

CSS for Mobile Devices Lab

You and your partners have decided to use your site as the go-to site on all devices. Let's make sure it looks good on any sized device.

Prepping for responsive development

1. Let's do an experiment. Add this to the bottom of site.css:

```
@media (max-width:768px) {  
  body {  
    background-color: yellow;  
  }  
}
```

2. Browse to any page on your site and narrow the window to less than 768 pixels. You should see the background change to yellow when you do. This is a trick developers use while testing so we can instantly see where the break is. Leave that in until the end of the lab if you like. You'll want to remove it eventually.

The register page

We have a register page with four sections. We've decided that the page looks great on a phone or tablet but on a laptop or bigger we have plenty of room to put the two middle sections side-by-side.

3. Look at your register page and add ids to the Account Information and Address Information sections.
4. Our goal is for these two sections to be 50% width and float: left if the screen is 992 pixels or larger. Edit site.css and add that new "band" to a media query at the bottom of the stylesheet. (Hint: To un-float something, use clear: both; and set width: 100%;).
5. Let's test it out. Open register.html in a browser and adjust the width. When they stack on small devices but appear side-by-side on large devices, you've done it!

Responsive featured items

Our main page holds a featured items section. And while it may look good on large screens, four items across will be too hard to read on a phone. Let's fix that.

6. Remind yourself of the id of the featured items section's flex container.
7. Edit site.css. Find your media query for phone-sized screens ($\leq 768\text{px}$).
8. Set the flex-direction for that flex container to column.
9. That should do it. It was easy because we used a flex container. Look at it in a browser to confirm that on a phone-sized screen, the featured items are stacked and on anything larger, they're side-by-side.

Making index.html lighter on phones

Let's say that our market research team has realized that when people browse to us on a phone, they are distracted by our *Why Us* section. Let's remove that on a phone.

10. Change your phone-size media query to set that section's display property to "none".
11. Test it out. That section should be seen on larger width screens but simply disappear on smaller widths.

Adjusting the header of all pages

The logo and header just don't work on smaller devices. We went out of our way to put these things side-by-side earlier. We like the look on large screens but on smaller devices, let's make the logo bigger and put it on a line by itself.

12. Open index.html in an editor and remind yourself of the structure of the header, logo, h1, h2 and the paragraph.
13. Edit site.css. Add to your phone-size media query. When the screen is phone-sized or smaller, make the logo's max-width 100% and set float to *none*. You'll also need to set the headerH1's width back to *auto* and its float to *none*.
14. Once again, test it out. On a small screen the logo should be stacked above the header but on a larger screen they should be side-by-side.

Congrats! You have a responsive site. There are lots of other things we could do to our site, but this will suffice for learning purposes.