>
<p>[Your purpose here]</p>  <!-- change this -->
```

Replace [\[Your purpose here\]](#) with your purpose statement.

## Audience Statement

Another important factor to keep in mind when planning a website, is the audience for the site. Remember, that websites are all about content. The purpose helps define what content you should include, and the content should target your specific audience or audiences (websites can have more than one).

**Write a statement specifically identifying your target audience.** Try to answer these questions:

- Who are the target customers? Describe their life (or business) situation (age, financial situation, interests, and so on)
- What do they want?
- What are their needs that aren't being met?
- How will my audience usually access my site (desktop, laptop, mobile device)?

Add your Audience to your site plan just like you did for the Purpose above.

## Choose a logo

You can keep the logo that is already in the site plan, or you can find or create a logo of your own. If you find an different logo or rafting company name, you are welcome to change the name from 'Dry Oar' to whatever name you'd like to have.

The current logo is linked directly to an image that is already on the Web.

```
site-plan-rafting.html X # site-plan-rating.css U
wwr > site-plan-rafting.html > html > body > main > img
30 <hr>
31 <h2>Branding</h2>
32 <h3>Website Logo</h3>
33 <!-- Replace this with some sort of logo for your site. A logo can be as
34 
35 <hr>
36 <h2>
37 Style Guide
38 </h2>
39
```

Current logo location on web

If you want to replace the logo with another one, place the image file into the image folder. You can even drag it into the folder. Then change the path value of the src to reference your new logo.

```
site-plan-rafting.html X # site-plan-rating.css U
wwr > site-plan-rafting.html > html > body > main > img
29
30 <hr>
31 <h2>Branding</h2>
32 <h3>Website Logo</h3>
33 <!-- Replace this with some sort of logo for your site. A logo can be as
34 
35 <hr>
36 <h2>
37 Style Guide
38 </h2>
39
```

Changed logo location to local images folder

The path is now pointing to the images folder and then the name of the image file inside the folder. Yours might have a different name than 'dryoarlogo.png'. Use the image file name you have in the images folder. The .png file type is usually best for logos.

Here are some logos you can use: [Logo Examples](#). You can right-click on them and save the image into your images folder.

## Color Palette

Color plays an essential role in any website and so the colors you use should be chosen deliberately. The follow video is on color theory and can help with these choices.

Below is a video with some tips on how to create your color palette. It covers steps 7-9.

## Using Coolors.co

Visit [coolors.co](https://coolors.co) and, keeping your site purpose and audience in mind, build a color scheme using it. Start by thinking of a color you like or think will work well for your site.

Explore the color palettes. You will see five color stripes. Each color has four items on that show when you hover over them: Alternative Shades, Drag, Adjust, and Lock. Play around with one color until you end up with something like you imagined above, **then lock it**. Hit the space bar on your keyboard to generate different complementary color schemes for the color you chose. Feel free to lock other colors and generate until you end up with 3-4 colors you like.



## Record the color codes

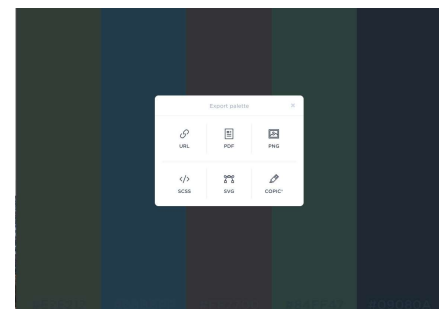
Once you have built your palette, note that at the bottom of each color column there is a six digit code. This is what is known as a hexadecimal color code and is usually preceded by a "#". These codes are the most common way to represent colors on the Web. Copy that code for the 3-4 colors you would like to use into the site plan document. Place the color you want to use the most often as the primary.



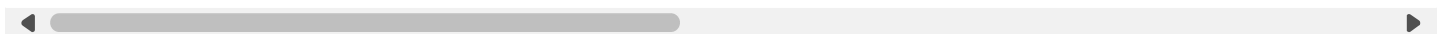
After entering the codes into the HTML file, switch to the `site-plan.css` file. Notice that there is a place to record your colors there as well. Do so, then look at the Preview. You should see your colors showing in different areas!

## Save your color scheme

You should save your chosen palette URL from colors.co in case you need to make any changes later. Copy the URL of your palette. (it might look something like this: <https://coolors.co/b7d3f2-afafdc-8a84e2-84afe6-79beee>) and paste it into your `site-plan.html` in two places.



Palette URL: `<a href="https://coolors.co/396e94-e7c24f-a43312-381d2a-a2"`



Notice that when working with links in HTML the destination of the link goes in the `src` attribute. Then the part you select goes inside of the `<a>` element. Often those will be different...but in this case we are going to make them both the URL to your color scheme.

## About Fonts

Along with the colors you choose, fonts have the next largest (or maybe even larger in some cases) impact on the look and feel of your site. To help you think about this, watch the following video:

The next video gives some tips on adding the fonts to your site plan. It covers steps 11 and 12.

## Choose your fonts

Choose one or two fonts to use for the site and record them in the site plan. Then you need to enter your fonts into the CSS file as well so you can really see what they will look like.

If your fonts are from [Google](#), then you will need the @import similar to what is at the top of the css file, `site-plan-rafting.css`. This line lets our page know where to find the fonts needed. Change the example font names there to your fonts. Or just replace the @import statement that is there, with the @import statement you got from Google Fonts. Notice that if your font name has any spaces in it you should replace them with '+'. Look at my example:

The @import that is already in the site plan has two fonts from Google: **IM Fell** **French Canon SC** and **Lato**

```
@import url('https://fonts.googleapis.com/css2?family=IM+Fell
```

Notice in the @import my first font got changed to: **IM+Fell+French+Canon+SC** (spaces replaced with +). My second font was a simple one word font to it was added just as it was with a **|** (shift + \) separating my fonts.

You must also change the values of the heading and paragraph font variables to the new font names for you to see the changes on your page. Replace the "IM Fell French Canon SC" value of `--heading-font:` to your headline font. Remember if it has spaces in it, use quotes. Also change the 'Lato, Helvetica, sans-serif' value of `--paragraph-font:` as well to your paragraph font.

## Color your fonts

Often you want your text to appear on a colored background. Let's pick some colors from your palette that will work well together.

See if there are two colors in your scheme that can be used as text color or background color where the text remains readable (you need a high contrast between the colors). Black and white are always fair game even if they aren't in your color scheme. You can always visit [contrast-ratio.com](https://contrast-ratio.com) to make sure your text and the background of the text have a passing ratio.

Switch to the **site-plan-rafting.css** file find the lines that look like this:

```
--headline-color-on-white: black; /* headlines on a white background *  
--headline-color-on-color: #FFFFFF; /* headlines on a colored backgroun  
--paragraph-color-on-white: black; /* paragraph text on a white backgro
```

```
--paragraph-color-on-color: #FFFFFF; /* paragraph text on a colored background
--paragraph-background-color: #000000;
```

Change the `--paragraph-background-color` and then `--headline-color-on-color` and `--paragraph-color-on-color` values. If done correctly you should see an example of your colors together in the Preview of your page on Live Server. If any of the text is hard to read, then try different colors.

## Style your navigation

Finally from your *color palette* choose the colors you would like to use for your navigational links (normal, hover, and backgrounds) and enter those colors in the `site-plan-rafting.css` file. Hover over the menu option to make sure the hover colors look good too.

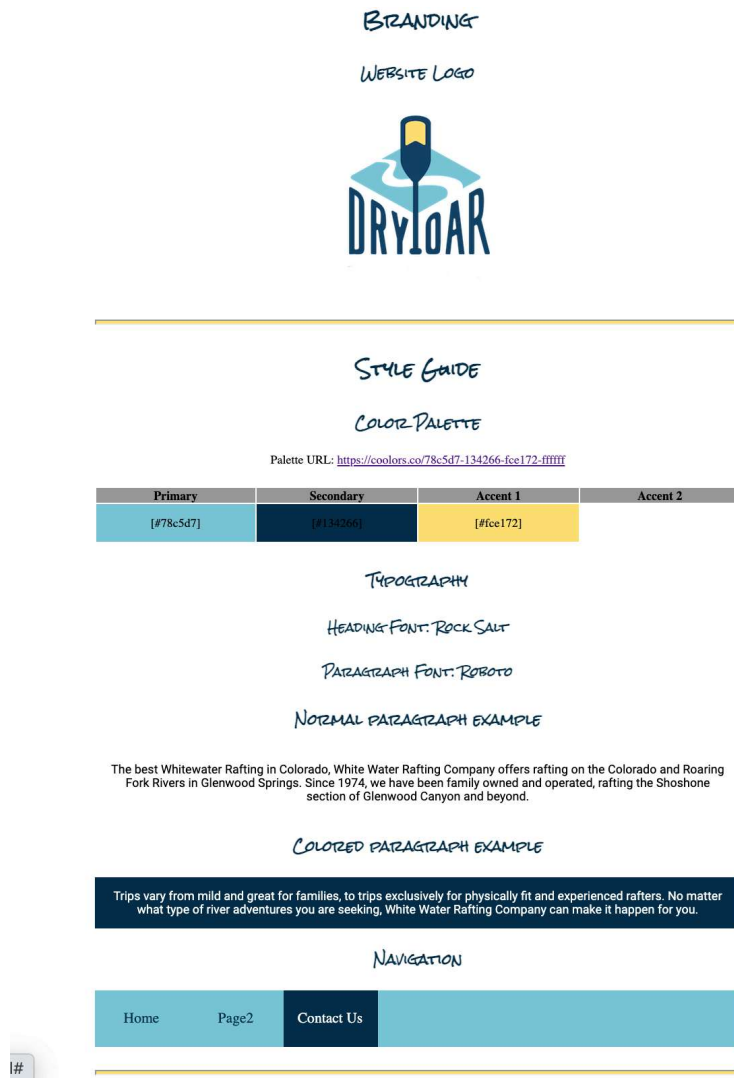
## Site Map & Wireframes

Under the Navigation section under Site Map, you'll notice that there is a page called [Page2]. This will eventually be replaced with a more meaningful page name. Since we haven't decided on this page yet, you can leave that as it is for now. But later we will be adding a child page to our rafting site, and we can come back and change it then.

The last section has a wireframe. The wireframe shows what the home page will look like, or in other words, how it will be laid out. We will not be changing anything in this area, just review it. In the following weeks, you will use the wireframe provided to help you to create the home page for the site. Later in the semester, you will create a wireframe for the child page.

## Submit to Canvas

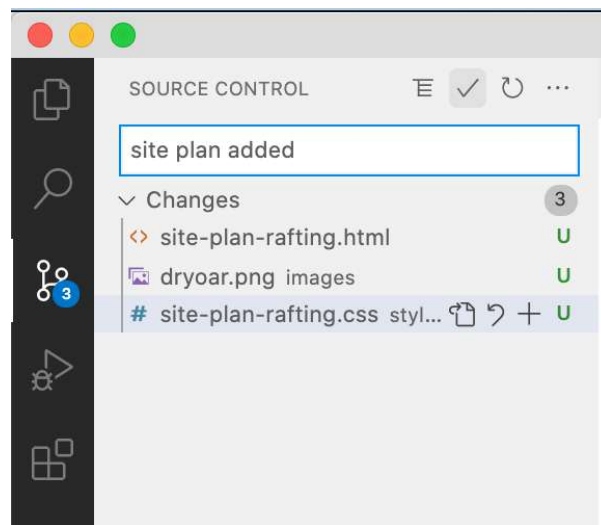
Check your site-plan-rafting.html in VSCode using Live View to make sure it looks good.



Check your site plan using Live Server

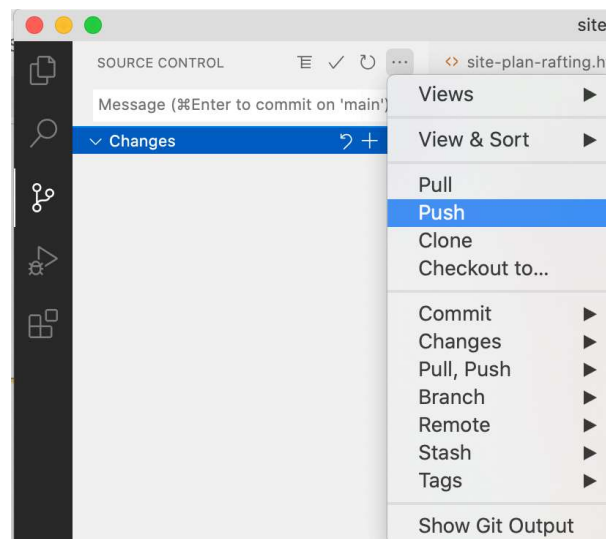
All the colors should be from your color palette and the fonts you chose should be showing. Also make sure all the [ ] square brackets have been replaced with new content.

Let's review how to commit and push your files to GitHub so they can be viewed live on the Internet. Select the Source Control icon and you should see two or three files that have changed. Type in a message like I have here 'site plan added' and select the check mark to commit your new files.



Type the commit message and commit

Then select the ... three dots to bring up the menu and choose 'Push' to send your changes to GitHub



Push the changes to GitHub

Now if you had GitHub Pages all set up, in a few minutes you should see your site plan at [https:// yourgithubaccount .github.io/wdd130/wwr/site-plan-rafting.html](https://yourgithubaccount.github.io/wdd130/wwr/site-plan-rafting.html)

You will submit this URL or web address of your site-plan-rafting.html