CSS Layout Lab

Now that we have the sections themselve positioned nicely we should turn our attention the getting the elements on the page laid out in relation to one another.

Floating sections

Let's work on the top of every page. Let's say we want the logo next to the <h1> and <h2>. The <h1> is a block element so it has a break before and after. One way we can fix this is to float the image.

1. First, we want the <h1> and <h2> to remain together. So wrap them in a div with a class of *headerH1*.
2. Add a new empty div after your headerH1 div. Give it a class of "clearfix".
3. Add a class of *headerImg* to the image.
4. Next add styles to site.css. For the headerImg class, add a width of about 25%. And for the headerH1 class, add a width of about 75%. You may have to adjust these for padding later.

If you look now, they're still one on top of the other. Here's the magic...

5. Add to both of those styles *float: left;*
6. Now refresh and you'll see they're side-by-side. You may not like how the paragraph floats up also. To fix that add a style for the clearfix class. It should clear: both;.
7. Refresh again and adjust the widths until you like how they look.
8. Go ahead and add that new structure and those classes to all pages.

Using inline-block

Open index.html in a browser. Look at the footer. Notice that you have links one on top of another. This is because s are block elements. Let's put them side-by-side.

9. Edit your html and give each of these s a class of *list-inline*.
10. Now edit site.css and create a new style for that class and set the display property to inline-block;
11. Refresh your page. They should now all be side-by-side. Feel free to drop in some padding to spread it out nicely.

Using flexbox

On your index page you have four divs, each with links to our featured products and an image. These things would look better side-by-side. Let's do that with flexbox.

12. Open index.html in an editor. Find those four divs.
13. To each of them, add a class of featuredFlexItem.

14. Since flexbox requires a container, wrap the four divs in a bigger div. Give it an id of featuredFlexContainer.
15. Open site.css. Add a new style for that id. It should set display to *flex*.
16. Take a look at your page in a browser now. See any difference? You should!
17. Add a style for everything with a class of featuredFlexItem. Set flex-basis to 25%, flex-grow to 1 and flex-shrink to 1.
18. Run and test it again, you should see the sections remain uniform even when the width of the page changes.

19. Bonus!! At the top of every page you have an unordered list of links that is serving as your navigation. These are block elements so they're on top of one another. Try to get these things side-by-side. Use any of the above techniques you like.