

2.3.5 Make Initial Site Edits

It's time to make the site fully responsive. When it comes to making a site look good on all devices and screen sizes, there are usually two routes to take:

1. We can adjust our current CSS styles so they look good on all devices by finding lengths and sizes that aren't too big for mobile and aren't too small for a computer screen. This isn't always a viable option, depending on the overall design of the site we're building, but it does reduce the amount of code we write.
2. Create CSS rules for those screen sizes using media queries. This is primarily used when we need to drastically change how the page looks on different devices, like make two columns stack on top of one another instead of side by side. They're a bit more involved to set up but not difficult to use, so we'll go with this approach.

Before we move on to implementing the second method, let's use the first one to save ourselves from doing more work than we need to later. Instead of having these values adjusted at different screen sizes, we can adjust some of them now to strike a nicer balance across all devices. We'll start by changing some of our current styles.

- Find the CSS rule for `header nav ul li a`:

- Replace the `margin` with a `padding` of `10px 15px` to create a tighter navigation menu
- Find the CSS rule for `.hero`:
 - Add an `align-items: flex-start` declaration to it
- Find the CSS rule for `.hero-cta`:
 - Change the `line-height` to `1.2` to make the text easier to read
- Find the CSS rule for `.section-title`:
 - Change the `font-size` property from 55px to 48px, which is still plenty big for all screens but the current value of 55px on a mobile device may seem a little cramped

PAUSE

If there is no `align-items` property set for a flexbox, what is the default value and how does that affect the two flex children?

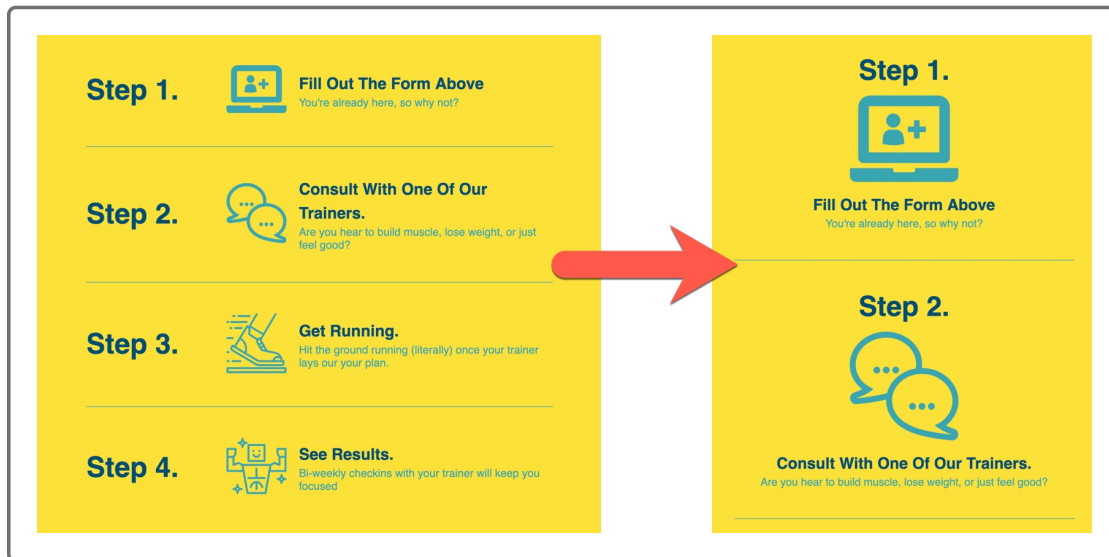
[Show Answer](#)

These changes may seem minor at the moment, but keep in mind that we're trying to find a nice balance for these styles so they look good on any device. Sometimes what seems negligible on one screen makes a world of difference on another.

Of the two methods described above, we just used the first one by updating styles that apply to all screen sizes. But there are several parts of the UI that simply cannot look good at different sizes without some major changes. For example, if we attempt to make the page smaller, the newly

laid out "What You Do" steps will have a hard time sharing the horizontal space and will get very tight looking.

Let's fix it by stacking the content vertically when viewed on a mobile device, as shown in this image:



The above image's shift in layout cannot be achieved by simply adjusting our current CSS, so we need to add to our style sheet. This is where media queries come in.