



Checkpoint 2

Outline

# HTML basics

Have you ever noticed what makes up the vast majority of website content? Throughout your many experiences online (and with your new understanding of the HTML and CSS working behind the scenes), you've probably noticed that text and images make up nearly all the core content for every web page on the internet. HTML serves as the backbone for those images and text, so knowing how to code those elements is an essential next step. With these HTML skills under your belt, you'll be able to build content for your future web pages.

By the end of this checkpoint, you will be able to do the following:

- Debug errors related to HTML text within prebuilt web pages
- Build your own custom web page with text and images

## Anatomy of HTML

In the previous checkpoint, you learned that HTML looks a bit like the code below. Which pieces do you recognize?



```
<main>
  <h1>This is a simple title created within HTML.</h1>
  <p>This is a simple paragraph created within HTML.<
</main>
```

Though it may look complicated at first blush, there's actually some repetition at play here. The code above uses several common HTML components, and you'll first explore two of the most important ones: elements and tags.

Outline



## Elements and tags

You started learning about elements and tags in the previous checkpoint. Now, it's time to dive a little deeper.

Take a look at the code below. This is called an *element*—specifically, an `<h1>` element. In fact, the entire line, from beginning to end, is referred to as an `<h1>` element.

```
<h1>This is a simple title created within HTML.</h1>
```

An HTML element is often made up of two *tags*: an *opening tag* and a *closing tag* (sometimes called a begin tag and an end tag, respectively). The opening tag begins the HTML element, and it's enclosed in brackets `<>`, as seen below.

```
<h1>
```

The closing tag, which you can see below, ends the HTML element. Like the opening tag, it is enclosed in `<>`. But it has one important difference: the closing tag also has a forward slash `/` before the element name. The closing tag will always have this `/`, which signifies the end of an element.

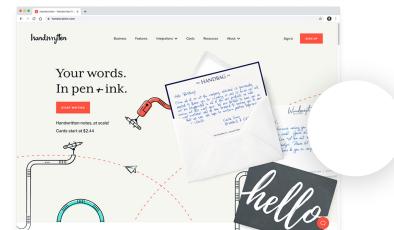
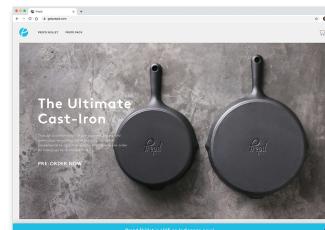
```
</h1>
```

## Outline

# HTML text

Text in HTML is unique. This is because of how the text is grouped visually using code. HTML text can be grouped into two primary categories: *headings* (sometimes called headers) and *paragraphs*.

Take a look at the links to three web pages and the accompanying screenshots shown below. Can you identify the headings? What about the paragraphs? Can you tell the difference?



Source: [Upstream](#)

[International](#)

Source: [Prepd](#)

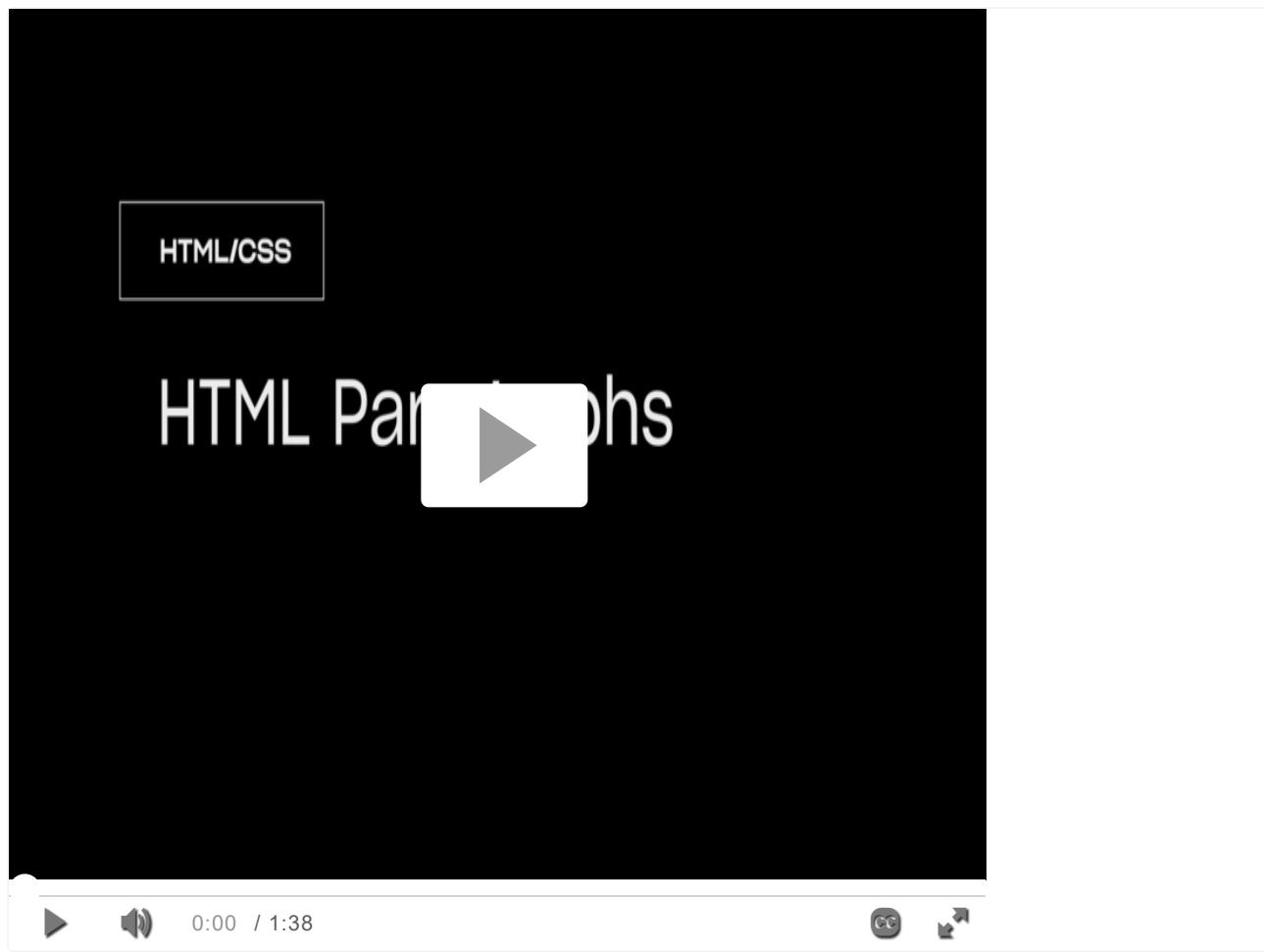
Source:

[Handwrytten](#)

## HTML paragraphs

At its most basic level, an HTML paragraph is a block of content, such as text, that is represented by a `<p>` element. An HTML paragraph is set off with `<p>` and `</p>`, and these tags give website content their familiar paragraph formatting. Watch the following video to learn more about how HTML paragraphs work.

Outline



It's time to dig a little deeper into HTML paragraph code. Take a look at this screenshot—can you identify the HTML paragraphs? What about the HTML headings?

# Dogs or Cats?

## What kind of person are you?

The world is made up of three types of people: Cat people; dog people; and boring people.

Why do you think you have a preference for one over the other?

### Dogs



The dog's story is inexorably linked with the human's. Dogs are dogs, and not wolves, purely because of human tampering. They were made by us to fit us. The same is true of almonds, but almonds don't play fetch. We took, and take, the cute ones, the sweet ones, the ones we like the best and mold them and their offspring, often to the detriment of their health, into our perfect companions. And that they are.

### Cats



Cats are graceful, clumsy, silky soft with sharp edges, softly glowing eyes in the dark, purring, kneading balls of fur curled up closely against your body warmth, intelligent, annoying, funny, great conversationalists with comments about many things, and intriguing little creatures who display many emotions including affection, even love for their guardians.

You may have noticed that there are four paragraphs on the *Dogs or Cats?* web page. In HTML, every paragraph tends to have the same basic format. This means that each paragraph displays in the same font, size, and color, which helps provide some structural consistency across web pages.

It's also important to notice that each paragraph has a space (also known as a *hard return*) following the text, which separates each paragraph from the one above it and the one below it. However, these hard returns do not show in the code itself; they're defined by the `<p>` and `</p>` tags. Because web browsers try to present web pages for human eyes, web pages will always display only a single space in text—regardless of how many spaces you add between sentences in a paragraph, or even between words.

## Demo: HTML paragraphs

Take a moment to answer these questions while reviewing the code in the Repl.it below. Feel free to make changes to the code to see what happens!

- What is the difference between an opening paragraph tag and a closing paragraph tag?
- What does adding spaces and hard returns between the sentences and paragraphs in the HTML code do? How do those spaces affect the paragraphs when viewed in the web page?
- What happens if the HTML code used to begin and end paragraphs is removed from all the text?

[Run ▶](#)[open in repl.it](#)

index.html



```
1 <p>I used to think that rain was romantic. That it  
2   intoxicating and it whispers of new beginnings as  
3 <p>When it rained, you could imagine your car brea  
4   car to help you out the rain plastering their hair  
5   leaning in closer to them with the roaring of the  
6   kissing, with rain pattering relentlessly on top o  
7   froze.</p>  
8 <p>But it would be alright. Being cold doesn't mat  
9   always thought rain was like that, so wonderful an
```

<https://Paragraphs--thinkful.repl.co>

## Outline

[Console](#)[Shell](#)

As you've witnessed firsthand in this demo, you need a `<p>` to begin a paragraph and a `</p>` to end a paragraph. As you add these paragraph tags to an HTML page, they will force space above and below the text to help make the content more readable. If you play around with the

spaces and hard returns, you'll notice that additional spaces or hard returns *without* paragraph tags may change the way the code appears, but they won't change the presentation of the web page. If all the tags are removed, the blocks of text will simply lump together and no longer be separated into discrete paragraphs.

## Drill: Paragraph practice

Now, it's time to apply what you've learned. Within this REPL, update the HTML so that each paragraph displays properly. You should have eight HTML paragraphs.

[Run ▶](#)[open in repl.it](#)

index.html



```
1 So, when they came to a good-sized farmhouse, Doro  
2  
3 A woman opened it just far enough to look out, and  
Lion with you?"  
4  
5 "We wish to pass the night with you, if you will a  
and comrade, and would not hurt you for the world.  
6  
7 "Is he tame?" asked the woman, opening the door a  
great coward, too. He will be more afraid of you t  
8  
9 "Well," said the woman, after thinking it over and  
you may come in, and I will give you some supper a  
10  
11 So they all entered the house, where there were, b  
hurt his leg, and was lying on the couch in a corn  
company, and while the woman was busy laying the t  
12
```

<https://Practice-Paragraphs--thinkful.repl.co>

## Outline

[Console](#)[Shell](#)

When you're done, feel free to compare your practice project with the following REPL, which has the code completed properly.

## Practice Paragraphs Completed

# HTML headings

As the name suggests, HTML headings are titles or subtitles that are used throughout a web page to break up, organize, and highlight content. HTML headings are defined using tags, like `<h1>`, in which the numbers in the tag suggest where the heading sits in the hierarchy. Watch the video below to learn more about HTML headings.

Outline

The video player displays a black screen with white text. In the top left corner, there is a white rectangular button with the text "HTML/CSS". In the center, the words "HTML Headings" are written in a large, white, sans-serif font. To the right of the text is a white play button icon containing a grey triangle pointing right. At the bottom of the video player, there is a control bar with a play button, a volume icon, and the text "0:00 / 1:10". On the far right of the control bar are two small icons: one for closed captions ("CC") and one for sharing.

Take a moment to revisit the *Dogs or Cats?* sample web page shown above. Did you notice that this web page has three different, and distinct, headings?

1. **A primary heading:** The heading *Dog or Cat?* is a primary heading.

2. A secondary heading: The subheading *What kind of person are you?* is a secondary heading.

3. A tertiary heading: There are two tertiary headings: *Dogs* and *Cats*.

## Demo: Headings and paragraphs

It's time to review a code sample that showcases both headings and paragraphs written in HTML. Take a moment to answer the following questions while reviewing the code in the Repl.it below.

- What HTML code is used to begin headings?
- What HTML code is used to end headings?
- How many different heading sizes are possible in HTML (in other words, `<h1>` through what number)?

**Outline**

As you've seen, headings help identify the blocks of content that sit below them on a web page. Headings also tend to be displayed larger, more prominent text. This gives the content a visual and logical

hierarchy, helping the viewer understand the various levels of importance and relevance within the content.

All headings in HTML use heading tags. These tags begin with an `h` and are followed by a number, like `h1`. Like paragraph tags (and other tags in HTML), heading tags are enclosed in brackets `<>`, like `<h1>`. And like the paragraph closing tags, a closing tag for a heading must also include a forward slash `/` to end the heading, which looks like this: `</h1>`.

There are six valid headings in HTML—`<h1>` through `<h6>`—with `<h1>` being the largest heading and `<h6>` being the smallest. However, it's important *not* to think of these headings simply in their visual style, like their size or boldness. You should think of each heading as representing a different level of importance.

## Outline

An `<h1>` element is often the first, and more prominent, heading on a web page. It helps to identify the most important information, and it is often used for the company name, the website name, or a web page title. Because search engines use HTML tags to understand how to present content, a best practice is to use an `<h1>` only once per web page to help the `<h1>` keep its relevance in search engines.

An `<h2>` element is best used for large, but slightly less important subheadings, and it can be used several times within a single web page.

An `<h3>` element is best used for a subheading under another subheading, like the tertiary headings of *Dogs* or *Cats* mentioned above. It can always be used as much as needed, and can be broken down further into `<h4>`, `<h5>`, or `<h6>`, depending on how

wish to structure the information. You can use these smaller subheadings as often as you'd like, as well.

It's important to note that there is no `<h7>` tag, or any heading with a higher number in it. However, HTML will not provide any visible errors if you use it, so it's up to you to keep an eye out.

## Drill: Headings and paragraphs

Within the Repl.it below, update the HTML code by making the proper headings and paragraphs. The notes suggesting headings and paragraphs can be removed.

**Outline**

Feel free to compare your practice project with this REPL that has the code completed properly.

**Heading and Paragraph Practice Completed**

# Attributes

Now that you're familiar with elements and tags, you're ready to explore a new HTML concept: attributes. Revisit the image below and take a look at the code below that.



You'll notice that this HTML element looks a bit different from what you've been working with. What makes it different?

Outline

```

```

In HTML, *attributes* are extra bits of information that are tied with certain elements. They are always written inside the opening tag, and they will always follow this structure: `attribute="value"`. Understanding this syntax will help you as you dive into HTML images.

## HTML images

At one point or another, you've likely heard of "embedding an image." Though that phrase may be used in the context of HTML, it's technically a misnomer. In HTML, images are actually *not* embedded into a page, as they might be in a PDF or Google Doc. All images in an HTML

page are separate files that must be located on a web server and then referenced on the web page in order to appear on the internet.

## Drill: HTML images

Take a moment to explore a bit more on your own. Try to answer the following questions using the code in the Repl.it below. You don't need to look up the answers or read ahead—just see what you can learn from reviewing and playing around with the code.

- How is the `<img>` element different from the `<h1>` and `<p>` elements?
- What does `img` stand for?
- What does `src` stand for?
- What does `width` stand for?
- What does `alt` stand for?
- No measurement is given within the width value. What measurement is it?
- When you change the width value, what happens to the image?
- When `width= "500"` is removed, what happens to the image?

**Outline**

As you may have figured out, `<img>` is the opening image tag in HTML, and the `img` stands for *image*. Images do not have closing tags. Tags that open and close are needed to wrap around content, like text, to define where the content starts and ends. But with images in

HTML, there is no need for a closing tag. Only one image is added at a time, so only the opening `<img>` is needed.

The `src` in the image code stands for *source*. It references the online location of the image; in this case, `src` is referencing the `waterfall.jpg` that is located in the same folder as the HTML page. The `width` refers to the *width that the image will display* on the page, and it is always set in pixels. When a width size is not set, the image will display at whatever size the image file is naturally saved as.

The `alt` represents the *alternative text* that describes the image, and it should be applied to every image on a web page. Alternative text is important for several reasons:

- If the image link is broken, the alternative text will display for the user so they will know what image should appear on the screen.
- Alternative text is essential for accessibility. It's used by visual web readers, which will describe images to users who are blind or visually impaired.
- Providing an `alt` value will help improve the visibility of a website on the internet because it provides search engines with more detailed information.

## Drill: Practice with images, headings, and paragraphs

You're ready to apply your new knowledge. In this Repl.it, fork the code and update the HTML to complete the practice project. The nc

suggesting headings and images can be removed. Set the `src` value to the following for the crow image and scarecrow image, respectively.

```
"./crow.jpg"  
"./scarecrow.jpg"
```

Using your own judgment, set an appropriate width for each image that presents each alongside the text in an attractive, visually compelling way.

**Outline**

Feel free to compare your practice project with the following REPL that has the code completed properly.

**Image, Heading, and Paragraph Practice Completed**

# FAQs: Elements and tags

As a newcomer to HTML, you may still have some questions about using tags and elements. Below are some frequently asked questions that will help you out.

**Question:** Should tags be written in capital letters or lowercase letters?

**Answer:** HTML5 doesn't require one case or the other. However, it is considered a best practice to type all code as lowercase.

**Question:** Does everything need a tag to show up on an HTML page?

**Answer:** Text is the only content you could technically add without tags. But without tags, there's no way to change the style of the text. Images, links, and containers can't be created without tags.

**Question:** What if I forget to add a closing tag? Or what if I don't add the slash to a closing tag?

**Answer:** If you miss these key pieces, bad things will happen to your web page. But the good news is that if you make a mistake like this, it's easy enough to fix it. Often, you'll get a clear cue that something's wrong: visual errors will appear on your page, containers will not display properly, or content will not look like you expect it to (often right before a closing tag is needed). By looking at the code in that area, you'll likely be able to spot the problem and add or fix the proper tag.

**Question:** Do all HTML elements except images require both opening and closing tags?

**Answer:** No. There are 16 elements that are self-closing, meaning they only require a single tag. Images are one example of a self-clos-

element. The syntax of a self-closing tag is as follows: `<element attributes />`.

# Assignment

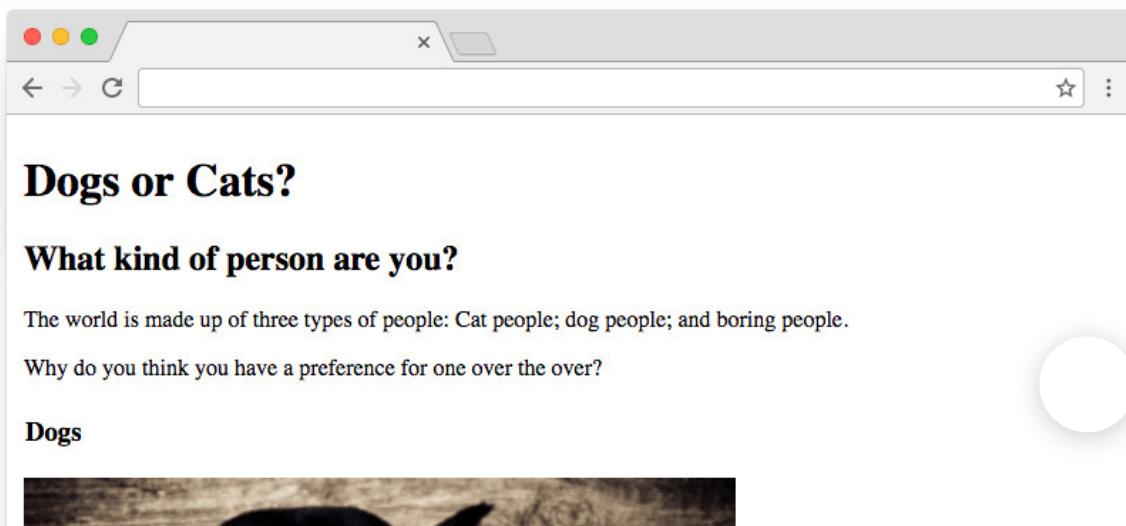
For this assignment, you will be recreating the *Dogs and Cats?* web page, which is shown again here. This is a simple web page with text and images, and you've totally got the skills to replicate it!

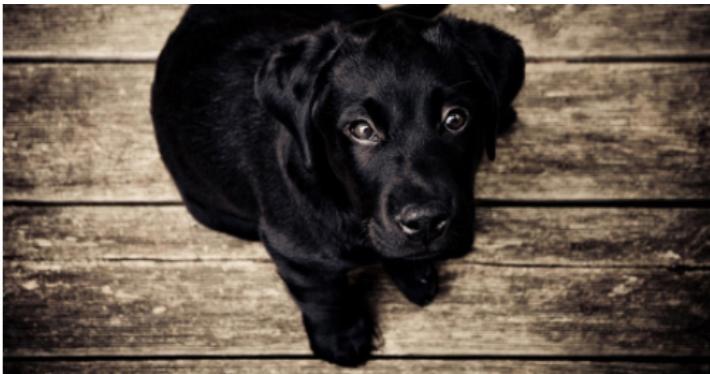
This assignment allows you to practice an essential skill in web development: the ability to look at a web page and recreate it with code based on what you see. It's important that you push yourself here. Don't peek at the underlying HTML code in the completed project unless you are completely stuck.

Outline

Here's what to do: first, review the code in the Repl.it below. Then, take some time to update the code using both what you know about HTML and the image of the web page below.

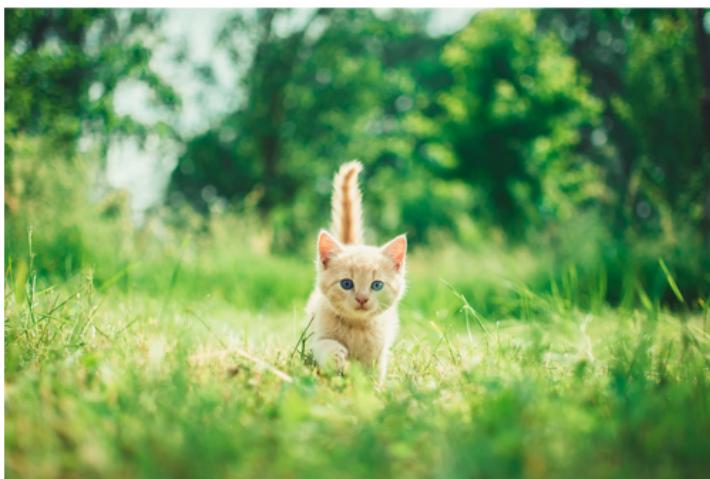
This is a major step forward in your journey to becoming a web developer! When you're done, submit a link to your Repl.it in the box below.





The dog's story is inexorably linked with the human's. Dogs are dogs, and not wolves, purely because of human tampering. They were made by us to fit us. The same is true of almonds, but almonds don't play fetch. We took, and take, the cute ones, the sweet ones, the ones we like the best and mold them and their offspring, often to the detriment of their health, into our perfect companions. And that they are.

## Cats



Cats are graceful, clumsy, silky soft with sharp edges, softly glowing eyes in the dark, purring, kneading balls of fur curled up closely against your body warmth, intelligent, annoying, funny, great conversationalists with comments about many things, and intriguing little creatures who display many emotions including affection, even love for their guardians.

**Outline**

When you have completed your practice project, feel free to compare your code with the completed project example.

**Practice Project Complete**

# Checkpoint

Submit your ideas or a link to your work here and use it as a conversation starter during your next mentor session.

This checkpoint will not be graded, but is still required.

## Your work

03.04.21



Share your ideas here...

Outline

**\*\*bold\*\* \_italic\_ `code` > quote - bullet list**

Preview

Completed

Next checkpoint



How would you rate this content?

[Report a typo or other issue](#)

[Go to Overview](#)

**Outline**