

1.4.6 Style the "What You Do" Section

The setup for this next section is going to be similar to the last one we tackled. First, add a class to the `<section>` element:

```
<section id="what-you-do" class="steps">
```

In your style sheet, let's knock out some of the higher-level styling for the overall section. In `style.css`, add the following CSS to align and add background color to the section:

```
.steps {  
  text-align: center;  
  background: #fce138;  
}
```

Then add some styling to the heading:

```
.steps h2 {  
  font-size: 55px;  
  color: #024e76;
```

```
margin-bottom: 35px;  
padding: 0 100px 20px 100px;  
border-bottom: 3px solid;  
border-color: #39a6b2;  
}
```

By the way, it's a good idea to keep all of the CSS rules related to the "What You Do" section close together in the style sheet so it's easier to find them. It would be bad practice to have one `.steps` rule at the top and several more scattered throughout the middle and bottom of the file.

Huh. You know what? Those CSS declarations look almost identical to the ones we wrote for `.intro h2`. The only thing that's different is the `border-color`. In programming, we want to cut down on duplicate code as much as possible, and this is a great example of unnecessary duplication.

HIDE PRO TIP

An important principle in software development that you might hear along the way is the term DRY, or Don't Repeat Yourself. To read about it, check out [Wikipedia's entry on DRY \(https://en.wikipedia.org/wiki/Don%27t_repeat_yourself\)](https://en.wikipedia.org/wiki/Don%27t_repeat_yourself).

We could consolidate these CSS rules into more generic classes that all of our headings (including "Meet the Trainers" in the next lesson) could pull from. Let's reorganize our CSS with a few new rules:

```
.section-title {  
  font-size: 55px;  
  color: #024e76;  
  margin-bottom: 35px;  
  padding: 0 100px 20px 100px;  
}
```

```
display: inline-block;
border-bottom: 3px solid;
}

.primary-border {
  border-color: #fce138;
}

.secondary-border {
  border-color: #39a6b2;
}
```

Now, because we've combined their declarations, we can delete the previous CSS rules for `.intro h2` and `.steps h2`:

```
/* remove this entire CSS rule */
.intro h2 {

}

/* and this one */
.steps h2 {

}
```

We also need to update the HTML code to use these new classes. In `index.html`, add the following class information beneath the opening `<section>` tags for each section:

```
<section id="what-we-do" class="intro">
  <h2 class="section-title primary-border">What We Do</h2>
  ...
</section>

<section id="what-you-do" class="steps">
  <h2 class="section-title secondary-border">What You Do</h2>
  ...
</section>
```

Note that these `<h2>` elements in different sections can use the same `section-title` class, which gives them all the same baseline in terms of size, spacing, etc. We can then add additional classes by separating the class names with a space. The attribute `class="section-title primary-border"` is actually pulling in CSS declarations from two different classes.

What's left is some general clean-up to size and color elements correctly. Even though this may feel like repetitive work, still type out each one instead of copy/pasting. It'll help you memorize the syntax faster!

Add the following CSS to `style.css`:

```
.steps div {
  margin-bottom: 80px;
}

.steps img {
  width: 15%;
  margin: 10px 0;
}

.steps h3 {
  color: #024e76;
  font-size: 46px;
  margin-top: 10px;
}

.steps p {
  color: #39a6b2;
  font-size: 23px;
}
```

DEEP DIVE ▲

DEEP DIVE

Something you might have noticed is that the parent `text-align: center;` property also centered the SVG icons. This can be a bit confusing at first because the property has "text" in its name, and images are not text. `text-align` essentially centers everything inside an element—text or otherwise. However, if you want to use `text-align` to center an image, make sure you apply it to the image's parent element, not the image itself.

It may seem like we're done now, but anytime we're given a design mock-up to follow, we need to be absolutely sure we covered everything. Designers don't appreciate when developers say, "Eh, it's close enough."

One thing that would be easy to overlook is the font size differences in the step titles, as shown here for Step 1:

Step 1: Fill Out The Form Above.

Note that "Step 1:" is a bigger font than "Fill Out The Form Above." Ugh, does that mean we have to create two separate elements and meticulously position them side by side? Thankfully, no. We can more easily solve this problem using another HTML element: ``.

In `index.html`, rewrite the HTML for the `<h3>` element in each step to have a `` element wrap a section of the text. Each one should look something like this code:

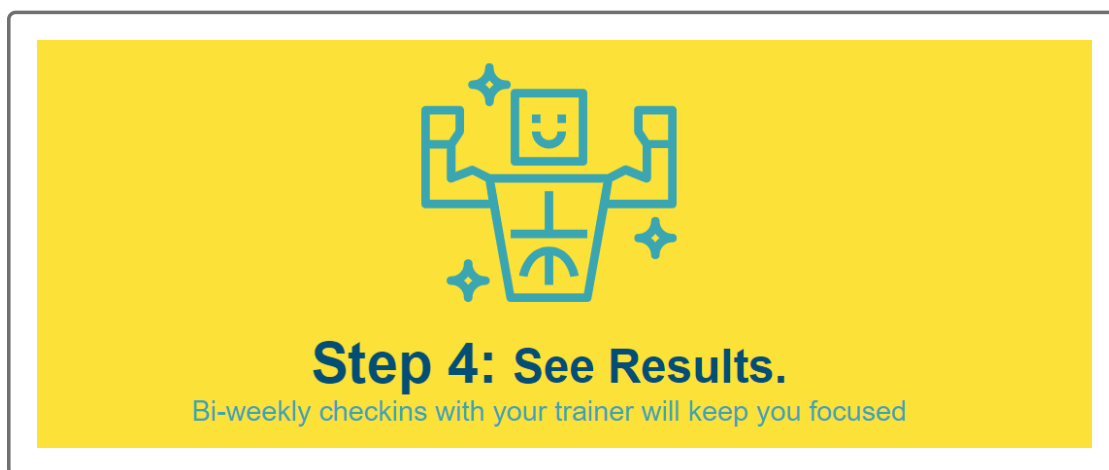
```
<h3>Step 1: <span>Fill Out the Form Above.</span></h3>
```

Refresh the browser. Nothing changed—what gives? This is because the `` element doesn't affect anything visually. It is an inline element meant for highlighting sections of larger text blocks, where the highlighting is something we define ourselves with CSS. Let's define the CSS for `` now.

In `style.css`, add the following:

```
.steps span {  
  font-size: 38px;  
}
```

This will shrink the font size of the `` while the adjacent text in the `<h3>` keeps its `46px` font size. Update Steps 2–4 to follow suit. When you're done, you should see some nice results:



Great job! Take a moment now to go through the following knowledge check:

© 2020 Trilogy Education Services, a 2U, Inc. brand. All Rights Reserved.

Question 1.1

What is the purpose of the `alt` attribute for images?

- ☐ To make the image load faster
- ☐ To make it easier to style the image with CSS
- ☐ To prevent search engines from indexing the image
- ☒ To provide context for the image



Response-specific feedback

Yes, this is particularly useful for users who rely on a screen reader

Question 1.2

How can you add more than one class to an HTML element?

- ☐ Add a second `class` attribute
- ☐ Add a comma between the class names
- ☒ Add a space between the class names
- ☐ Add a `class-2` attribute



Response-specific feedback

Yes, the syntax would look like this: `class="one two"`

Question 1.3

What CSS declaration could you add to a 50%-width `<div>` to center it?

- ☐ `text-align: center`
- ☒ `margin: 0 auto`
- ☐ `float: center`
- ☐ `align: center`



Response-specific feedback

Yes, this will tell the browser to calculate the margins for you, thus centering the element