

# Lesson 8: Creating a Page Layout

## Welcome Back!

If you've been enjoying this course so far, you're really going to have fun with this lesson. You're going to learn about **page layout**—what it means and how it's done in Web pages. Many people think page layout is one of the most exciting parts of creating Web pages.

With the index.htm and recipe.htm pages you created, you got some hands-on practice creating different kinds of design elements using HTML tags and CSS. While they're perfectly acceptable as Web pages, most websites actually use a page *layout* that makes it easy to recognize what site you're in—and easy to get from one page to the next within the site (assuming the site has more than one page).

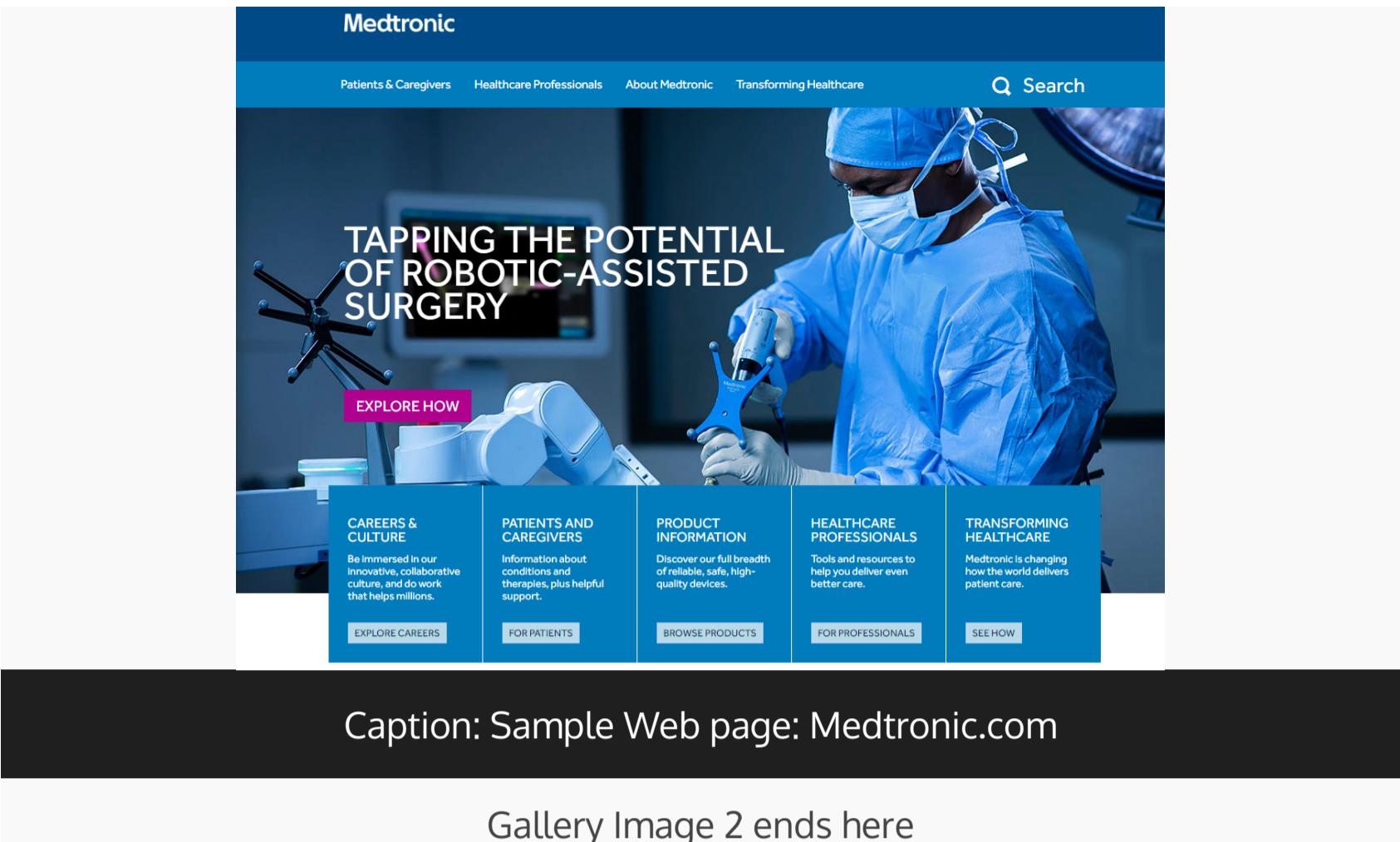
Gallery Image 1 starts here

The screenshot shows the Itron website homepage. At the top, there is a dark navigation bar with links for North America, Blog, Careers, Investors, Sign In, Contact, and a search icon. Below the navigation bar is the Itron logo. The main header features the text "WHY ITRON?" and "We're proven." Below this, there is a paragraph of text: "We have decades of experience, delivering innovative, secure solutions for utilities and cities to 8,000+ customers in more than 100 countries. Our devices, networks, software and services have all been proven at scale—in some of the least hospitable environments on earth. And, our vast partner ecosystem ensures you always have the broadest avenue to innovation." Below the text is a banner with four colored boxes: blue (GAS), green (ELECTRICITY), teal (WATER), and red (SMART CITIES). Each box contains an icon and a brief description. At the bottom of the screenshot is a black caption box containing the text "Caption: Sample Web page: [itron.com](http://itron.com)".

Caption: Sample Web page: [itron.com](http://itron.com)

Gallery Image 1 ends here

Gallery Image 2 starts here



Gallery Image 2 ends here

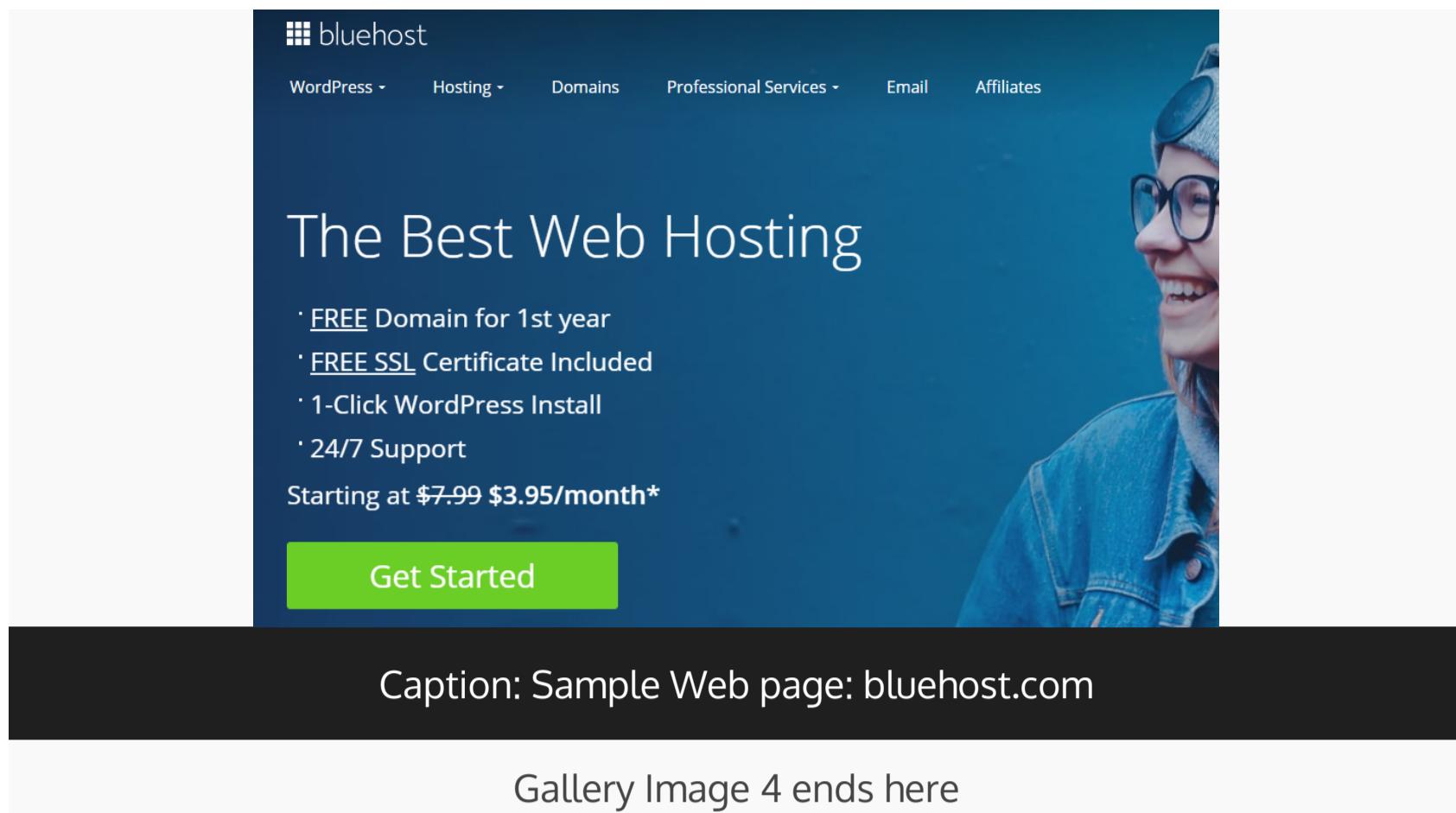
Gallery Image 3 starts here

The image shows the homepage of Lou's Records' website. At the top, there is a navigation bar with links for "HOME", "ABOUT", "CONTACT", "NEWS & EVENTS", "SOCIAL", "SHOP", and "DOWNLOAD STORE". In the center is the Lou's Records logo, which is a circular emblem with the text "LOU'S RECORDS" and "ENCINITAS, CALIFORNIA". Below the logo is a photograph of the interior of a record store. The store is filled with shelves of vinyl records, and there are various posters and signs on the walls, including one of a man's face. A sign above the entrance reads "USED: NEW ARRIVALS". A "TOP TITLES" sign hangs from the ceiling. The floor has a polished wood finish. A black banner at the bottom contains the text "Caption: Sample Web page: lousrecords.com".

Caption: Sample Web page: lousrecords.com

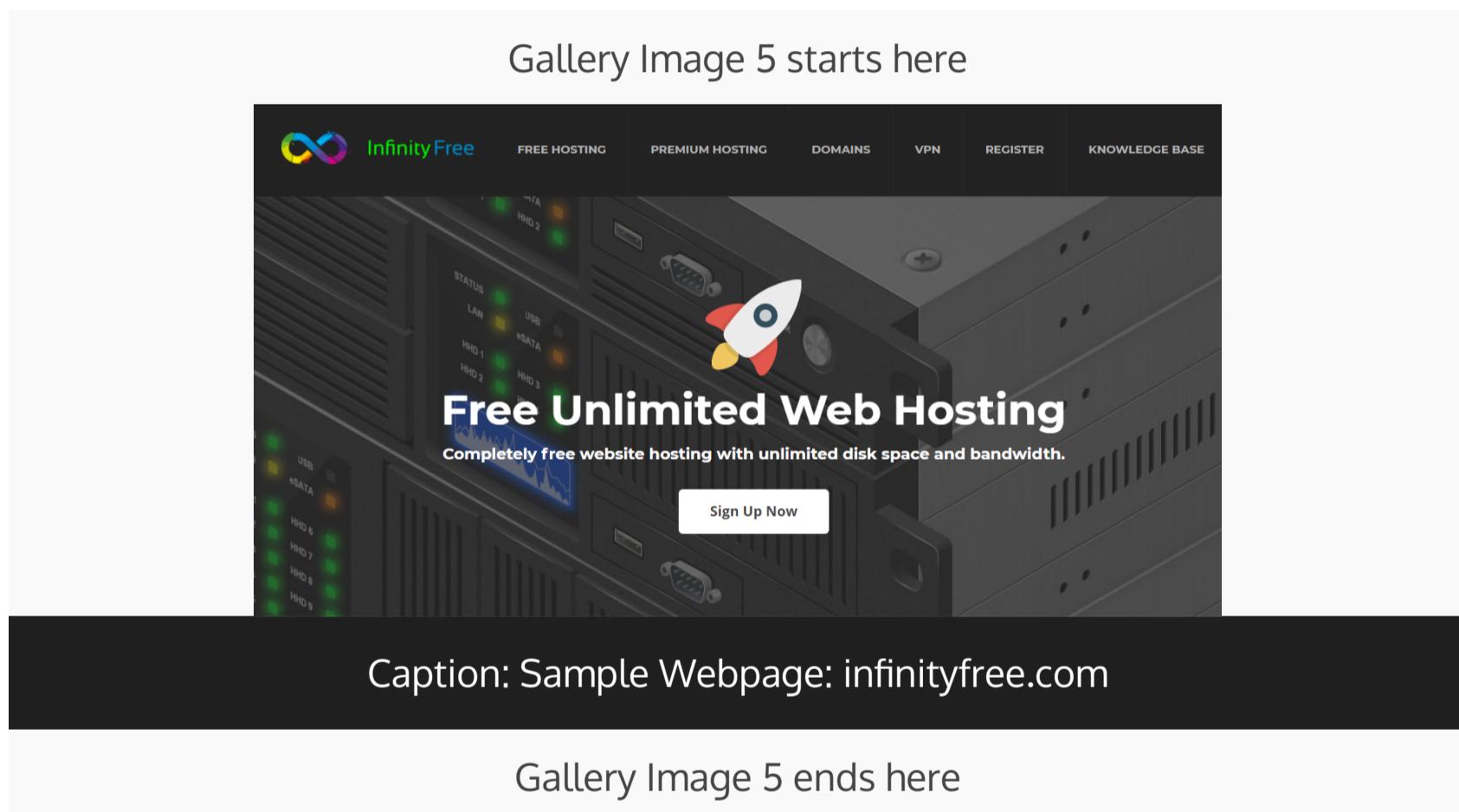
Gallery Image 3 ends here





Caption: Sample Web page: bluehost.com

Gallery Image 4 ends here



Caption: Sample Webpage: infinityfree.com

Gallery Image 5 ends here

Even though these pages don't really look alike, they're organized in similar ways:

- **Site Logo and Title:** Up near the top is the logo, title, or perhaps both. This information is usually aligned left, but it can also be aligned center or right. This area is referred to as the *page header* (or *branding bar*). Studies have shown that the first place most people look when viewing a page is the top left corner. And so the header often contains the name of the site, or perhaps a logo if it's a site for a business that has a logo.
- **Site Menu:** Below, or perhaps to the left or right, of the logo you'll find the *navigation bar* which provides links to other pages within the site. If you click those links, sub-menus may appear with more options.

- **Page Content:** Then the **main content** (or **content area**) for the page typically appears below the menu. This section can contain anything—text, pictures, lists, tables—whatever it takes to best convey the information you're trying to convey. Unlike the page header and links, which are typically the same on any page within the site, the main content section is unique to each page.

So you can consider each web page as having at least these three sections.

## Your Online Course and Lessons

The stuff at the top of this page that you are reading right now (assuming you're looking at the online lesson) is the same as all the other lessons and chapters you've seen. Only this part that you're reading now...the main content of the page...is unique to each page.



Let's get started creating a page layout!



### Before you get started...

Let's remove some current styling so we can explore some new ways to style things.

### Here are the Steps

1. Open index.htm in your editor.
2. Go to your internal style sheet (the code between the `<style>...</style>` tags).
3. Remove the **text-align** and **background-color** from the `body{}` style rule.
4. Next, go to the content between your `<body>...</body>` tags.
5. Remove `style="text-align:center;"` in the `<body>` tag, if you haven't already.
6. Then remove `style="text-align:left;"` from the two `<p>` tags.

## 7. Save your changes.

The code near the top of your index.htm file should now look like this:

```
<!DOCTYPE html>
<html>
<head>
<!-- Title in browser window -->
<title>My Website</title>
<!-- Start of internal style sheet -->
<style>
body {
color: #2d3b71;
}
h1{
color: #64001f;
}
</style>
<!-- End of internal style sheet -->
</head>
<body>
<!-- Main page title -->
<h1>Welcome to My Site</h1>
<p></p>
<p>This is text on my home page. I am a sentence that contains some <strong>boldface</strong> text and some <em>italic</em> text.</p>
<p>This is a practice paragraph that contains multiple sentences. A longer paragraph like this will allow me to see <em>word wrap</em> in action in my Web browser. Word wrap means that the text will wrap to fit the width of the browser window. So long as the browser window is a reasonable width, text will not shoot off past the right edge of the window forcing me to scroll to the right to see it. The wrapping occurs at spaces between words at the ends of lines. That prevents any individual word from being split across two lines.</p>
<p>
<a href="recipe.html">See My Hot Dog Recipe</a>
</p>
</body>
</html>
```

If you look at the page in a browser now, you will have lost some of your styling. Don't worry about that. What matters is that you still have the content of the page.

# Common Layout Tags

Modern Web pages are often organized into a page layout that consists of multiple sections or page divisions. HTML5 contains tags to identify these common parts of a page. The tag pairs we'll be using in this lesson, and what they represent, are summarized below.

Tags	Purpose
<header>...</header>	Used to contain content that is usually displayed at the top of each page.
<nav>...</nav>	Navigation, displays links to other pages within the site.
<main>...</main>	The main content of the page, and is unique to each page.
<div>...</div>	This creates a division (or section) within a page.

Common Layout Tag Pairs

When added to a page, each tag pair just starts a new **block** of content that can contain any text and tags you wish. In other words, the tags listed in the table above don't display any content of their own on the page. They are just there to indicate where the page header, nav, main and other sections begin and end.

We'll use that as our basic starting point, and now let's add a page header.

## Creating a Page Header

### Adding a Header

A page header typically starts at the top of each page. As you may recall, the content that's visible in a web browser starts just below the <body> tag in the code. So let's start by adding header tags just under the <body> tag in index.htm.

### Here are the Steps

1. Open index.htm in your editor.
2. Just below the <body> tag, add the following comment:

```
<!-- Main page header -->
```

3. Press ENTER and then type or copy/paste in <header> tags:

```
<header>  
</header>
```

## Note



If you were to save your changes and look at the page in a browser, you would see no difference at all. Why? Because the header tags just indicate where you want the header to appear. You haven't put anything into the header yet.

4. Any content that should show in the header goes between the <header>...</header> tags, so let's add some content now. So, go ahead and put the cursor between the <header>...</header> tags.

5. Add the page title *Page Header* using h1 tags.

6. Now, go to the <h1> and </h1> tags below the </header> tag (surrounding *Welcome to My Site*) and change them to <h2>...</h2> tags. You don't want this heading to be the same size as the new page header.

7. Save your changes.



## Sneak Peek

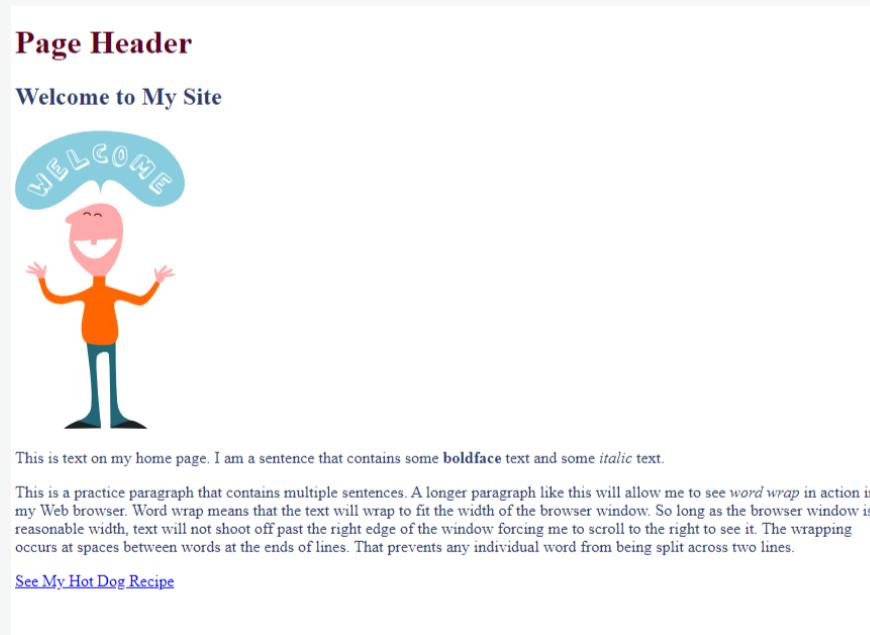
Here is what the top of your index.htm page should look like now.

```

<!DOCTYPE html>
<html>
<head>
<!-- Title in browser window --&gt;
&lt;title&gt;My Web Site&lt;/title&gt;
<!-- Start of internal style sheet --&gt;
&lt;style&gt;
body {
color: #2d3b71;
}
h1 {
color: #64001f;
}
&lt;/style&gt;
<!-- End of internal style sheet --&gt;
&lt;/head&gt;
&lt;body&gt;
<!-- Main page header --&gt;
&lt;header&gt;
&lt;h1&gt;Page Header&lt;/h1&gt;
&lt;/header&gt;
&lt;h2&gt;Welcome to My Site&lt;/h2&gt;
</pre>

```

In the web browser, you can see the header up top and the page title below it. It doesn't look like much yet, because we haven't done much styling. But it's a good start.



## Styling the Header

Our header is in place, but right now it's just a line of text. So, it doesn't really look like a page header. Typically, a page header stands our more than that. Let's add some styling to our page header to make it look more professional.

## Here are the Steps

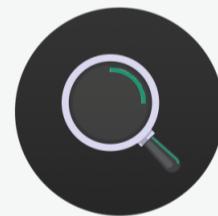
1. Open index.htm open in your editor.
2. Put an empty style rule for the header between the body{} and h1{} style rules, it should look something like this:

```
header {  
}  
}
```

3. Let's give the header its own background color. Between the curly braces for the header style rule, add the following code:

```
background-color: #2d3b71;
```

4. Save those changes and view the page in a browser.



## Have a Closer Look

Now you can see the header near the top of the page. However, the text is really hard to read. The color we chose for the text and background don't have enough contrast between them.

The screenshot shows a web browser window. At the very top is a dark blue header bar with the text "Page Header" in white. Below this is a white content area. In the content area, there is a cartoon illustration of a smiling orange character with arms raised, standing under a blue speech bubble that says "WELCOME". Below the character is a paragraph of text:  
**Welcome to My Site**  
This is text on my home page. I am a sentence that contains some **boldface** text and some *italic* text.  
This is a practice paragraph that contains multiple sentences. A longer paragraph like this will allow me to see *word wrap* in action in my Web browser. Word wrap means that the text will wrap to fit the width of the browser window. So long as the browser window is a reasonable width, text will not shoot off past the right edge of the window forcing me to scroll to the right to see it. The wrapping occurs at spaces between words at the ends of lines. That prevents any individual word from being split across two lines.  
[See My Hot Dog Recipe](#)

## Remember

As you may recall, we changed the heading that's on the page to an h2, so it's different in size from the page header. That's the heading we wanted in that dark red color, so let's change the h1 style rule so that it's styling the h2 rather than the h1 heading.



5. Go to the h1{} style rule, and change h1 to h2.
6. Then go to the header{} style rule, and apply white as the text color, using the following code:

```
color: white;
```

7. Save your changes.



## Sneak Peek

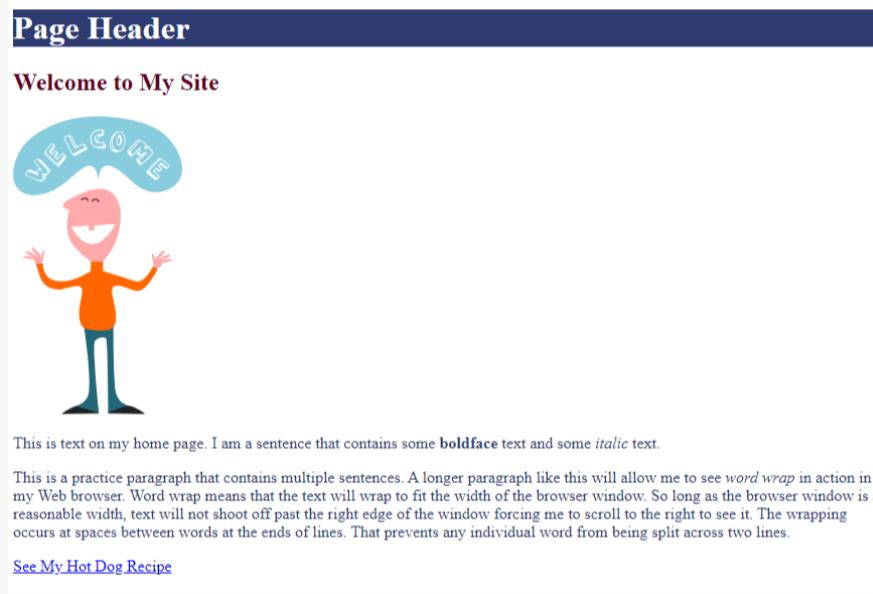
Your style rules should now match those below with the new color: property and h1 style rule changed to h2:

```

<style>
  body {
    color: #2d3b71;
  }
  header{
    background-color: #2d3b71;
    color: white;
  }
  h2{
    color: #64001f;
  }
</style>

```

In the browser things look a little better. Not great...but better. We just have a bit more styling to do.



## Formatting Your Header

### Changing Page Margins

One thing that may not be readily apparent is that there is some empty space above, and next to, the page header. This is because your header is nested inside the body element. That extra space around the edges of the header is the ***page margin*** which is just empty space to keep the text in a page from touching the edges of the browser window.

It's hard to tell with the white background, so let's temporarily change the background color of the body to *yellow*.

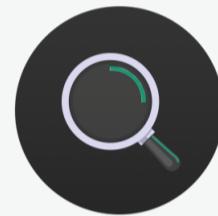
### Here are the Steps

1. Open index.htm in your editor.

2. Add a background color of yellow to the body {} style rule like this:

```
body {  
    color: #2d3b71;  
    background-color: yellow  
}
```

3. Save your changes, and look at the page in a browser.



## Take a Closer Look

You can now see the page background (yellow) above and to the left of the page header.

This is text on my home page. I am a sentence that contains some **boldface** text and some *italic* text.  
This is a practice paragraph that contains multiple sentences. A longer paragraph like this will allow me to see *word wrap* in action in my Web browser. Word wrap means that the text will wrap to fit the width of the browser window. So long as the browser window is a reasonable width, text will not shoot off past the right edge of the window forcing me to scroll to the right to see it. The wrapping occurs at spaces between words at the ends of lines. That prevents any individual word from being split across two lines.  
[See My Hot Dog Recipe](#)

We don't really want that extra space there. To eliminate that space change the **margin** property of the body element to zero.

4. Add the following property:value pair to your body {} style rule:

```
margin: 0;
```

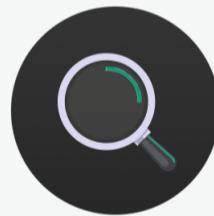


## Important

Make sure you type the number zero, not the letter O, into your body style rule as below:

```
body {  
    color: #2d3b71;  
    margin: 0;  
    background-color: yellow  
}
```

5. Save your changes, and look in the browser again.



## Take a Closer Look

You should now see the space to the left of the header is gone. But there's still a bunch of empty space above it. What's causing that?

Page Header

Welcome to My Site

**WELCOME**

This is text on my home page. I am a sentence that contains some **boldface** text and some *italic* text.

This is a practice paragraph that contains multiple sentences. A longer paragraph like this will allow me to see *word wrap* in action in my Web browser. Word wrap means that the text will wrap to fit the width of the browser window. So long as the browser window is a reasonable width, text will not shoot off past the right edge of the window forcing me to scroll to the right to see it. The wrapping occurs at spaces between words at the ends of lines. That prevents any individual word from being split across two lines.

[See My Hot Dog Recipe](#)

Oddly enough, that empty space above the page header is the margin on the h1 heading. There is no way you would ever know that just by looking. But we can prove it's true by removing the default margins from h1 headings.

6. In your code editor, put the cursor just above the h2{} style rule in your style sheet.

7. Type in this style rule:

```
h1 {  
margin: 0;  
}
```

8. Save your changes.



## Sneak Peek

So now your style rules should look like this:

```
<style>  
body {  
color: #2d3b71;  
margin: 0;  
background-color: yellow;  
}  
header{  
background-color: #2d3b71;  
color: white;  
}  
h1{  
margin: 0;  
}  
h2{  
color: #64001f;  
}  
</style>
```

If you look in the browser, you'll see that the empty space above, and to the left of, the page header is now gone.



So that's good. But now the text is right up against the edge of the screen. We'll deal with that a little later.

For now, let's talk about adding a logo.

## Adding a Logo

If you work for a business that has a logo, or you create a business and its logo, chances are you're going to want to show that logo in the top left corner of every page. You probably don't have a logo handy, so here is one you can use for practice.



## Download

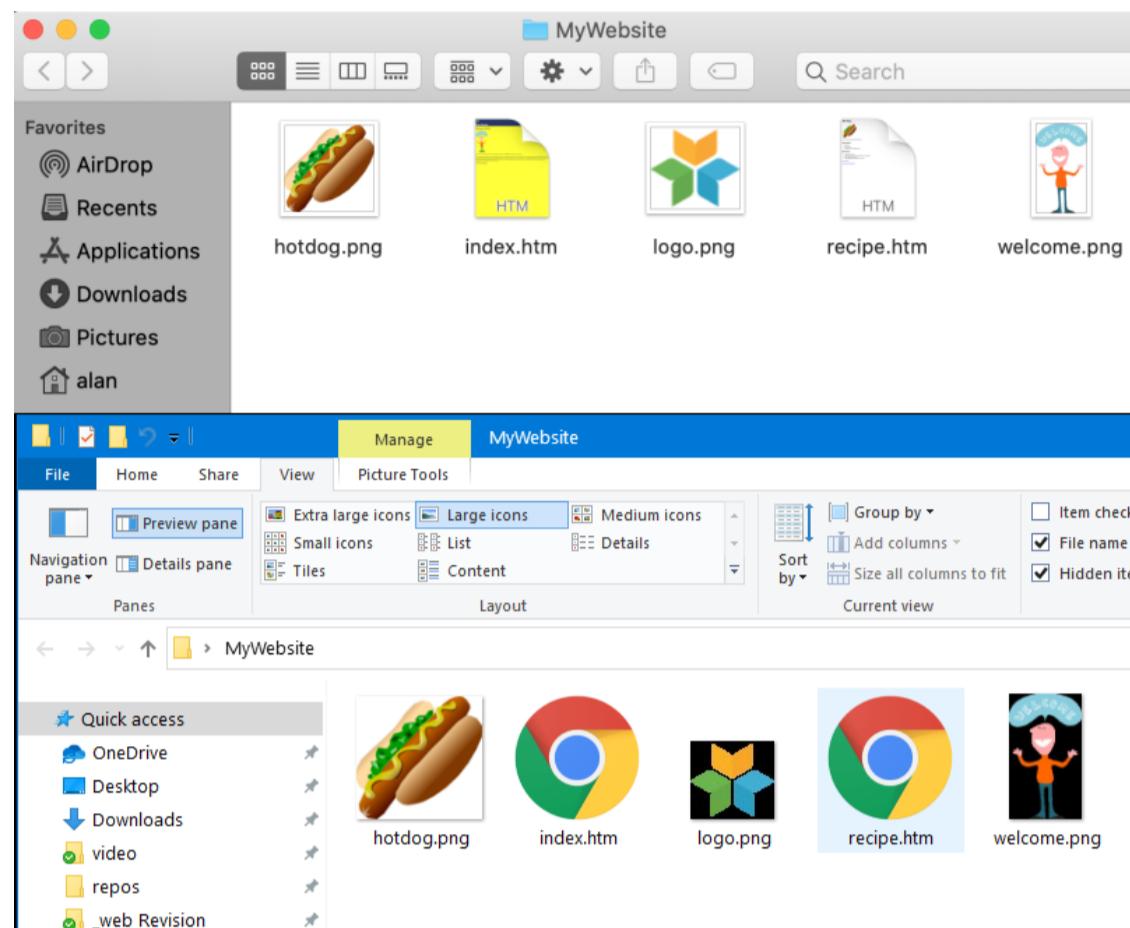
You should be able to right-click this image and choose **Save Picture As** or **Save Image As** depending on your web browser. Make sure you save it in your *MyWebsite* folder and change the file name to *logo.png*.



However, if you have any issues, below is the link to the ZIP file that contains the *logo.png* file you'll need to work with. All you have to do is click the link below to download the ZIP file, save it to your *MyWebsite* folder and extract the file.

[Download Lesson-08\\_Logo.zip](#)

Once you've saved that image file in to your MyWebsite folder and named it properly, you should be able to see its icon in your MyWebsite folder, perhaps like one of these images, depending on your operating system and view settings.



Of course, just putting a graphic image file in your website's folder doesn't make it magically appear on the page. The logo is a graphic image, so you need to add an <img> tag to your header to show the logo.

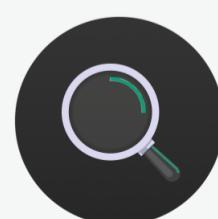
## Here are the Steps

1. Open index.htm in your editor.
2. Place your cursor just to the left of the <h1> tag in the header.
3. Type, or copy/paste, the img tag below:

```

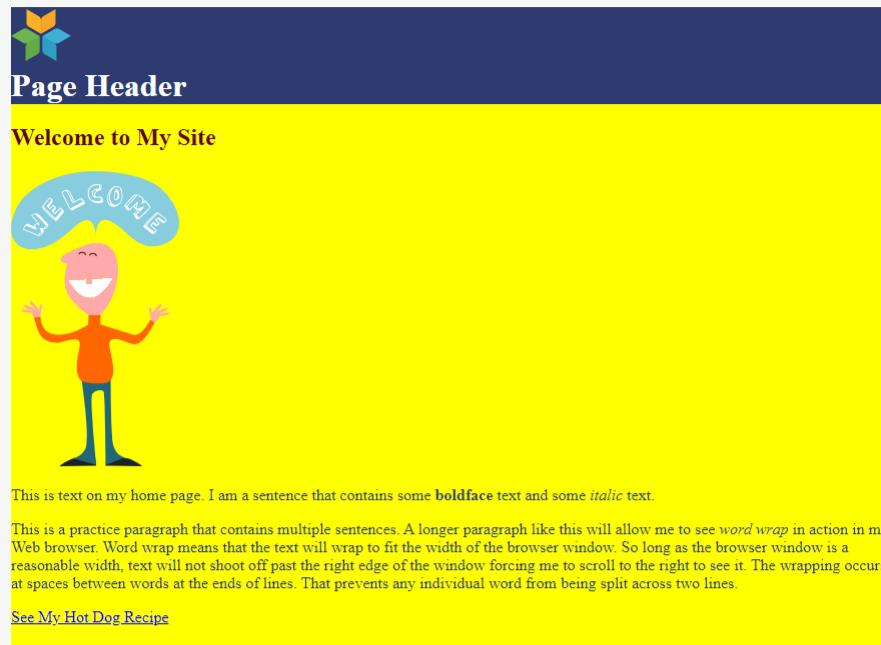
```

4. Save your changes and view the page in a browser.



# Take a Closer Look

Right now, the logo is above the <h1> heading, which isn't really what we want.



To make the logo image float to the left of the header text, we're going to add some inline CSS styling.

5. Place your cursor after the src="logo.png" in the img tag, but before the >, add a space and type:

```
style="float:left;"
```

6. Save your changes.



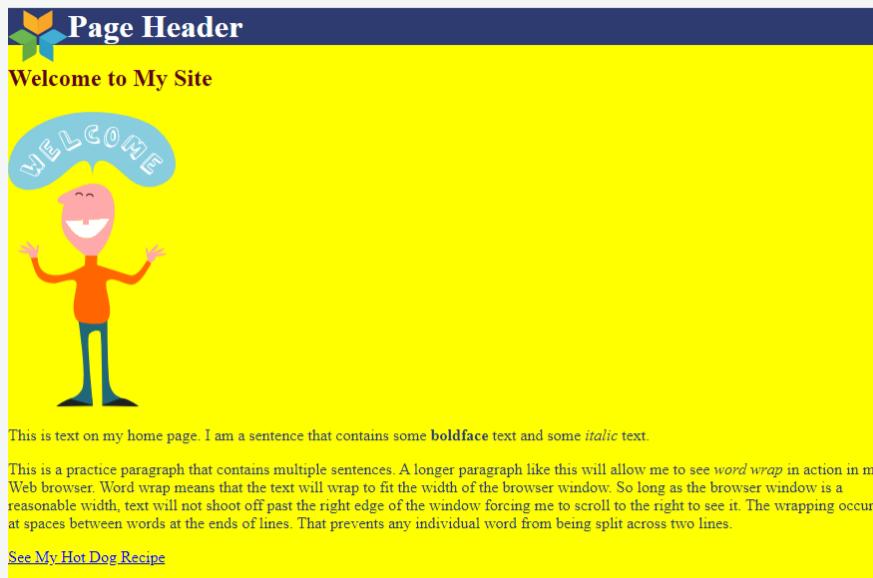
## Sneak Peek

Your header HTML code should look like this:

```
<header>

<h1>Page Header</h1>
</header>
```

Now if you look at the page in a web browser, the logo is floating to the left of the <h1> heading. Not exactly what we want, but we're getting there.

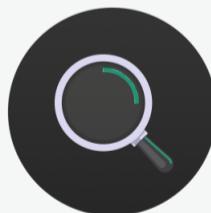


## Adjusting Height

The header would look better if the logo wasn't sticking out past the bottom. One way to deal with that would be to make the header as tall as the image. Or perhaps, decide how tall we want the header to be, then adjust the height of the logo image to fit. Let's try that approach. We'll use the CSS height: property to do this. and the logo about 58 pixels tall.

## Here are the Steps

1. Open index.htm in your editor.
2. Make the header 60 pixels tall, by add `height:60px;` to the `header{}` style rule.



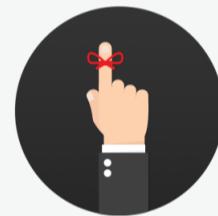
## Take a Closer Look

So now your header style rule looks like this:

```
header {  
background-color: #2d3b71;  
color: white;  
height:60px;  
}
```

3. Next, make the logo 58 pixels tall by adding the following inline CSS style to the img tag:

```
height: 58px;
```



## Remember

Your height property:value pairing should look like this be placed between the semi-colon and the closing quotes:

```

```

4. Save your changes.

At this point your logo should appear to the left of your Page Header text, and it should no longer overhang into the yellow body content area.

Welcome to My Site



This is text on my home page. I am a sentence that contains some **boldface** text and some *italic* text.

This is a practice paragraph that contains multiple sentences. A longer paragraph like this will allow me to see *word wrap* in action in my Web browser. Word wrap means that the text will wrap to fit the width of the browser window. So long as the browser window is a reasonable width, text will not shoot off past the right edge of the window forcing me to scroll to the right to see it. The wrapping occurs at spaces between words at the ends of lines. That prevents any individual word from being split across two lines.

[See My Hot Dog Recipe](#)

## Adding Alt Text

Any time you put an img tag into a web page, you should include **alt= "text"**, which contains text that's read aloud by screen readers. In this case, the image is just a logo so it's really more decorative in nature. It's not a complex illustration, that provides important information you need to describe. In cases like this, a screen reader for the blind doesn't really need to say anything about this image. However, if you don't want the screen reader to say anything about the image, you should add the following attribute to your img tag:

```
alt=""
```

That's two quotation marks, with nothing in between. This let's the screen reader know that it can skip the image. It says: "don't say anything about this image".

Okay, so now your HTML code for your header should look like this:

```
<header>

<h1>Page Header</h1>
</header>
```

## Aligning Your Header

Right now, our header title text *Page Header* is aligned to the top of the page. However, it might look better if the title were centered vertically next to the logo and had a little more space between. To center the text vertically we can set the *line-height* of the header (which is the height of text inside the header) so that it's equal to the height of the header itself. This will make the text in the header look vertically centered.

## Remember

We set the height of the header is 60px in the `header{}` style rule. So we just have to set the line-height to the same value.



Let's give it a try!

## Here are the Steps

1. Open `index.htm` in your editor.
2. Add *line-height: 60px* to your `header{}` style rule, as below:

```
header {  
background-color: #2d3b71;  
color: white;  
height:60px;  
line-height:60px;  
}
```

3. Save your changes.



## Sneak Peek

Now in the browser, the text is more vertically centered.

Welcome to My Site

This is text on my home page. I am a sentence that contains some **boldface** text and some *italic* text.

This is a practice paragraph that contains multiple sentences. A longer paragraph like this will allow me to see *word wrap* in action in my Web browser. Word wrap means that the text will wrap to fit the width of the browser window. So long as the browser window is a reasonable width, text will not shoot off past the right edge of the window forcing me to scroll to the right to see it. The wrapping occurs at spaces between words at the ends of lines. That prevents any individual word from being split across two lines.

[See My Hot Dog Recipe](#)

The text in the header is still nearly touching the logo. A little space there might look better, right? We'll discuss how to add space next.



## Before we move forward...

Let's get rid of that yellow background color for the page, since we don't really need it anymore. To do this, simply remove `background-color:yellow;` from the body {} style rule, so that it looks like this:

```
body {  
color: #2d3b71;  
margin: 0;  
}
```

## Adding Non-Breaking Spaces

You may have noticed that the text in our header is practically touching the logo image. Maybe you could put a bunch of spaces in the code to add some space? Of course, you can use the spacebar, but for multiple spaces the spacebar doesn't behave as you'd expect.

Simply using the spacebar, the most you'll ever get is one space. But why? Well, if you remember, spaces are used to indent code. So, the browser assumes extra spaces are just code indentations, which means it doesn't display those extra spaces in the browser.

When you want to force a space and ensure the code doesn't ignore it as a code indent, you want to use the ***non-breaking space*** special character. To add a non-breaking space, you type the following code:

```
&nbsp;
```

This special character code translates into a blank space. So this...

```
&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
```

...appears as eight blank spaces in the browser.

Let's break down this code so you understand it:

- & – This lets the code know you are inserting a special character.
- nbsp – This stands for non-breaking space.
- ; – This marks the end of the special character code.

## Important

Make sure you don't put any spaces in between these characters!



Of course, it gets a bit messy having thousands of `&nbsp;` characters in your code, so you probably won't use it for all styling. But it works fine for adding one or two spaces to a header or other text.

Let's try it out!

## Here are the Steps

1. Open index.htm in your editor.
2. Place your cursor just after the `<h1>` tag in the header.

3. Type &nbsp; so that special character is between the h1 tag and the first letter in the title, like this:

```
<h1>&nbsp;Page Header</h1>
```

4. Save your changes and take a look in the browser.



## Sneak Peek

Now, the header should show some additional space between the logo and the *Page Header* title.

**Welcome to My Site**

This is text on my home page. I am a sentence that contains some **boldface** text and some *italic* text.

This is a practice paragraph that contains multiple sentences. A longer paragraph like this will allow me to see *word wrap* in action in my Web browser. Word wrap means that the text will wrap to fit the width of the browser window. So long as the browser window is a reasonable width, text will not shoot off past the right edge of the window forcing me to scroll to the right to see it. The wrapping occurs at spaces between words at the ends of lines. That prevents any individual word from being split across two lines.

[See My Hot Dog Recipe](#)

Now that you have your header in place, it might actually look better if that *Welcome to My Site* text and image were centered.

## Dividing Up Your Content

### Applying Formatting in Sections

Things look a little cramped on the left side of your page. The page would look better if the first text and image under the header were centered. We could do that using inline styles. But it's more efficient, and more common, to elements you want to customize inside their own **division** (or section) using **<div>...</div>** tags and then apply the formatting to the division as a whole.

Once you have your ***div*** (short for division) set up, any styling you apply to the division applies to elements within the division. Let's do it!

## Here are the Steps

To center everything in the division, use the CSS ***text-align:center*** *property:value* pair. Here are the steps:

1. Open index.htm in your editor.
2. Put the cursor just after the </header> tag and press **ENTER**.
3. Add a comment for future reference by typing the following on the new line:

```
<!-- Center page title and image -->
```

### Note

This is optional but will serve as a reminder for you in the future.



4. Press **ENTER** and then add the following open <div> tag for your division:

```
<div style="text-align:center;">
```

5. Move your cursor to just after the <p>...</p> tag that has the img tag for the welcome.png image in it.

6. Press **ENTER**, type in the closing </div> tag.

```
</div>
```





## Take a Closer Look

That code below the header should look like this:

```
<!-- Center page title and image -->
<div style="text-align:center;">
<!-- Main page title -->
<h2>Welcome to My Site</h2>
<p></p>
</div>
```

The indentations are optional. They're just there to make it easier to see where that div begins and ends, because the code inside it is indented.

7. Save your changes and take another look in the browser.



## Sneak Peek

The *Welcome to My Site* heading and welcome.png image should now be centered.



## Welcome to My Site



This is text on my home page. I am a sentence that contains some **boldface** text and some *italic* text.

This is a practice paragraph that contains multiple sentences. A longer paragraph like this will allow me to see *word wrap* in action in my Web browser. Word wrap means that the text will wrap to fit the width of the browser window. So long as the browser window is a reasonable width, text will not shoot off past the right edge of the window forcing me to scroll to the right to see it. The wrapping occurs at spaces between words at the ends of lines. That prevents any individual word from being split across two lines.

[See My Hot Dog Recipe](#)

There is still some room for improvement in our styling, but is a work in progress. We also need to add a navigation bar, so let's get started with that next.

## Building a Navigation Bar

We've added a page header to the home page (index.htm) of our site. Next, we'll add a ***navigation bar***, which is also called a ***navbar*** for short. The navbar for a site typically includes a link to each page in the site. In an exceptionally large site, each link might open a sub-menu to display a group of page links that are similar. This is a section of the site that contains multiple pages. Either way, the purpose of the nav bar is to give the person browsing the site easy access to the main pages (or sections) within the site.

## Adding a Navbar

To create a navbar you add a pair of `<nav>...</nav>` tags and add links to pages in between. The nav tags themselves don't alter the styling of the page. This means that in the browser, the page will look the same with or without those tags in it. The `<nav>` tags work similar to the `<div>` and `<header>` tags, in that they allow you to call out a group of elements and apply custom formatting to them.

## Here are the Steps

1. Open `index.htm` in your editor.
2. Place the cursor after the closing `</header>` tag and press **ENTER** to start a new line.

3. Then add the following comment:

```
<!-- Navigation bar -->
```

4. Press **ENTER** and type `<nav>` to start your navbar.

5. Press **ENTER** twice and type `</nav>` to add some space and close your navbar.

6. Save your changes.

Between the `<nav>` and `</nav>` go all the links that you want to show in the navbar. There's nothing special about those links. They're just the same `<a>...</a>` tags you learned about earlier. Let's go ahead and add them.

## Adding Links to Your Navbar

We only have two pages in our site, so we'll add a link to an external site as a third link, just to flesh out the navbar a bit. Most of the time you'll have more than two to work with, so this will make our working example more like real life.

### Your navbar should include a link to the current page.

That might sound strange at first. Technically, you don't *need* a link to the home page on the home page, because you're already on the home page. However, it is *customary* to put a link to every page in the site, even the current page, on the navbar. It's been the standard for so long that if you don't do it, it may be more confusing than to visitors—even if it's unnecessary.



## Here are the Steps

1. Go back to index.htm in your editor.

2. Place your cursor below the `<nav>` tag.

3. Add a link to the home page, index.htm, as shown below:

```
<a href="index.htm">Home</a>
```

## Remember

The word *Home* is the visible text of the link that shows on the page. The *index.htm* is actual link to the page to which a user is taken when they click the link.



4. Save your changes and open the page in a browser.



## Sneak Peek

You should now see the *Home* link below the header. Clicking that link won't do anything, though, because you're already on the Home page. But don't worry about that right now.



5. Ok, let's go back to index.htm in your editor so we can add more links.

6. Add a link to the recipe.htm page below the link to the home page:

```
<a href="recipe.htm">Hot Dog</a>
```

7. Now let's add another link right after that:

```
<a href="https://allrecipes.com/">All Recipes</a>
```

## Why does this link have https://?

Well, since allrecipes.com is external to the site (meaning its not part of the site we are building) we need to have the https:// in there so that the browser knows it has to retrieve it from Web on a page outside the current site. This is what is referred to as an ***external link***. The other links we created are ***internal links***. They don't need the https:// because they are actually retrieving the page from the site files, rather than the Web.



8. Save your changes.



## Take a Closer Look

At this point your navbar code should look like this:

```
<!-- Navigation bar -->  
<nav>  
  <a href="index.htm">Home</a>  
  <a href="recipe.htm">Hot Dog</a>  
  <a href="https://allrecipes.com/">All Recipes</a>  
</nav>
```

If you look in the browser now, you'll see three links. However, without any formatting, the links don't look like much. So let's look at some ways we can style those links.

## Styling the Nav Bar

The nav tags don't add any styling to the links. The link is colored and underlined because links are styled that way by default. Whether they're in nav tags or not, an unclicked link is typically blue underlined text. This indicates a page to which you've never browsed. The purple color indicates a page you have visited.



## Welcome to My Site



It's a little difficult to tell the links apart, so let's put some extra spaces between them.

## Spacing Your Links

We can use the non-breaking space special character, &nbsp; to add space between the navbar links. Feel free to style to your own taste. But for starters we'll put three non-breaking spaces to the left of each link, as below:

```
<!-- Navigation bar -->  
<nav>  
  &nbsp;&nbsp;&nbsp;<a href="index.htm">Home</a>  
  &nbsp;&nbsp;&nbsp;<a href="recipe.htm">Hot Dog</a>  
  &nbsp;&nbsp;&nbsp;<a href="https://allrecipes.com/">All Recipes</a>  
</nav>
```

Those will translate to more space at the left of each link in the web browser:



## Welcome to My Site



Next we'll add some CSS styling to our nav bar and links.

### Adding CSS Rules for the Navbar

There are many things we can do in CSS to make the nav bar look better. Let's start by giving it a little background color, and maybe a little more height.

### Here are the Steps

1. Open index.htm in your code editor.
2. Move the cursor so it's just in front of the h1{} style rule and press ENTER.

3. Add a style rule for the navbar, like this:

```
nav{  
}
```

4. Inside that style rule, add a background color using the following hex code of #BD905E (or any color you like).

```
background-color: #BD905E;
```

5. To make the nav bar a little more noticeable, lets increase its height by adding this line to the style

```
height: 30px;
```

rule.

6. To vertically center each link in the nav bar, let's set the line height of the text so its equal to the height of the navbar) by adding this line to your nav{} style rule:

```
line-height: 30px;
```

7. Save your changes.

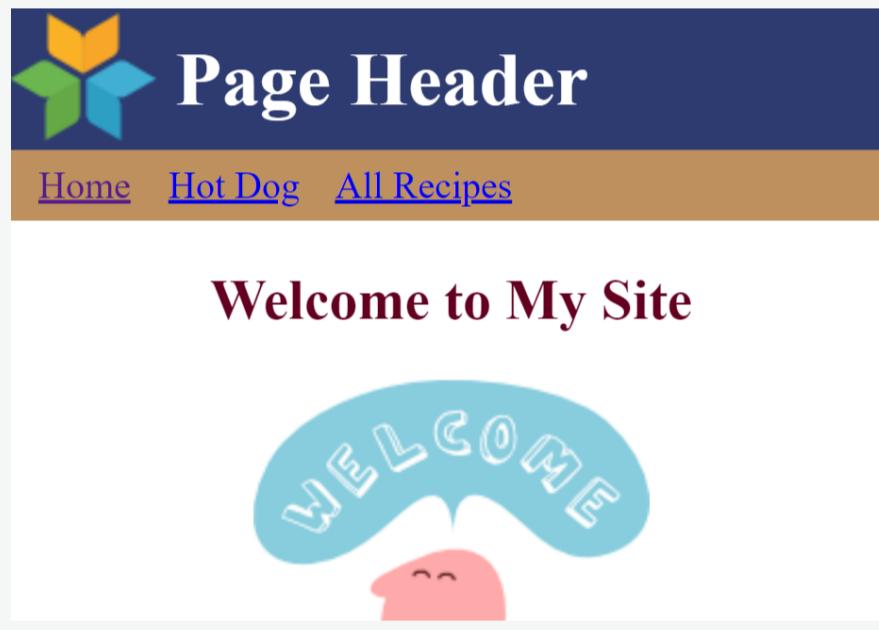


## Sneak Peek

So now if you look at your internal style sheet, the header, nav, and h1 style rules should look like this:

```
header {  
    background-color: #2d3b71;  
    color: white;  
    height: 60px;  
    line-height: 60px;  
}  
  
nav {  
    background-color: #BD905E;  
    height: 30px;  
    line-height: 30px;  
}  
  
h1 {  
    margin: 0;  
}
```

In the browser, things should be starting to look a bit better, but we still have some styling to do.



## Styling Your Links

Inside the navbar, each link is identified by a pair of `<a>...</a>` tags. The style rule for styling links, therefore, needs just the letter "a" as its selector. You can put this one right under the style rule for the header, if you like.

## Here are the Steps

1. Go to index.htm in your editor.
2. Put the cursor below the closing curly brace for the `header{}` style rule and above the `h1{}` style rule.

3. Add an empty style rule for your links:

```
a{  
}
```

4. Normally, links are underlined. However, let's remove the underlines from the links. To do that, you need to set the ***text-decoration*** CSS property to *none*. So, add the following property:value pair to the style rule:

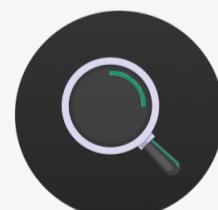
```
text-decoration:none;
```

5. You can also change the color of the links, by using the ***color*** property. We'll change our links to *black*, but you can color things however you like. Add the following property:value pair to the style rule.

```
color:black;
```

6. To make them a little larger, we can use the ***font-size*** property. To make them 15% larger than the default size. To do this we'll set the font-size value to *115%*.

```
font-size:115%;
```



## Take a Closer Look

So now your a{} style rule should look like this:

```
a{  
    text-decoration:none;  
    color:black;  
    font-size:115%;  
}
```

7. Save the page and take a look in the browser.



## Sneak Peek

The navbar should be tan with black text.

This is text on my home page. I am a sentence that contains some **boldface** text and some *italic* text.

This is a practice paragraph that contains multiple sentences. A longer paragraph like this will allow me to see *word wrap* in action in my Web browser. Word wrap means that the text will wrap to fit the width of the browser window. So long as the browser window is a reasonable width, text will not shoot off past the right edge of the window forcing me to scroll to the right to see it. The wrapping occurs at spaces between words at the ends of lines. That prevents any individual word from being split across two lines.

[See My Hot Dog Recipe](#)

You may have noticed that the *See My Hot Dog Recipe* link at the bottom of the page was changed too. It has the same styling as the links in the navbar. While it's perfectly okay to make the styling of links inside the navbar the same as they are outside the navbar. We want our other links to follow the regular link format, so what do we do? The solution is simple, we simply need to tell it to only apply this styling to links *within* the navbar.

## Using Descendant Selectors

You can make the styling for links in the navbar different from other links by using a CSS ***descendant selector***. What that means is that you start the style rule using the selector(s) for the section to which the styling applies (e.g. nav), then you follow that with a space and add the selector(s) for the type of elements you want to style within that block.

For our example, we want to style the links inside the nav section. We've already created the style we want to apply to these links, using the link selector so all we need to do is specify that its for the nav links only. To do this, we simply need to add the nav selector ahead of our link selector like so:

```
nav a{  
}
```

Then the styling in that style rule will apply only to links inside the navbar, not outside if it.

Let's try it out!

## Here are the Steps

1. Open index.html in your editor.
2. Find the a{} style rule and place your cursor just before the letter *a*.
3. Type *nav* and press the **SPACEBAR**.

4. Now your internal style sheet should look like this:

```
<!-- Start of internal style sheet -->
<style>
body {
color: #2d3b71;
margin: 0;
}

header{
background-color: #2d3b71;
color: white;
height:60px;
line-height:60px;
}

nav{
background-color: #BD905E;
height: 30px;
line-height: 30px;
}

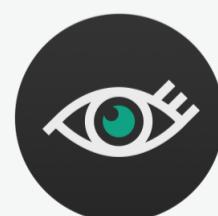
nav a{
text-decoration:none;
color:black;
font-size: 115%;
}

h1{
margin:0;
}

h2{
color: #64001f;
}

</style>
<!-- End of internal style sheet -->
```

5. Save your changes and open the page in your browser.



Sneak Peek

Now, only links inside the navbar have the black, underline-free styling. The link at the bottom of the page should be blue (or purple) with an underline, since it has the original (default) link styling.

The screenshot shows a web page with a dark blue header containing a stylized logo of three overlapping shapes in yellow, green, and blue. To the right of the logo, the text "Page Header" is displayed in white. Below the header is a brown navigation bar with the links "Home", "Hot Dog", and "All Recipes". The main content area has a white background. At the top of the content, the text "Welcome to My Site" is centered in red. Below this, there is a cartoon illustration of a pink-headed character with a wide smile, wearing an orange shirt and teal pants. A blue speech bubble above the character contains the word "WELCOME" in white. Further down the page, there is some sample text: "This is text on my home page. I am a sentence that contains some **boldface** text and some *italic* text." and "This is a practice paragraph that contains multiple sentences. A longer paragraph like this will allow me to see *word wrap* in action in my Web browser. Word wrap means that the text will wrap to fit the width of the browser window. So long as the browser window is a reasonable width, text will not shoot off past the right edge of the window forcing me to scroll to the right to see it. The wrapping occurs at spaces between words at the ends of lines. That prevents any individual word from being split across two lines." At the bottom of the content area, there is a blue link "See My Hot Dog Recipe".

Our page is fine the way it is. There is room across the nav bar to for links to many more pages. If your site is likely to have links to many pages, you may want to leave the styling as it is. However, placing your navbar links on their own line is only one way to do things. With sites that don't have a lot of links, you can save space by putting the navbar links inside the header. Let's look at how to style that.

## Restyling the Navbar

Since our example only has three links, we don't really need to have the navbar on its own line. Let's see how the navbar would look on the right side of the header instead.

## Moving the Navbar

First we'll move the nav bar so its inside the header. An easy way to do that would be to just move the closing </header> tag.

## Here are the Steps

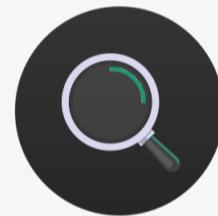
1. Open index.htm in your editor.

2. Move the closing </header> tag so it's below the </nav> tag as below:

```
<!-- Main page header -->
<header>

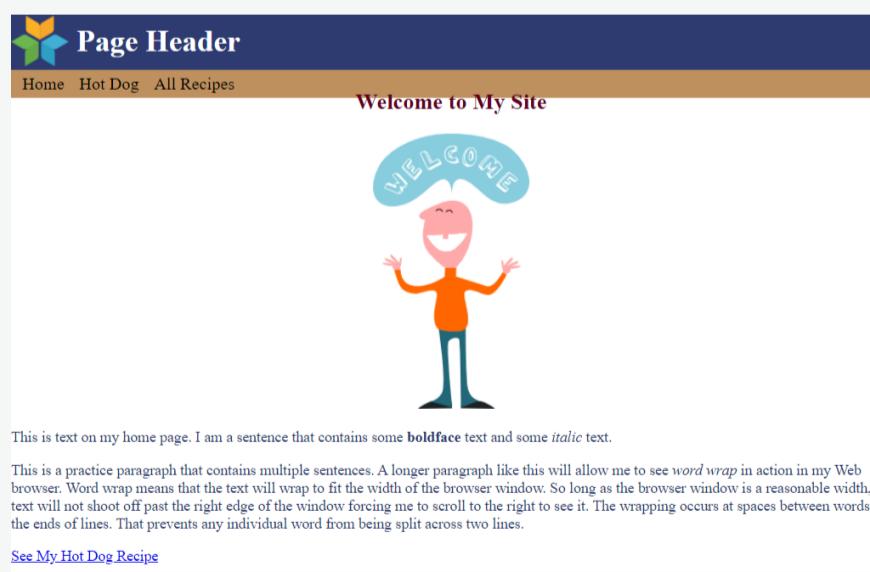
<h1>&nbsp;Page Header</h1>
<!-- Navigation bar -->
<nav>
&nbsp;&nbsp;&nbsp;<a href="index.htm">Home</a>
&nbsp;&nbsp;&nbsp;<a href="recipe.htm">Hot Dog</a>
&nbsp;&nbsp;&nbsp;<a href="https://allrecipes.com/">All Recipes</a>
</nav>
</header>
```

3. Save your changes.



## Take a Closer Look

In the browser the bar would still look like it's below the header, although you may see some wonky formatting depending on your browser. For instance, you may see some overlap of the navbar and the *Welcome to My Site* text.



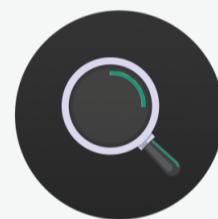
## Resizing the Navbar

To make it fit inside the header, we need to make some room for it and reduce its width. Right now, the logo image floats to the left of the h1 heading. Though you can't tell from looking at the code, that h1 is as wide as the entire page. In order to put anything to right of that heading, we need to float the h1 heading to the left as well.

## Here are the Steps

1. Open index.htm in your editor.
2. Go to the <header> tag, and place your cursor right after the h1 in the <h1> tag.
3. Press the **SPACEBAR** and add a left float style:

```
style="float:left;"
```



## Take a Closer Look

So now the header code should look like this:

```
<!-- Main page header -->
<header>
  
  <h1 style="float:left;">&nbsp;Page Header</h1>
  <!-- Navigation bar -->
  <nav>
    &nbsp;&nbsp;<a href="index.htm">Home</a>
    &nbsp;&nbsp;<a href="recipe.htm">Hot Dog</a>
    &nbsp;&nbsp;<a href="https://allrecipes.com/">All Recipes</a>
  </nav>
</header>
```

4. Save your changes and open/refresh the page in your browser.



## Take a Closer Look

The navbar is as wide as the header, and still has its own background color. So if you were to look at the page in the browser now, you'd see that we're not done. We just have a little more styling to do.

The screenshot shows a web page with the following structure:

- Header:** A decorative logo consisting of three overlapping shapes (blue, green, yellow) is on the left. To its right, the text "Page Header" is displayed in white on a dark blue background.
- Navigation Bar:** A light brown horizontal bar above the main content area contains the links "Home", "Hot Dog", and "All Recipes".
- Main Content:** The main area has a white background. At the top, the text "Welcome to My Site" is centered in a bold, dark red font. Below this, there is a cartoon illustration of a smiling character with a pink head, orange torso, and teal legs. Above the character is a blue speech bubble containing the word "WELCOME" in white, curved text.
- Text:** Below the main content, there is some sample text:

This is text on my home page. I am a sentence that contains some **boldface** text and some *italic* text.

This is a practice paragraph that contains multiple sentences. A longer paragraph like this will allow me to see *word wrap* in action in my Web browser. Word wrap means that the text will wrap to fit the width of the browser window. So long as the browser window is a reasonable width, text will not shoot off past the right edge of the window forcing me to scroll to the right to see it. The wrapping occurs at spaces between words at the ends of lines. That prevents any individual word from being split across two lines.

[See My Hot Dog Recipe](#)

5. Now, you need to revisit your style sheet. Go back to index.htm in your code editor.

6. Adjust the navbar width and float it to the right by adding the following lines to the `nav{}` style rule:

```
width: 250px;  
float: right;
```

7. Save your changes and open/refresh the page in your browser.



## Sneak Peek

The navbar is now only 250 pixels wide and positioned on the right side of the header. But we're not done yet.



## Adjusting the Style Sheet

Now that we've moved our navbar to our header, it would look better if we removed the tan background and made a few adjustments to its position. However, the black text won't work well with the blue we're using in the header, so let's change the nav link text color from black to white so that they stand out better.

## Here are the Steps

1. Go back to index.htm internal style sheet in your editor.
2. Remove the following styles from the nav{} style rule:
  - background color
  - height
  - line-height
3. In the nav a{} style rule, change the link color from *black* to *white*.



## Take a Closer Look

Here is how the CSS code for the nav{} and nav a{} style rules should look now.

```
nav {  
    width: 250px;  
    float: right;  
}  
  
nav a {  
    text-decoration: none;  
    color: white;  
    font-size: 115%;  
}
```

4. Save your changes and open the page in the Web browser.



## Sneak Peek

The nav bar should now just be three white text links over on the right side of the header.



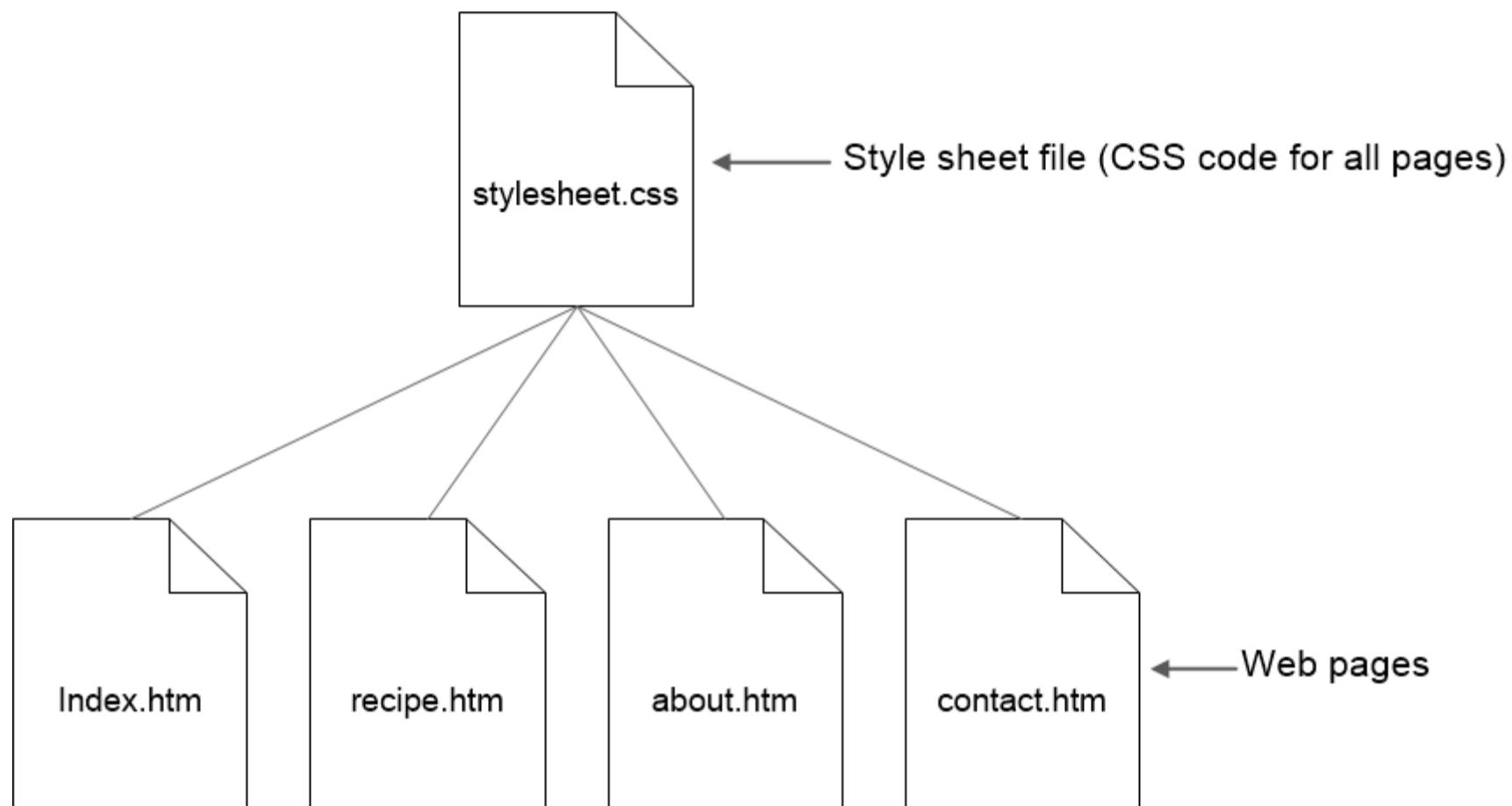
That will do it for the header and nav bar for now. Up next, we're going to transform our internal style sheet to an external style sheet.

## Styling Multiple Pages

Now that we've taken the time to create and style a page header and navbar for our site, it's important that these elements are consistent across all of the site pages. The CSS code will be the same in every page. Rather than repeat the same CSS code over and over in each page, we can put all the CSS code in one file. This is referred to as an *external style sheet*.

## What is an External Style Sheet?

In this lesson we're going to move all the CSS code out to its own separate file. Then, each page in the site can refer to that style sheet, and apply its styles without you having to repeat the same code in each page of your site.



The external style sheet file will not use the .htm extension, instead it will use .css extension because it contains only the CSS code for styling (e.g. `stylesheet.css`). This CSS file can then be linked to each page in a site. Right now, our working site has only two pages: the home page (`index.htm`) and the recipe page (`recipe.htm`). But in real life a web site can have any number of pages. Think of the thousands of pages that make up Amazon or Facebook!



### Talk about a time-saver!

There are two big advantages to externalizing the style sheet:

- **You only have to create the style code once.** Rather than repeating the same style code over and over again in every page of your site, you just link your pages to the style sheet file.

- **You only have to made changes once.** If you want to make a stylistic change, you just make the change in the style sheet file and it filters out to every page in your site.

Consider a huge site that contains 1,000 pages. Would you want to make a stylistic change to the header on every single page? Well, with an external style sheet, you make the change in that one file and your done.

## Coding CSS Files

Before we create an external style sheet, its important to note that they use CSS code exclusively. As you've probably already noticed, CSS is very different from HTML. So, we'll be noting some important differences along the way.

### Adding CSS Comments

We've already discussed the importance of adding programmer comments for yourself. Whether these notes are for you as you are learning, or for future reference when you (or someone else) goes in to update a code file, comments help decipher code.

HTML uses <!-- to start and --> to end comments, but CSS uses /\* to start a comment and \*/ to end a comment. Each comment should look like this:

```
/* Put your CSS comment here */
```

### Important

Never put a space between the slash and an asterisk.



There are two places where you can use CSS comments:

- **Anywhere in an external CSS style sheet.** This is a separate .css file.
- **Anywhere a internal CSS style sheet.** This is the CSS code between the <style>...</style> tags in a web page.

You cannot add CSS comments anywhere else in a web page (.htm or .html file). Now, let's create an external style sheet, and put in a few CSS comments in.

## Creating an External Style Sheet

A external CSS style sheet is a text file, just as every webpage is a text file. Any editor you use to create webpages, can also be used to create CSS style sheets. There are two big differences between a webpage and an external style sheet:

- A style sheet file uses the .css extension, rather than .htm or .html.
- A style sheet contains only CSS style rules, no HTML tags.

Let's go ahead and create a style sheet now.

### Here are the Steps

1. Open your code editor and create a blank page.

2. Type the following CSS comment on the first line:

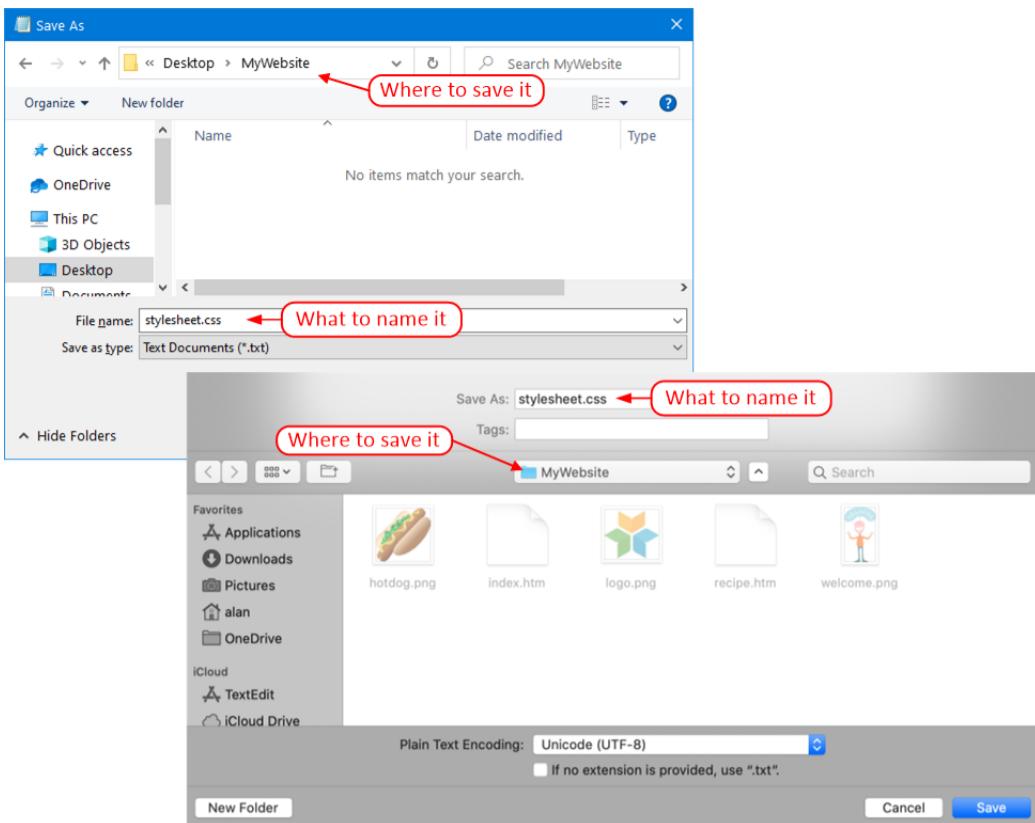
```
/* This is stylesheet.css an external style sheet */
```

3. Choose **File > Save** (or use whatever method you normally use) to save files in your editor.

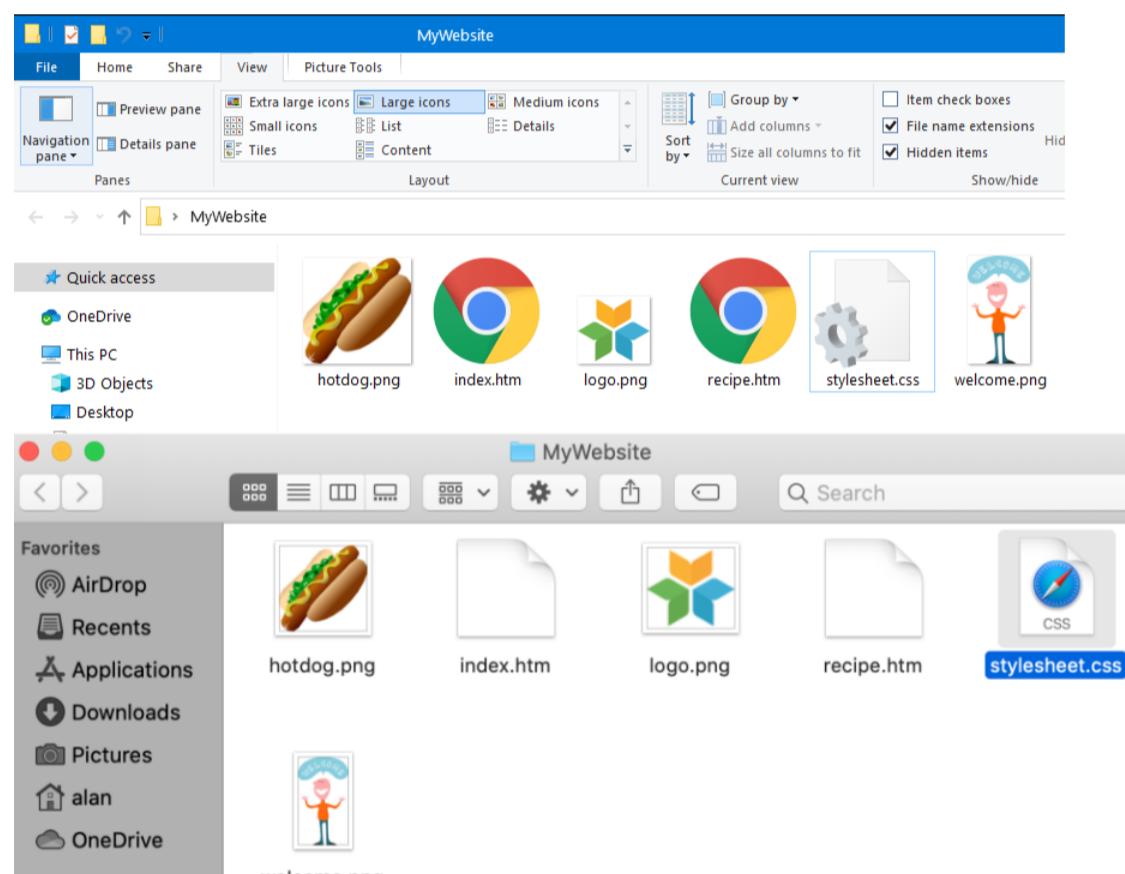
4. Choose your **MyWebsite** folder as the location to store the file.

5. Name the file *stylesheet.css* (no spaces).

6. Click **Save** to save the file.



To verify that you did it correctly open your *MyWebsite* folder, and check that you see the style sheet file *stylesheet.css* is in there.



## Note

Depending on your operating system and computer settings your icons might not look the same as those in the picture, but that's not important. All that truly matters is that:

- The file is in your *MyWebsite* folder.
- The file name is *stylesheet.css*.

This is important because you will need to create a reference to the file within your HTML pages.

## Adding Code to a Style Sheet

Now that you have an external style sheet, you need to add some code. It's the same process as adding code or content to any other file. You can type in the CSS style rules by hand, or you can copy/paste them in from another source.

### Remember

Only CSS code is allowed, no images, no HTML, no content for people to look at.



We're going to take all of the code that's currently in the `index.htm` page's internal style sheet (between the `<style>...</style>` tags) into our new external style sheet.

## Here are the Steps

1. Open `stylesheet.css` in your code editor.
2. Open `index.htm` in your code editor.
3. Copy all the code between the `<style>...</style>` tags in `index.htm`.
4. Then, paste it into `stylesheet.css`.
5. Save your changes to `stylesheet.css`.



## Sneak Peak

The code in your style sheet should now match the following:

```
/* This is stylesheet.css an external style sheet */

body {
    color: #2d3b71;
    margin: 0;
}

header{
    background-color: #2d3b71;
    color: white;
    height: 60px;
    line-height: 60px;
}

nav{
    width: 250px;
    float: right;
}

nav a{
    text-decoration: none;
    color: white;
    font-size: 115%;
}

h1{
    margin:0;
}

h2{
    color: #64001f;
}
```

There is no point opening the style sheet in a web browser, because it doesn't contain any HTML code or content for the browser to visually render. It's simply a set of style rules, nothing more. So, there is nothing to look at. For this style sheet to be useful, we need to link it.

## Linking to an External Style Sheet

### Removing Your Internal Style Sheet

Before we create a link to our external style sheet, we need to get rid of the style rules in *index.htm* since we won't need those once the page is linked to the style sheet. So let's get rid of the original styles now.

Here are the Steps

1. Open *index.html* in your editor.
2. Delete everything from start of the **<!-- Start of internal style sheet -->** comment to the end of the **<!-- End of internal style sheet -->** comment and everything in between.
3. Verify your index.htm page code now matches this:

```
<!DOCTYPE html>
<html>
<head>
<!-- Title in browser window --&gt;
&lt;title&gt;My Website&lt;/title&gt;
&lt;/head&gt;
&lt;body&gt;
<!-- Main page header --&gt;
&lt;header&gt;
&lt;img src="logo.png" style="float:left; height:58px;" alt=""&gt;
&lt;h1 style="float:left;"&gt;&ampnbsp;Page Header&lt;/h1&gt;
<!-- Navigation bar --&gt;
&lt;nav&gt;
&ampnbsp;&ampnbsp;&ampnbsp;&lt;a href="index.htm"&gt;Home&lt;/a&gt;
&ampnbsp;&ampnbsp;&ampnbsp;&lt;a href="recipe.htm"&gt;Hot Dog&lt;/a&gt;
&ampnbsp;&ampnbsp;&ampnbsp;&lt;a href="https://allrecipes.com/"&gt;All Recipes&lt;/a&gt;
&lt;/nav&gt;
&lt;/header&gt;
<!-- Center page title and image --&gt;
&lt;div style="text-align:center;"&gt;
<!-- Main page title --&gt;
&lt;h2&gt;Welcome to My Site&lt;/h2&gt;
&lt;p&gt;&lt;img src="welcome.png" alt="Welcome"/&gt;&lt;/p&gt;
&lt;/div&gt;
&lt;p&gt;This is text on my home page. I am a sentence that contains some &lt;strong&gt;boldface&lt;/strong&gt; text and some &lt;em&gt;italic&lt;/em&gt; text.&lt;/p&gt;
&lt;p&gt;This is a practice paragraph that contains multiple sentences. A longer paragraph like this will allow me to see &lt;em&gt;word wrap&lt;/em&gt; in action in my Web browser. Word wrap means that the text will wrap to fit the width of the browser window. So long as the browser window is a reasonable width, text will not shoot off past the right edge of the window forcing me to scroll to the right to see it. The wrapping occurs at spaces between words at the ends of lines. That prevents any individual word from being split across two lines.&lt;/p&gt;
&lt;p&gt;
&lt;a href="recipe.html"&gt;See My Hot Dog Recipe&lt;/a&gt;
&lt;/p&gt;
&lt;/body&gt;
&lt;/html&gt;</pre>
```

4. Save your changes.



## Your styling has been temporarily removed.

If you were to look at index.htm in a browser right now, it wouldn't look quite the same. This is because we removed all the CSS styling from the page.

The screenshot shows a web page with the following elements:

- Page Header:** Includes a logo with three colored shapes (blue, green, yellow) and the text "Page Header".
- Navigation:** Links to "Home", "Hot Dog", and "All Recipes".
- Title:** "Welcome to My Site".
- Illustration:** A cartoon character with an orange shirt and blue pants, waving with both hands. Above the character is a blue speech bubble containing the word "WELCOME".
- Text:** "This is text on my home page. I am a sentence that contains some **boldface** text and some *italic* text.  
This is a practice paragraph that contains multiple sentences. A longer paragraph like this will allow me to see *word wrap* in action in my Web browser. Word wrap means that the text will wrap to fit the width of the browser window. So long as the browser window is a reasonable width, text will not shoot off past the right edge of the window forcing me to scroll to the right to see it. The wrapping occurs at spaces between words at the ends of lines. That prevents any individual word from being split across two lines."  
[See My Hot Dog Recipe](#)

We did put the styling in an external file. But the index.html page doesn't know that yet.

In order for *any* page to use an external style sheet, you have to *link* the page to that style sheet, as we'll discuss next.

## Adding a Link to a CSS file

Putting CSS style rules in an external style sheet doesn't make those styles visible to all pages in the site automatically. In fact, the *only* pages that use an external style sheet are pages that link to the style sheet. You need to add a <link> tag between the <head> and </head> tags using this syntax:

```
<link rel="stylesheet" href="path">
```

- The `rel="stylesheet"` segment simply tells how the external filename is *related* to the page. When creating a link to a **relative link** (`rel=`) to a style sheet, the value is always `stylesheet` for a style sheet, no matter what the file name or path of that style sheet.

- The word *path* is just a placeholder for the file path to the style sheet. If the style sheet file is in the same folder as the page, then that path is just the filename (*stylesheet.css* in our case).



## Note

In other people's code you might see some extra code in there, like this:

```
<link rel="stylesheet" href="path" type="text/css">
```

The property:value pairing `type="text/css"` is a leftover from older HTML versions and is no longer required. Modern browsers just ignore that part so there's no harm in having it in the tag, but it's not required by modern browsers.

Let's go ahead and link `index.htm` to our style sheet now.

## Here are the Steps

1. Open `index.htm` in your editor.
2. Place your cursor just before the `</head>` tag and press **ENTER**.
3. On the new line, type the following comment:

```
<!-- Link to external CSS style sheet -->
```

4. Press **ENTER** and add your link code on the next line:

```
<link rel="stylesheet" href="stylesheet.css">
```

## 5. Save changes to the page.

The code at the top of your index.htm page should now look like so:

```
<!DOCTYPE html>
<html>
<head>
<!-- Title in browser window -->
<title>My Website</title>
<!-- Link to external CSS style sheet -->
<link rel="stylesheet" href="stylesheet.css">
</head>
```

Remember, any mistakes will prevent it from working properly, so check your work now. We suggest you close the page in both your editor and your browser. Then, reopen index.htm in your browser to confirm the changes.

## Refreshing Your Page

If it looks like the styling still hasn't kicked in, click the browser's **Reload** or **Refresh** button. Or better yet, do a **hard refresh** by pressing **CTRL + F5** (in Windows) or **COMMAND + F5** (on a Mac). When the styling is right, the header and navbar will be as they were before we externalized the styles.



If you can't get your external style sheet to work, here are the most likely reasons:

- The file name doesn't match because you mis-typed something or added a space.
- The file isn't in your MyWebsite folder.
- There is a typographical error somewhere in your <link> tag.
- There is a typographical error elsewhere in index.htm or stylesheet.css

## Join the Discussion



If you can't get it to work feel free to stop by the Discussion Area for help. However, remember copy/paste all the code from your page and/or style sheet (or attach the files) with your question. That way we troubleshoot your code.

That's it for this lesson. We'll be doing some more layout in the next lesson. For now, hop on over to Chapter 5 and review what you've learned.

## Lesson 8 Review

Congratulations! You've learned quite a bit about organizing information on web pages into separate chunks with various page layout tags:

- You used `<header>...</header>` tags to create a page header.
- You used `<nav>...</nav>` tags to create a navigation bar.
- You also created an *external style sheet* named `stylesheet.css`.

Right now, you've only added these to the `index.htm` file, but putting CSS style rules in an external file allows you to share those styles across multiple web pages.

### It makes it easy to maintain consistent styling across pages.

Segmenting your page into sections, and applying CSS code to those sections with an external CSS style sheet allows you create one style sheet. This helps you avoid repeating the same code over again in every page and makes it easier to make changes. If you do want to make a style change across multiple pages in a site, you just need to change the style in the external style sheet, rather than on each individual page. That's a big time-saver, especially with a site that's grown to dozens or even hundreds of pages.



For all you've learned, there's still more to come. You'll grow your newfound page design and layout skills in Lesson 9. See you there!

## Learning Check

### Page Layout

Text equivalent start.

Item to Match	Matching Choices	Correct Match
<div>...</div>	<ul style="list-style-type: none"><li>• This section displays links to other pages within the site.</li><li>• This section contains content that is usually displayed at the top of each page.</li><li>• This section contains the content that is unique to each page.</li><li>• This creates a section within a page.</li></ul>	<ul style="list-style-type: none"><li>• This creates a section within a page.</li></ul>
<main>...</main>	<ul style="list-style-type: none"><li>• This section contains content that is usually displayed at the top of each page.</li><li>• This section displays links to other pages within the site.</li><li>• This section contains the content that is unique to each page.</li><li>• This creates a section within a page.</li></ul>	<ul style="list-style-type: none"><li>• This section contains the content that is unique to each page.</li></ul>
<nav>...</nav>	<ul style="list-style-type: none"><li>• This section contains the content that is unique to each page.</li><li>• This section displays links to other pages within the site.</li><li>• This section contains content that is usually displayed at the top of each page.</li><li>• This creates a section within a page.</li></ul>	<ul style="list-style-type: none"><li>• This section displays links to other pages within the site.</li></ul>

Item to Match	Matching Choices	Correct Match
<header>...</header>	<ul style="list-style-type: none"><li>• This creates a section within a page.</li><li>• This section displays links to other pages within the site.</li><li>• This section contains the content that is unique to each page.</li><li>• This section contains content that is usually displayed at the top of each page.</li></ul>	<ul style="list-style-type: none"><li>• This section contains content that is usually displayed at the top of each page.</li></ul>

Instructions: Read the item in the first column and consider which choice(s) it matches to in the second column. Read the third column to find out if you are correct.

Text equivalent stop.

---