



Web Development 1

Html

Learn the basics of web development in this introductory HTML & CSS course.

Table of Contents

1. Humble Beginnings
2. Breaking News
3. Paragraph Paragon
4. Headliner
5. Illustrious Imagery
6. Big and Tall

- 7. Dangerous Divide
- 8. Listing Liaison
- 9. Orders Wanted
- 10. Stylish Intent
- 11. Classy
- 12. Identification, Please
- 13. Wanted Poster

#1. Humble Beginnings

Level Overview and Solutions

Intro

CodeCombat Web Development Track

Learn to make your own webpages!

Default Code

```
<!-- Welcome to CodeCombat Web Development! -->
<!-- You will learn the basics of web development! -->
<!-- Blue lines are comments. Read them for help! -->

<!-- Write anything below to change the webpage. -->
Edit me!
```

Overview

HTML

HTML, or the **HyperText Markup Language**, is the building block of the web! It is an well-defined way of structuring various parts of a website.

For now, all you need to do is change the text written on your Web Parchment to the left by adding or removing letters! Try it out!

Humble Beginnings Solution

```
<!-- Welcome to CodeCombat Web Development! -->
<!-- You will learn the basics of web development! -->
<!-- Blue lines are comments. Read them for help! -->

<!-- Write anything below to change the webpage. -->
Edit me!
Anything!
```

#2. Breaking News

Level Overview and Solutions

Intro

The `
` tag places a **break** between two lines of text.

Default Code

```
<!-- In HTML, you must use tags to format your text. -->
<!-- All HTML tags begin with < and end with >. -->
<!-- The text between < > determines how the tag acts. -->
<!-- The <br> tag creates a break in lines of text. -->

Welcome to HTML!
<br>
You'll be learning many fun things along the way.
<!-- Add a <br> tag below: -->

I should be on my own line, not after another sentence!
<!-- Add another <br> below, and write an another line: -->
```

Overview

HTML Tags

HTML tags are a way of telling the computer how to display certain content. While it is helpful to view text on multiple lines, the computer doesn't know if there should be a break between two lines unless you tell it to.

The `
` tag is the way you can do that! It will break text apart and put each part on its own line.

```
I'm on my own line.
```

```
I'm after the last sentence. :(
```

```
I'm on my own line.
```

```
<br>
```

```
I am now on my own line! :)
```

Breaking News Solution

```
<!-- In HTML, you must use tags to format your text. -->
<!-- All HTML tags begin with < and end with >. -->
<!-- The text between < > determines how the tag acts. -->
<!-- The <br> tag creates a break in lines of text. -->
```

```
Welcome to HTML!
```

```
<br>
```

```
You'll be learning many fun things along the way.
```

```
<!-- Add a <br> tag below: -->
```

```
<br>
```

```
I should be on my own line, not after another sentence!
```

```
<!-- Add another <br> below, and write an another line: -->
```

```
<br>
```

```
hi
```

#3. Paragraph Paragon

Level Overview and Solutions

Intro

The `<p>` tag is used to group text into paragraphs.

Don't confuse it with the `
` tag which is used to force a line-break!

Default Code

```
<!-- The <p> tag is used to group paragraphs of text. -->
<!-- Most tags need a opening tag and a closing tag. -->
<!-- An opening tag is: <p>, the closing tag is </p> -->
<!-- The / marks the end of an HTML element. -->
<!-- Note the / is before p, not after! -->

<!-- <br> tags are for forcing a line break... -->
<!-- ... <p> tags are for grouping paragraphs of text. -->
<p>HTML tags can be on the same line, or...</p>
<p>
    ... on different lines! Just make sure the opening and closing tags exist.
</p>
<!-- Add a <p> tag below: -->

    This should be a paragraph. Surround me!
<!-- Add a </p> tag below: -->

<!-- Finally add a <p> and </p> with text between. -->
```

Overview

HTML Start/End Tags

HTML has a few special empty tags, like `
`, but a majority of HTML tags require a start and end tag. Or, more commonly: opening and closing tags.

An opening tag is the one that tells the computer everything after it is contained within the rules of that tag. The paragraph element's start tag is: `<p>`.

A closing tag tells the computer that it shouldn't apply any rules to the content after it. The paragraph element's end tag is: `</p>`

Remember to close all tags, or your webpage will look weird!

```
<p>I am a standalone paragraph! I'm pretty neat.</p>
<p>I'm another paragraph, I'm cool too, I promise!</p>
```

Remember line-spacing doesn't matter to the computer! Using indentation and line breaks, you can make your HTML nice and easy to read. It'll get more messy in the future, so it's a good habit to practice now!

```
<p>
    I'm still just a regular paragraph, nothing has changed.
</p>
<p>You can mix styles, as well! Depending on what feels like the best way.</p>
<p>
    You can even use the br-tag inside p-tags!
    <br>
    Look, I'm on another line, in a paragraph!
</p>
```

Paragraph Paragon Solution

```
<!-- The <p> tag is used to group paragraphs of text. -->
<!-- Most tags need a opening tag and a closing tag. -->
<!-- An opening tag is: <p>, the closing tag is </p> -->
<!-- The / marks the end of an HTML element. -->
<!-- Note the / is before p, not after! -->

<!-- <br> tags are for forcing a line break... -->
<!-- ... <p> tags are for grouping paragraphs of text. -->
<p>HTML tags can be on the same line, or...</p>
<p>
    ... on different lines! Just make sure the opening and closing tags exist.
</p>
<!-- Add a <p> tag below: -->
<p>
    This should be a paragraph. Surround me!
<!-- Add a </p> tag below: -->
</p>
<!-- Finally add a <p> and </p> with text between. -->
<p>
    Continue to be alphabetical!
</p>
```

#4. Headliner

Level Overview and Solutions

Intro

The `<h1>` , `<h2>` , and `<h3>` tags are used to define headers.

They are good for labelling content.

Default Code

```
<!-- Header tags are used to label content. -->
<!-- <h1> tags are used for the page title. -->
<!-- <h2> through <h4> tags are sub-headers. -->

<h1>[What?]</h1> <!-- △ Change this line to anything. -->
<h2>Ogreball</h2>
<!-- Add another <h2> with a name of a sport! -->
```

Overview

Header Tags

Headers are a way of labeling content. Add headers using `<h1>`, `<h2>`, and `<h3>` tags. These are open and close tags, so include closing tags like `</h1>`.

```
<h1>Best Game</h1>
<p>
    CodeCombat has ogres! Ogres are blue and stomp around angrily. I beat up ogres and get experience
    for winning.
</p>
```

Headliner Solution

```
<!-- Header tags are used to label content. -->
<!-- <h1> tags are used for the page title. -->
<!-- <h2> through <h4> tags are sub-headers. -->

<h1>[What?]</h1> <!-- Δ Change this line to anything. -->
<h2>Ogreball</h2>
<!-- Add another <h2> with a name of a sport! -->
<h2>Griffin Soccer</h2>
```

#5. Illustrious Imagery

Level Overview and Solutions

Intro

The `` tag requires a `src` attribute.

To add an attribute , include it between the `<` and `>`.

Default Code

```
<!-- It is possible to add images to webpages. -->
<!-- You will need to know the URL of the image. -->
<!-- Click "Image Gallery" for CodeCombat images. -->

<h1>My Friends</h1>
 <!-- Δ Change this src! -->
<!-- Add 2 more images to fill up the page! -->
```

Overview

Images

The `` tag is used for adding image to the page. It requires a `src` attribute to understand what image to display.

Attributes

attributes are pieces of information or data which is included inside of the HTML tags themselves. For example, `` tags have a mandatory attribute called `src` which is the URL "source" for the image.

To add an attribute to an HTML tag, include it between the `<` and `>`. It must have a `=` and `"` surrounding the value assigned to that value. Not much different from creating a variable! But instead, you are setting the value for the `tag`.

```

```

Illustrious Imagery Solution

```
<!-- It is possible to add images to webpages. -->
<!-- You will need to know the URL of the image. -->
<!-- Click "Image Gallery" for CodeCombat images. -->

<h1>My Friends</h1>
 <!-- Δ
    Change this src! -->
<!-- Add 2 more images to fill up the page! -->


```

#6. Big and Tall

Level Overview and Solutions

Intro

attributes are information, or data, that can be included inside of an element's tag. For example, it is possible to resize img elements by including a width and height attribute .

```
<!-- The follow adds 2 new attributes other than the src attribute! -->
<!-- width and height control how tall and wide elements are. -->

```

Default Code

```
<!-- <img> tags can be resized using attributes. -->
<!-- width= and height= force the size for <img> tags. -->
<!-- The image scales proportionally if one is set. -->
<!-- If both are set, the <img> stretches to the size. -->

<!-- This image is 32x32 instead of the default 100x100 -->

<!-- This image is forced to be 200x100! -->

<!-- Add another <img> tag and use width or height: -->
```

Overview

Attributes Con't

There are more attributes than just `src`. To define the size of an `` you can use `height` and `width`. `height` and `width` default to "pixel" or `px` sizing. So setting a 200 pixel wide image to 50 will scale it down 25%, not 50%.

If only one of `width` or `height` is included, the image will scale proportionally. Or, that is, if `width` is set, then the image's height will automatically change to keep the image's original proportions.

If *both* `width` and `height` are set, then image will force resize to match those numbers.

```
<!-- An image tag with the width, height and src attributes. -->

```

Big and Tall Solution

```
<!-- <img> tags can be resized using attributes. -->
<!-- width= and height= force the size for <img> tags. -->
<!-- The image scales proportionally if one is set. -->
<!-- If both are set, the <img> stretches to the size. -->

<!-- This image is 32x32 instead of the default 100x100 -->

<!-- This image is forced to be 200x100! -->

<!-- Add another <img> tag and use width or height: -->

```

#7. Dangerous Divide

Level Overview and Solutions

Intro

`<div>` tags are used for grouping pieces of content together.

Use them to wrap individual blog posts!

Default Code

```
<!-- The <div> tag is for grouping information. -->
<!-- Use <div> tags to organize pieces of your website. -->

<!-- Below, each blog post is contained inside a <div> -->
<div>
  <h2>Ogretober 3st, 1404</h2>
  <p>
    I met a new enemy today! Her name is Cindy and she shot my ogre allies from afar. I dislike her
    and her arrows.
  </p>
  
</div>
<!-- Add another <div> below: -->

<!-- Add an <h2> tag to title the blog post: -->

<!-- Add a <p> tag to give the blog post content: -->
```

Overview

Div Tags

`<div>` tags are a way of organizing information on your website. They are used to group certain elements, as well as add line-breaks between various parts of content.

Consider a blog (as that's what is being edited in this level,) with a bunch of blog posts.

The header itself is a `<div>`, the container of individual blog posts is a `<div>` and each individual blog post is it's own `<div>`. It helps separate the each piece of content into it's own editable chunk.

We will demonstrate why this is important, later!

Dangerous Divide Solution

```
<!-- The <div> tag is for grouping information. -->
<!-- Use <div> tags to organize pieces of your website. -->

<!-- Below, each blog post is contained inside a <div> -->
<div>
  <h2>Ogretober 3st, 1404</h2>
  <p>
    I met a new enemy today! Her name is Cindy and she shot my ogre allies from afar. I dislike her
    and her arrows.
  </p>
  
</div>
<!-- Add another <div> below: -->
<div>
  <!-- Add an <h2> tag to title the blog post: -->
  <h2>Stardate 393.4</h2>
  <!-- Add a <p> tag to give the blog post content: -->
  <p>
    I flew by the sun, but I am still trapped in space.
  </p>
</div>
```

#8. Listing Liaison

Level Overview and Solutions

Intro

The `` tag is for **unordered lists**.

The `` tag is for **list items** inside of lists like the `` tag.

Default Code

```
<!-- <ul> tags are for grouping <li> tags. -->
<!-- <ul> stands for "Unordered List" -->
<!-- <li> are individual list items. -->

<h1>Lists of Animals</h1>
<h2>Animals With Wings</h2>
<ul>
  <li>Bees</li>
  <li>Flying Antelopes</li>
  <li>Pterodactyl</li>
  <!-- Add another animal inside a <li> tag. -->

</ul>
<!-- Add another <h2> tag to define a new list -->
<!-- Add a <ul> tag to group <li> elements. -->
<!-- Add 3 or more <li> tags with names of animals! -->
```

Overview

Unordered Lists

The `` tag is for **unordered lists**. Perfect for a grocery list, or, a list of animals. For the computer to know what items are apart of a list, be sure to use the `` tag which stands for **list item** before the closing `` tag.

```
<ul>
  <li>Red</li>
  <li>Blue</li>
  <li>Green</li>
</ul>
```

Listing Liaison Solution

```
<!-- <ul> tags are for grouping <li> tags. -->
<!-- <ul> stands for "Unordered List" -->
<!-- <li> are individual list items. -->

<h1>Lists of Animals</h1>
<h2>Animals With Wings</h2>
<ul>
  <li>Bees</li>
  <li>Flying Antelopes</li>
  <li>Pterodactyl</li>
  <!-- Add another animal inside a <li> tag. -->
  <li>Pugs</li>
</ul>
<!-- Add another <h2> tag to define a new list -->
<h2>Annoying Animals</h2>
<!-- Add a <ul> tag to group <li> elements. -->
<ul>
  <!-- Add 3 or more <li> tags with names of animals! -->
  <li>Wasps</li>
  <li>That cat breed with no fur</li>
  <li>Galactic Bears</li>
</ul>
```

#9. Orders Wanted

Level Overview and Solutions

Intro

Ordered lists (`` automatically add numbers as bullet points for ``).

```
<h2>Steps for a delicious dinner</h2>
<ol>
  <li>Apply sugar</li>
  <li>Bake at 400 degrees</li>
  <li>Eat when cool</li>
</ol>
```

Default Code

```
<!-- The <ol> tag is used to create ordered lists. -->
<!-- It uses <li> still, but includes a number instead. -->

<h1>Top 4 Most Wanted</h1>
<h2>Enemies</h2>
<ol>
  <li>Headhunter Paul</li>
  <li>Munchkin #41</li>
  <li>Cow</li>
  <!-- Add another enemy here: -->

</ol>
<!-- Add another <h2> for Friends: -->

<!-- Add a new <ol> to contain those friends: -->

  <!-- Add 3 or more <li> elements inside the <ol>: -->
```

Overview

 Tags

The `` tag stands for *ordered list*, and is used to add numbers before `` *list items*.

Use to create a numbered list when order is important!

```
<ol>
  <li>Gold Medalist</li>
  <li>Silver Medalist</li>
  <li>Bronze Medalist</li>
</ol>
```

Results in:

1. Gold Medalist
2. Silver Medalist
3. Bronze Medalist

Orders Wanted Solution

```
<!-- The <ol> tag is used to create ordered lists. -->
<!-- It uses <li> still, but includes a number instead. -->

<h1>Top 4 Most Wanted</h1>
<h2>Enemies</h2>
<ol>
  <li>Headhunter Paul</li>
  <li>Munchkin #41</li>
  <li>Cow</li>
  <!-- Add another enemy here: -->
  <li>Evil Robots</li>
</ol>
<!-- Add another <h2> for Friends: -->
<h2>Friends</h2>
<!-- Add a new <ol> to contain those friends: -->
<ol>
  <!-- Add 3 or more <li> elements inside the <ol>: -->
  <li>Marcia</li>
  <li>Jan</li>
  <li>Cindy</li>
</ol>
```

#10. Stylish Intent

Level Overview and Solutions

Intro

The `<style>` tag can be used to **style** how the various elements behave.

It follows a special set of rules!

Default Code

```
<!-- The <style> tag style your HTML with CSS. -->
<!-- Check the hints for more information on CSS! -->

<h1>The News</h1> <!-- Name the newspaper. -->
<style>
  h1 {
    color: red;
    text-decoration: underline;
    /* Align to center using the text-align: */

  }
  p {
    text-align:center;
    /* Give all paragraph text a color: */
    /* Colors can be purple, orange, blue, etc! */

  }
</style>
```

Overview

Cascading Style Sheets

Cascading Style Sheets, or CSS, is the web's way of formatting the various parts of a website.

To include special style rules, use the opening and closing `<style>` tag. **Note: `<style>` is unlike other HTML tags! What you put between the two tags is different from other HTML tags!**

```
<style>
p /* This is the element selector */ { /* Use { } to contain CSS rules */
    color: red; /* The attribute name comes first, followed by value it should be set to. */
    /* Be sure to use a : between the name and value. Use a ; at the end! */
}
div /* Selector */ { /* Brackets { } */
    text-align: center; /* Attribute Name : Attribute Value */
    /* Note the : and ; */
}
img {
    width: 100px;
    height: 200px;
}
</style>
```

Stylish Intent Solution

```
<!-- The <style> tag style your HTML with CSS. -->
<!-- Check the hints for more information on CSS! -->

<h1>The News</h1> <!-- Name the newspaper. -->
<style>
  h1 {
    color: red;
    text-decoration: underline;
    /* Align to center using the text-align: */
    text-align: center;
  }
  p {
    text-align:center;
    /* Give all paragraph text a color: */
    /* Colors can be purple, orange, blue, etc! */
    color: maroon;
  }
</style>
```

#11. Classy

Level Overview and Solutions

Intro

HTML and CSS make use of the idea of `classes`. A `class` is a way of grouping similar elements by giving them a repeatable name.

```
<style>
    /* The "." before the class name tells the CSS code only to edit those with class "inner-div" */
    .inner-div {
        color: blue;
    }
    /* The "." before the class name tells the CSS code only to edit those with class "outer-div" */
    .outer-div {
        background-color:green;
    }
    /* A tag selector will select all elements with that specific tag! */
    p {
        color: red;
    }
</style>
<div class="outer-div">
    <h2>I'm a sub-title!</h2>
    <div class="inner-div">
        <p>Hello, world!</p>
    </div>
</div>
<div class="outer-div">
    <h2>I too, am a subtitle!</h2>
    <div class="inner-div">
        <p>Bonjour, monde.</p>
    </div>
</div>
```

Default Code

```
<!-- The "class" attribute makes styling easier. -->
<!-- Use it to style specific repeated elements! -->

<style>
  /* The "." tells CSS to target a "class" name. */
  .character {
    background-color:orange;
  }
</style>

<div>
  <h1>Romeogre and Juliet</h1>
  <h2>The Cast</h2>
</div>
<!-- The class attribute has been added to these <div> -->
<div class="character">
  
  <h3>Cindy Archer</h3>
  <p>as Juliet</p>
</div>
<!-- Include the main star Romeogre! -->
<!-- Add another <div> with the "character" class: -->
```

Overview

Class Attribute

The `class` attribute is used to make styling elements easier. Assign the attribute to an element by including it inside an element's opening tag. Inside the `<style>` element, target specific classes using the `. + className` CSS selector.

```
<style>
  /* This sets all elements with the "neat" class to the color: blue. */
  .neat {
    color: blue;
  }
</style>
<!-- This <div> element has the "neat" class. -->
<div class="neat">
  <p>Hello, world!</p>
</div>
```

Classy Solution

```
<!-- The "class" attribute makes styling easier. -->
<!-- Use it to style specific repeated elements! -->

<style>
  /* The "." tells CSS to target a "class" name. */
  .character {
    background-color: orange;
  }
</style>

<div>
  <h1>Romeogre and Juliet</h1>
  <h2>The Cast</h2>
</div>
<!-- The class attribute has been added to these <div> -->
<div class="character">
  
  <h3>Cindy Archer</h3>
  <p>as Juliet</p>
</div>
<!-- Include the main star Romeogre! -->
<!-- Add another <div> with the "character" class: -->
<div class="character">
  
  <h3>Borp</h3>
  <p>as Romeogre</p>
</div>
```


#12. Identification, Please

Level Overview and Solutions

Intro

The alternative to the CSS `class` is `id`. `id` attributes mark elements with a specific name to recall later. While a `class` can be repeated across multiple elements, and `id` is for **one** specific element. Think of it as an piece of identification like a school ID or driver's license.

```
<style>
  #main {
    color: orange;
  }
</style>
<div id="main">
  There can only be one!
</div>
```

Default Code

```
<!-- Specific tags can be given an "id" attribute. -->
<!-- The "id" attribute should be a unique identifier! -->

<style>
    /* The "#" modifier is used to target an "id". */
    #element1 {
        color:red;
    }
    #element2 {
        /* Change this element's text-align: */
    }
    /* Add a rule for #element3 */

    /* Change #element3's background-color: */

</style>
<div class="descriptive">
    
    <h2>Archer Person</h2>
    <ul>
        <li id="element1">Has a hood.</li>
        <li id="element2">Shoots a bow.</li>
        <li id="element3">Carries arrows.</li>
    </ul>
</div>
```

Overview

ID Attribute

The `id` attribute is to assign a unique **identifier** to an element. This makes it easier to target inside your style.

```
<style>
  /* This sets the element with id 'main' to a red background color. */
  #main {
    background-color:red;
  }
</style>
<!-- Notice the id attribute! -->
<div id="main">
  Hello, world!
</div>
```

Identification, Please Solution

```
<!-- Specific tags can be given an "id" attribute. -->
<!-- The "id" attribute should be a unique identifier! -->

<style>
  /* The "#" modifier is used to target an "id". */
  #element1 {
    color:red;
  }
  #element2 {
    /* Change this element's text-align: */
    text-align: center;
  }
  /* Add a rule for #element3 */
  #element3 {
    /* Change #element3's background-color: */
    background-color: purple;
  }
</style>
<div class="descriptive">
  
  <h2>Archer Person</h2>
  <ul>
    <li id="element1">Has a hood.</li>
    <li id="element2">Shoots a bow.</li>
    <li id="element3">Carries arrows.</li>
  </ul>
</div>
```


#13. Wanted Poster

Level Overview and Solutions

Intro

Adjust the `` tags, text, and styles to customize a wanted poster for the most notorious CodeCombat outlaws.

Default Code

```
<style>
  h1 {
    color: red;
  }
  h1 {
    font-size: 7em;
  }
  h2 {
    /* Change the number to change the size of the header text. */
    font-size: 3em;
  }
  h3 {
    font-size: 1.5em;
  }
  li {
    font-size: 1.5em;
  }
</style>
<h1>Wanted</h1>

<!-- Find an image from the Image Gallery above, or use your own image link. --&gt;
&lt;img src="/file/db/thang.type/53ed1d9c2b65b0e32b9c96a9/portrait.png" id="most-wanted"&gt;

<!-- Who is wanted? --&gt;
&lt;h2&gt;Who?&lt;/h2&gt;

<!-- What are they wanted for? --&gt;
&lt;h3&gt;For Reckless, Inhuman Crimes:&lt;/h3&gt;
&lt;ul&gt;
  &lt;li&gt;Crime 1&lt;/li&gt;
  &lt;li&gt;Crime 2&lt;/li&gt;
&lt;/ul&gt;

<!-- What is the reward? --&gt;
&lt;h2&gt;Reward: What?&lt;/h2&gt;

<!-- Who are their partners in crime? --&gt;
&lt;h3&gt;Known Accomplices&lt;/h3&gt;
&lt;div&gt;
  &lt;div&gt;
    &lt;img src="/file/db/thang.type/53ed1d9c2b65b0e32b9c96a9/portrait.png"&gt;
    &lt;h3&gt;Who?&lt;/h3&gt;
  &lt;/div&gt;
&lt;/div&gt;</pre>
```

```

<h3>Who?</h3>
</div>

<!-- Add a third accomplice here. Choose whether they are class "captured". -->
</div>

<!-- How do want them captured? -->
```

Overview

This is an open ended project! Try to accomplish the goals, to make something to show your family and friends.

If it don't make sense, that's how you know you're web developmenttering!



Wanted Poster Solution

```
<style>
  h1 {
    color: red;
  }
  h1 {
    font-size: 7em;
  }
  h2 {
    /* Change the number to change the size of the header text. */
    font-size: 3em;
  }
  h3 {
    font-size: 1.5em;
  }
  li {
    font-size: 1.5em;
  }
</style>
<h1>Wanted</h1>

<!-- Find an image from the Image Gallery above, or use your own image link. --&gt;
&lt;img src="https://www.codecombat.com/file/db/thang.type/54eb540b49fa2d5c905ddf1a/portrait.png" id="most
-wanted"&gt;

<!-- Who is wanted? --&gt;
&lt;h2&gt;Bomb Bill&lt;/h2&gt;

<!-- What are they wanted for? --&gt;
&lt;h3&gt;For Reckless, Inhuman Crimes:&lt;/h3&gt;
&lt;ul&gt;
  &lt;li&gt;Exploding&lt;/li&gt;
  &lt;li&gt;Rolling&lt;/li&gt;
  &lt;li&gt;Priming&lt;/li&gt;
  &lt;li&gt;Smoldering&lt;/li&gt;
&lt;/ul&gt;

<!-- What is the reward? --&gt;
&lt;h2&gt;Reward: Some Gold&lt;/h2&gt;

<!-- Who are their partners in crime? --&gt;
&lt;h3&gt;Known Accomplices&lt;/h3&gt;
&lt;div&gt;
  &lt;div&gt;
    &lt;img src="https://www.codecombat.com/file/db/thang.type/5432f9d18364d30000d1f943/portrait.png"&gt;
    &lt;h3&gt;Chest o' Gems&lt;/h3&gt;
  &lt;/div&gt;
&lt;/div&gt;</pre>
```

```
</div>

<div>
  
  <h3>The Hand</h3>
</div>

<!-- Add a third accomplice here. Choose whether they are class "captured". -->
<div>
  
  <h3>cow</h3>
</div>
</div>

<!-- How do want them captured? -->
```