All websites use HTML and CSS. After learning both of these languages, you will be ready to build your own website!

**1.**

**3.**

**2.**

**1.**

**2.**

**1.**

**2.**

**1.**

**2.**

**1.**

**2.**

**1.**

**1.**

**2.**

**1.**

* *HTML* stands for Hyper Text Markup Language. It is used to give websites structure with text, links, images, and other fundamental elements.
* *CSS* stands for Cascading Style Sheets. It is used to change the appearance of HTML elements.

Instructions

An individual page of a website is referred to as a webpage. Notice the webpage in the web browser. During this lesson, we will explore the HTML elements used to build it. Click Next to learn about the anatomy of an HTML element.

Let's explore the basic anatomy of an HTML element. Line 9 of **index.html**contains a *heading* element:

<h1>You're Building a Website!</h1>

1. All HTML elements begin with an *opening tag*. In this case, the opening tag is <h1>.
2. Most elements require a *closing tag*, denoted by a /. In this case, the closing tag is </h1>.
3. The website user only sees the content between the opening and closing tags.

**Note**: There are several other HTML elements in **index.html** in addition to the heading element. Don't be alarmed! We will encounter most of these in more depth during the lesson.

Instructions

In **index.html**, change the text that appears between the <h1>opening tag and the </h1> closing tag.

Click Run to see the new text shown in the web browser!

Now, let's learn more about the heading element.

Headings are a frequently used HTML element. You can think of them like headlines in a newspaper. Your eyes may notice headings first because the words are large and bold compared to other text on the webpage.

There are six heading elements: h1, h2, h3, h4, h5, and h6. h1 is the largest heading and h6 is the smallest.

Instructions

Use the largest heading to indicate the name of the website.

In **index.html**, one line below <body>, type (don't copy/paste!) the code below:

<h1>Ollie Bike Sharing</h1>

Click Run to view the heading in the web browser.

Use an h3 heading to create the tagline for Ollie Bike Sharing on the next line.

<h3>Share Your Pedals with the World.</h3>

Click Run to view the tagline in the web browser.

Notice how the tagline is smaller than the main heading because it's in an h3 element.

Feel free to change the text of the heading and tagline of the website.

The webpage now has a heading and a tagline. Next, we will add a description of the company.

The HTML *paragraph* element, p, is used to hold one or more sentences, just like paragraphs in an essay or a book.

<p>Paragraph text here</p>

Let's use a paragraph element to add the company description.

Instructions

Below the h3 element, add a paragraph element that contains a description of the company. Here's an example:

<p>Need a set of wheels while you're in town? Use Ollie to pair your perfect vacation with a stylish, affordable bike rental.</p>

Click Run.

In the web browser, notice how text enclosed by the p tags is smaller than heading text and is not bold.

Nice work! The webpage is starting to come together.

What if you wanted to link users to a different webpage? The HTML *anchor*element makes it possible to do this with a single click.

<a href="http://google.com"> Click here for Google!</a>

Anchor elements use an *attribute* to link users to websites. Attributes customize the behavior or appearance of HTML elements. Anchor elements use the hrefattribute, which specifies the webpage to link to. In the example above, the text "Click here for Google!" links to http://google.com.

**IMPORTANT**: Web addresses, such as http://google.com, have a technical name: *URL*.

Instructions

Let's create an anchor element to send visitors to a webpage that lists cities where Ollie bike sharing is available.

Between the opening <p> and closing </p> tags, after the last sentence, enter this text:

Here is a list of cities where you can find us.

Click Run to continue.

Next, use an anchor element to link the word "list" to the URLcities.html.

Here is a <a href="cities.html">list</a> of cities where you can find us.

**Note:** cities.html is within the Ollie site folder, so we do not need to specify a full URL, such as http://olliebike.com/cities.html.

When you have added the anchor element, click Run.

In the web browser, you will now see the word "list" underlined, indicating that the word is a link. Click on the link to test it out!

Good job! Creating links with the anchor element is a fundamental web development skill.

The diagram to the right illustrates the different parts of the anchor element syntax. In the diagram, notice the following:

1. Any valid URL can be used for the value of the href attribute.
2. The URL must be enclosed with quotation marks.
3. Text between the <a> and </a> tags can be as few or as many words as you would like.

Click Next to learn how to add images to a webpage!

To add images to a webpage, use the HTML *image* element:

<img src="https://s3.amazonaws.com/codecademy-content/projects/make-a-website/lesson-1/bikes1.jpg"/>

Just like websites have URLs, images on the web also have URLs. Image URLs typically end with the .jpg or .png file extension. The src attribute sets the*source* for an image element.

Image elements are *self-closing*, which means they do not need a closing tag.

Let's use an image element to add a photograph to our webpage.

In **index.html**, on the line below the paragraph element, add an image element. Use the following as the image URL:

https://s3.amazonaws.com/codecademy-content/projects/make-a-website/lesson-1/bikes1.jpg

Remember, the URL must be enclosed in quotes!

Click Run to view the image in the web browser.

Want to change the photo that the image element displays? Simply set the value of src to a different photo's URL.

Here's another photo:

https://s3.amazonaws.com/codecademy-content/projects/make-a-website/lesson-1/bikes2.jpg

Replace the current photo with this one and click Run.

Awesome! The photo makes the webpage much cooler. What's cooler than a photo? A video!

The HTML *video* element can add video to a webpage.

<video width="320" height="240" controls>

<source src="video-url.mp4" type="video/mp4">

</video>

The video element uses a number of attributes. Let's take a look at them:

1. width and height: Set the size of the screen that displays the video.
2. controls: Adds play, pause and volume control.
3. source src: Sets the URL of the video to play.
4. type: Specifies different video formats.

In **index.html**, delete the image element so we can use that space on the webpage for a video.

Click Run.

Next, below the paragraph element, create a video element. Use the code above as a guide.

The video will need:

1. width and height attributes
2. a controls attribute
3. a source src set to:

https://s3.amazonaws.com/codecademy-content/projects/make-a-website/lesson-1/ollie.mp4

Remember, the attribute values must be in quotes, except forcontrols.

Click Run to view the video in the web browser.

Click the Play button to play your video. When you've watched enough, click Next to continue.

Impressive! With just five page elements, you've already created an interesting website!

Another essential HTML element is the *unordered list*. Items in an unordered list are referred to as *list items*. Each item is bulleted, not numbered. For example:

* A list item
* A second list item
* A third list item

The HTML code for the list above:

<ul>

<li>A list item</li>

<li>A second list item</li>

<li>A third list item</li>

</ul>

About unordered lists:

1. ul tags create the unordered list.
2. li tags contain each list item.

Unordered list elements can be used to organize content on a webpage in a number of ways. Below we will use one to organize our website's navigation menu, sometimes called a *navbar*.

Our navbar will go above the company heading h1 element, so first we will need to create some new lines. In **index.html**, press enter to create several new lines between <body> and <h1>.

First, we will create an h2 element to serve as the heading for the navbar. The text can read "Ollie".

<h2>Ollie</h2>

Click Run to continue.

Below the navbar h2, create an unordered list element. The list items will be different pages a site user can visit:

<ul>

<li>sign up</li>

<li>search bikes</li>

<li>reserve a bike</li>

<li>about us</li>

</ul>

After you insert all the list items, don't forget to close the unordered list with a closing </ul> tag.

Click Run to view the results in the web browser.

With the video and unordered list elements, you may have noticed something interesting: these HTML elements had other HTML elements *nested* inside of them.

For example, in unordered lists, li elements are nested inside the ul.

<ul>

<li>First item</li>

<li>Second item</li>

</ul>

Web developers refer to the enclosing element as the *parent* element and the enclosed elements as *children*.

Referring to HTML elements as parents and children may sound funny, but it's a core web development concept. The web browser also knows about these parent/child relationships, which will be important as we explore CSS in the next lesson.

Instructions

In the diagram to the right, notice the following:

1. The ul element is the parent of each li.
2. The li elements are children of the ul.
3. Code indentation signifies the relationship between parent and child elements.

After you have familiarized yourself with the diagram, click Next to continue the lesson.

Now that we know about HTML element nesting and parent/child relationships, let's see another way these concepts are applied on a real-life webpage.

*Div* elements divide your page by enclosing other elements. These enclosed groups of elements can then be organized, moved and styled independently from one another.

Div elements are often used with the *class* attribute. Here's an example:

<div class="main">

...

</div>

**Note**: The ... indicates where other HTML elements would normally be enclosed by the div.

Below, we will use divs to divide our page into three separate parts.

First, we will enclose all the elements between <body>...</body> in a div element with a class attribute of "container".

<div class="container">

...

</div>

All the HTML elements you added in previous exercises should be inside the opening <div class="container">:

1. the navbar h2
2. the navbar ul
3. the company heading h1
4. the tagline h3
5. the paragraph
6. the video

Make sure the closing </div> tag is at least one line below the closing</video> tag.

Click Run. Notice that the elements have shifted in the web browser. We will fix this effect later on.

Next, we will enclose the navbar h2 and ul in another div and give it a class attribute of "nav". You can use the code below as a guide:

<div class="nav">

<h2>Ollie</h2>

<ul>

<li>sign up</li>

<li>search bikes</li>

<li>reserve a bike</li>

<li>about us</li>

</ul>

</div>

Click Run.

Notice that the navbar section now has a yellow background.

Finally, enclose the company h1, the h3, paragraph and video elements inside a third div with the class attribute of "main".

Feel free to use the code below as a guide:

<div class="main">

<h1>Ollie Bike Sharing</h1>

<h3>Share Your Pedals with the World.</h3>

<p>Easy-to-use, free bike sharing now available in 27 cities.</p>

<video width="320px" height="240" controls>

<source src="https://s3.amazonaws.com/codecademy-content/projects/make-a-website/lesson-1/ollie.mp4" type="video/mp4">

</video>

</div>

There will now be a total of three divs on the page: the div with the "container" class contains the divs with the "nav" and "main" classes.

Click Run.

The file **main.css** contains CSS styling for the "container", "nav" and "main" classes, which enabled page elements to change. If you're curious, explore **main.css** for fun. We will learn how CSS works later in the course.

The last HTML elements we will explore are involved in *metadata* processes. You can think of these elements as the "brains" of a webpage because they communicate vital information to the web browser, but are not visible to a webpage visitor.

1. <!DOCTYPE html>: Tells the web browser to expect an HTML document.
2. <html>...</html>: The root of the HTML document and parent of all other HTML elements on the webpage.
3. <head>...</head>: Enclose other metadata about the site, such as its title.
4. <title>...</title>: Contains the site's title, which is one way users can find your site through a search engine, like Google.
5. <meta charset="utf-8"/>: Tells the web browser which character set to use. In this case, the character set is "utf-8".

Instructions

Identify the html, head, title and <meta charset="utf-8"/> elements, and the !DOCTYPE declaration.

**Note**: In addition to the metadata elements described above, you will also notice *link*. We will learn about link elements in the next Unit, *A Closer Look at CSS*.

Click Next to review the skills learned in this lesson.

Congratulations! You've learned enough HTML to create a great website!

Before we move on to styling with CSS, let's review what we learned in this lesson.

#### LANGUAGES

* **html**: stands for *hypertext markup language*, and is used to give a webpage structure.
* **css**: stands for *cascading style sheets*, and is used to style HTML elements.

#### HTML ELEMENTS

* *h1 - h6*: indicate text headings on a webpage. h1 is the largest heading; h6 is the smallest.

<h1>Heading</h1>

* *p*: used for non-heading text, such as the bodies of articles or company descriptions.

<p>Description of company here.</p>

* *a*: short for anchor and used to add links to other webpages. Anchor elements typically have an href attribute:

<a href="http://codecademy.com">Click here</a> to learn how to make a website!

* *img*: used to add an image to a webpage. Image elements are *self-closing*and do not require a closing tag:

<img src="https://images.com/favorite.png">

* *video*: used to add videos to a webpage, and uses multiple attributes and a nested source element:

<video width="320" height="240" controls>

<source src="https://movies.io/great-clip.mp4" type="video/mp4">

</video>

* *unordered list*: used to create lists on a webpage and requires li elements inside a ul:

<ul>

<li>list item</li>

<li>another item</li>

<li>yet another</li>

</ul>

* *div*: used to organize HTML elements into different groups, which can be given a class attribute:

<div class="main">

<h2>Subheading!</h2>

</div>

* *metadata tags*: provide metadata about a webpage.

#### WEB CONCEPTS

* **parent/child elements**: used to describe HTML elements that enclose or are enclosed by other elements. For example, below the ul is the parent and the li items are children:

<ul>

<li>...</li>

<li>...</li>

<li>...</li>

</ul>

Click Up Next to start learning about CSS!