

## 4.3.4 Add New HTML Lists

As we introduced earlier, a **Kanban board** is a tool often used in productivity apps to visually convey the current stage of all tasks in a project. This is done by defining columns that tasks can move through from left to right. Some projects may define columns for To Do, In Progress, Code Review, Testing, Completed, and Blocked. We'll keep our app simple by only using three columns: Tasks To Do, Tasks In Progress, and Tasks Completed.

### ON THE JOB

There are many Kanban-style project management apps that you'll come across in your career. Some of the more popular ones that companies use are **Trello** (<https://trello.com/>) and **Jira** (<https://www.atlassian.com/software/jira>).

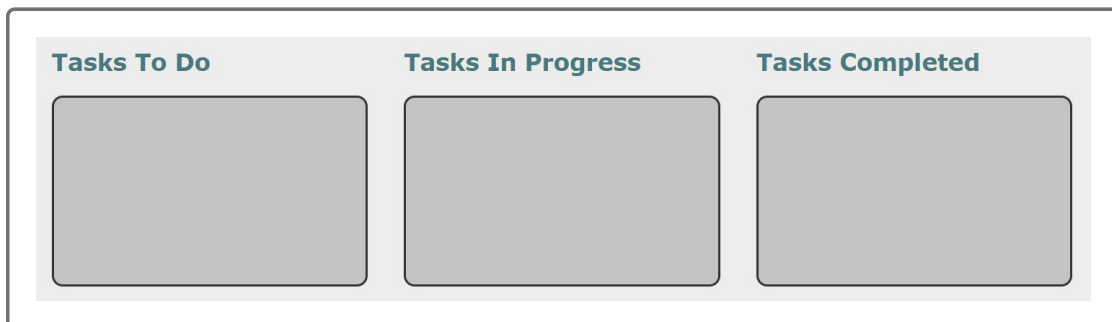
We already have HTML in place for the first column, Tasks To Do, represented by a `<section>` element with class `task-list-wrapper`. Let's revisit the HTML in `index.html` and add two more `<section>` elements underneath the first one:

```
<!-- Tasks In Progress -->
<section class="task-list-wrapper">
  <h2 class="list-title">Tasks In Progress</h2>
  <ul class="task-list" id="tasks-in-progress"></ul>
</section>

<!-- Tasks Completed -->
<section class="task-list-wrapper">
  <h2 class="list-title">Tasks Completed</h2>
  <ul class="task-list" id="tasks-completed"></ul>
</section>
```

Note that the `<section>` elements all have the same class, but the inner `<h2>` text and `id` attributes on the `<ul>` elements are different.

Save the `index.html` file and refresh the browser. The page should now look like this:



## PAUSE

What CSS features can make a layout of columns?

Flexbox, CSS grid, or even the `float` property.

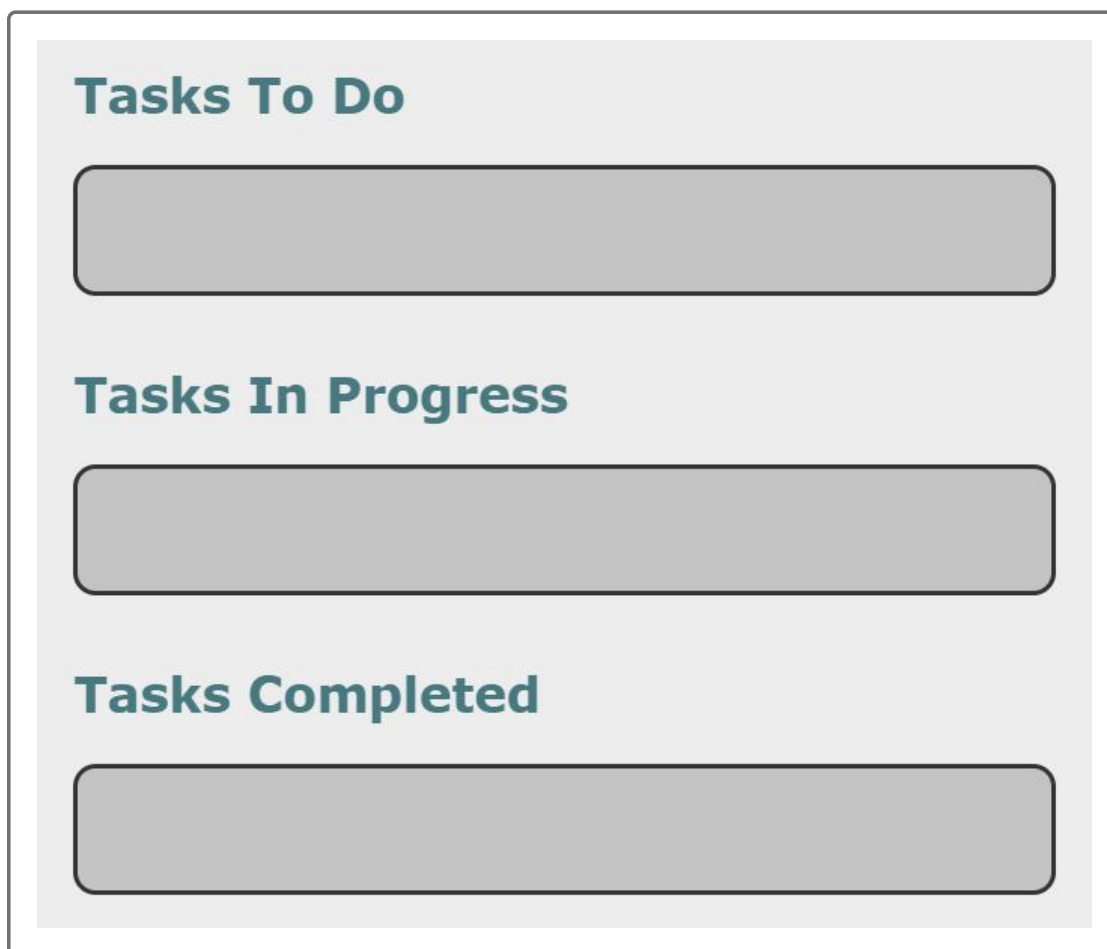
[Hide Answer](#)

As a quick CSS refresher, use the Chrome DevTools to inspect the columns and verify which CSS properties are being used. You'll notice that the `<main>` element has the declaration `display: flex`, allowing it to control the distribution of its content. In turn, each `<section>` element has a `flex: 1` declaration to specify that they should share space evenly.

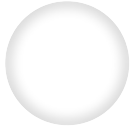
If you haven't already, try toggling the device toolbar from the DevTools by clicking on the following icon:



The browser should then mimic what the webpage looks like on a mobile device:



In the CSS media query, the `<section>` elements' `flex-basis` is set to `100%`, which defines a new width that takes up all available space. Thus, the elements become stacked.



## REWIND

---

A media query defines a set of CSS rules that won't be applied until a certain condition is met. For example, `@media screen and (max-width: 900px)` kicks in when the screen size is less than 900 pixels wide.

Fortunately, all of this CSS was provided ahead of time, but it's always helpful to understand what's happening, whether you wrote the code yourself or not.

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