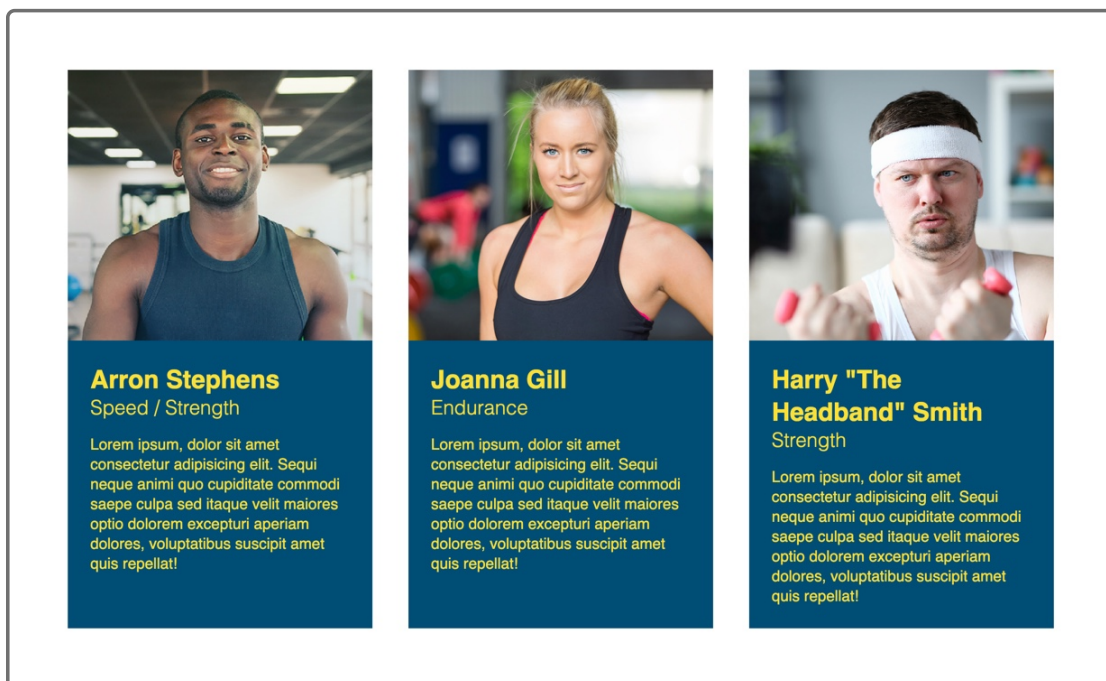


2.2.10 Trainer Trading Cards

The design team at Run Buddy must really love the idea of tightening up sections to take up less vertical space because we're about to give our trainers a treatment similar to the "What We Do" section we just finished up. This one requires less granular control, so it should go more quickly.

Our current section for the trainers has them stacked on top of one another, but we want them to fit in one row:

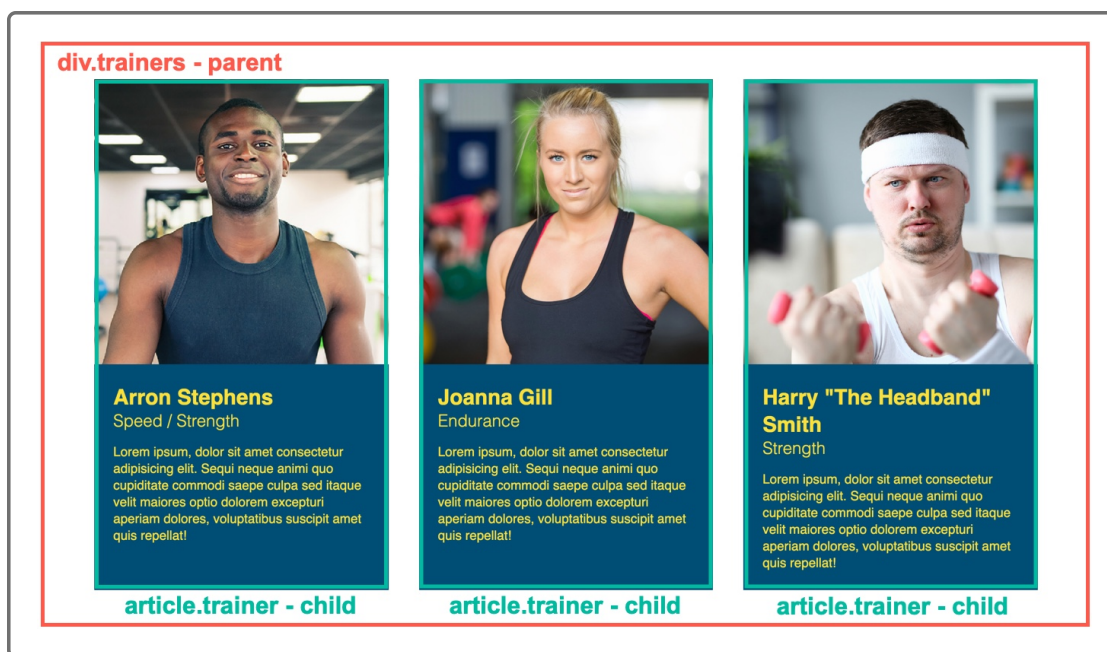


To do this, we'll change each trainer's `<article>` element from a side-by-side landscape layout to a top-to-bottom portrait layout.

NERD NOTE

When related HTML content is laid out in a narrow, vertical column, it's often referred to as a **card**.

Let's take the mock-up and wireframe it. It should look something like this:



We'll start on the easier side this time and update how the text content looks first:

- Find the CSS rule for `.trainer-bio h3`
 - Remove the declaration for `margin-bottom`
 - Change the `font-size` value to 28px

- Find the CSS rule for `.trainer-bio h4`
 - Change the `font-size` value to 22px
 - Change the `margin-bottom` value to 15px
- Find the CSS rule for `.trainer-bio p` and move its `line-height` declaration to the `.trainer-bio` rule

The next thing we want to do is go into the `index.html` file and move the second trainer's `` element above the `<div>` with a class of `trainer-bio` so it matches the other two trainers. In Run Buddy version 1.0, our three trainers alternated when it came to their image placement. For Run Buddy version 2.0, they will all have the same "image then text" layout.

When that's done, look through all three trainer `<article>` elements and remove any classes that mention `text-left` or `text-right` because we don't need those anymore. At the same time, add a `trainer-name` class to the `<h3>` and a `trainer-role` class to the `<h4>`.

They should now all have the following HTML structure (with varying content, of course):

```
<article class="trainer">
  
    <h3 class="trainer-name">Arron Stephens</h3>
    <h4 class="trainer-role">Speed / Strength</h4>
    <p>
      Lorem ipsum, dolor sit amet consectetur adipisicing elit. Sequi
      itaque velit maiores optio dolorem excepturi aperiam dolores, vo
    </p>
  </div>
</article>
```

Another thing we need to do is create a flexbox container around these three `<article>` elements, so wrap them with a `<div>` element so it looks

something like this:

```
<div>
  <!-- This is for show; keep the article tags you have -->
  <article>Trainer 1 Info</article>
  <article>Trainer 2 Info</article>
  <article>Trainer 3 Info</article>
</div>
```

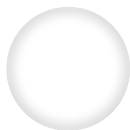
Lastly, we need to give this element a class, so remove the `class="trainers"` from the parent `<section>` element and move it to the `<div>` that was just created. We can do this because the `<section>` element doesn't need to be selected by a class anymore using CSS.

The only thing left to be done is to turn these trainer cards into a row, so turn your attention back to `style.css` and give the `.trainers` CSS rule the flexbox treatment:

```
.trainers {
  width: 100%;
  margin: 0 auto;
  display: flex;
  flex-wrap: wrap;
  justify-content: space-around;
}
```

Next, find the CSS rule for `.trainer` and make the following changes:

- Remove `width` and `overflow`
- Change `margin` to just 20px
- Add `flex: 1`



REWIND

If all flexbox children have a value of `flex: 1`, that means they will share any extra space evenly.

The result should look like this image:



It doesn't look great just yet, but that's because we have some styles in our newly created flexbox children (the `<article>` elements) that we need to adjust or remove:

- Find the CSS rule for `.trainer img`:
 - Remove `float`
 - Change `width` to 100%
- Find the CSS rule for `.trainer-bio`:

- Remove `float` and `width`
- Change `padding` to 25px

Save your changes and refresh the page again. The trainers should now look just like the mock-up!

This is a very common layout technique in modern web design. It's considered a "three-column card layout", meaning we created one horizontal row that can hold on to three separate pieces of information.

A huge takeaway from this lesson is the importance of wireframing. You can always search for an answer or syntax hint on the internet, but you still have to know the right question to ask to get that result.

There's only one more section left to fix. Before we do that let's add, commit, and push our code with the following commands:

```
git add .  
git commit -m "add flexbox to the trainers"  
git push origin feature/flexbox
```