

# **Big Project**

## **Part 1**



# What are we going to do today?

<

>



**Go over some info**

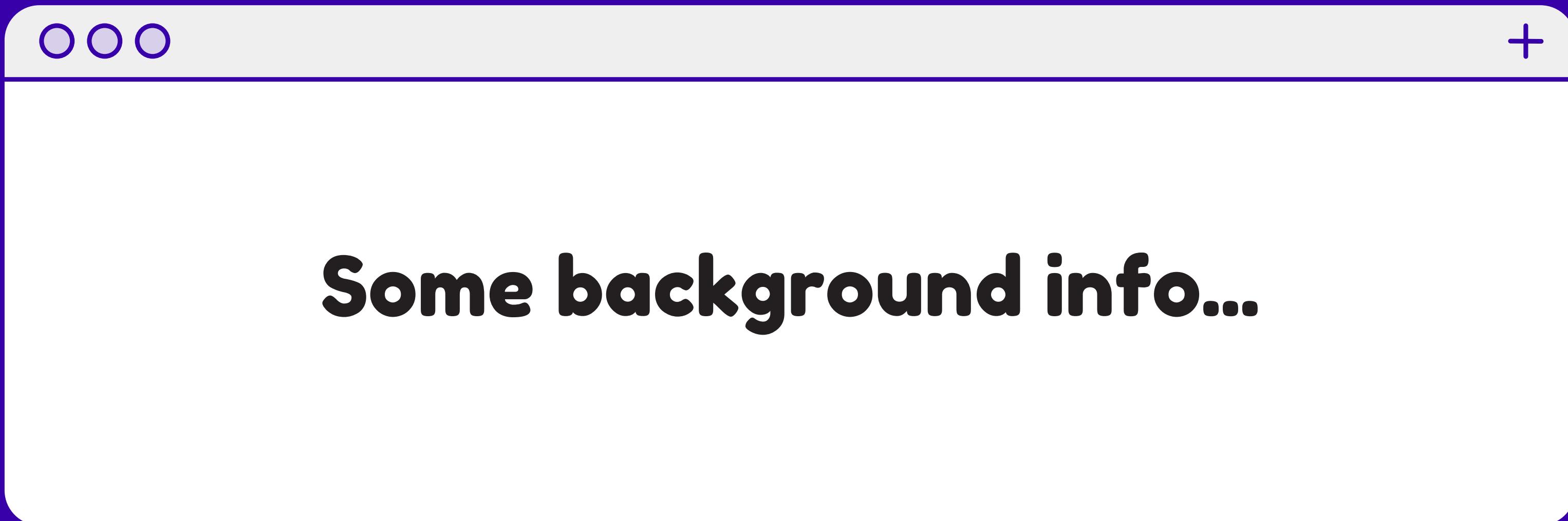
**Build the UI**

**Flexbox**

**Adapt scrolling**

**Personalize UI**

We hope you are having fun! ;)



# What is our Kanban board, really?

- A series of **columns** where tasks go from one column to the next.
- In code, we can represent this as **unordered lists** to build the four columns.
  - Each column will then have its own **unordered list** of task items.



# But how do we implement this in code?

- HTML: we add all the structure for the columns within our `<body>` `</body>`
- We need to create a "container" for our columns:
  - Check out the template in CSS. Can you find any class called `drag-container`?



# We said each column is a list...

So we have to create our lists, which are also classes!

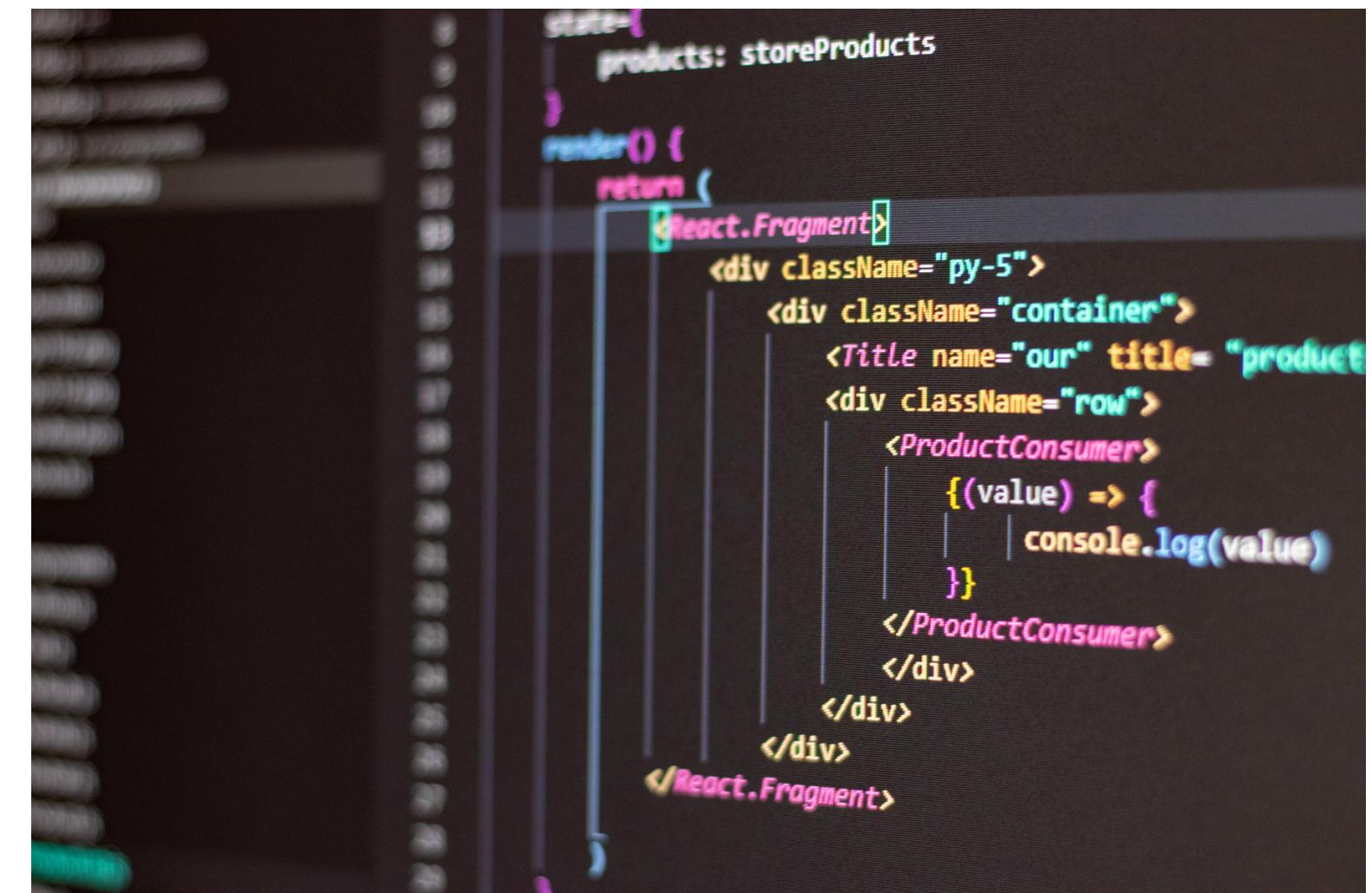
- In HTML, lists are referred to as "li".
- We can apply functions to our list by typing:
  - `li.drag-column-backlog-column`

★ ★ *Do it for* ★  
★ *yourself* ★ ★

# Some more HTML

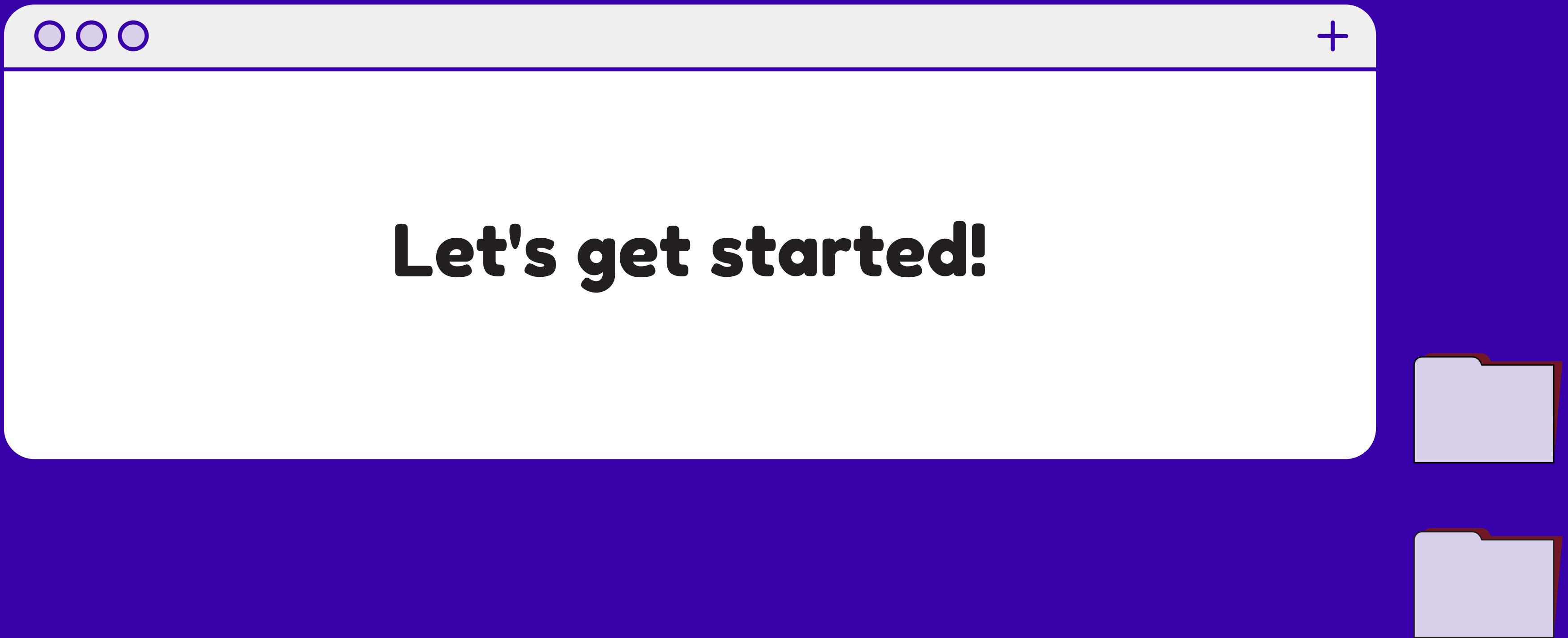
- ul: unordered list
- adding classes to lists using .classname
- we use span to create headers
- To add content:
  - we want another division
  - adding ids to items using #

Remember we use classes declared in CSS to create the structure in HTML!



```
state={ products: storeProducts }
}
render() {
  return (
    <React.Fragment>
      <div className="py-5">
        <div className="container">
          <Title name="our" title="product" />
          <div className="row">
            <ProductConsumer>
              {(value) => {
                | console.log(value)
              }}
            </ProductConsumer>
          </div>
        </div>
      </React.Fragment>
    )
}
```

We hope you are having fun! ;)

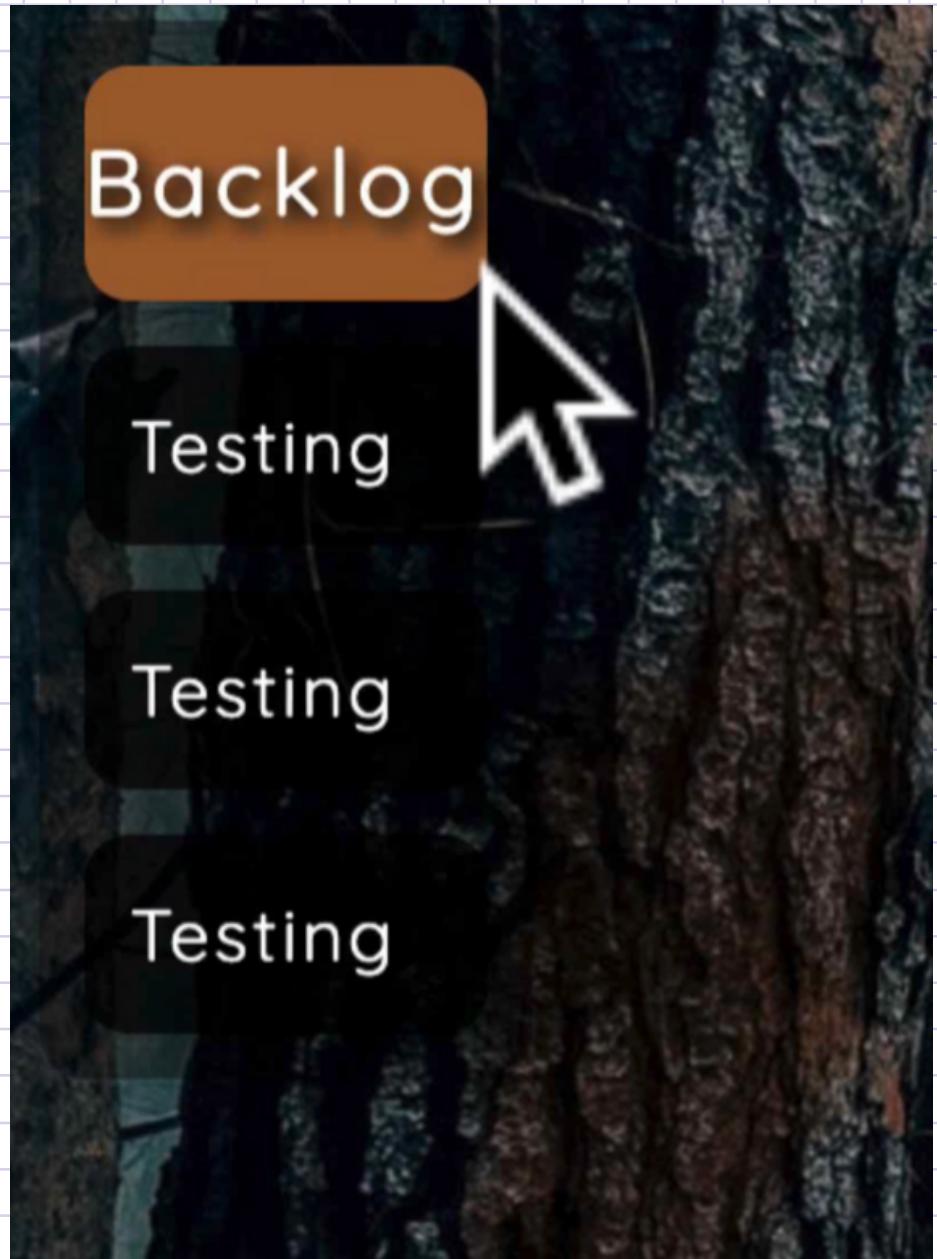


If you were not here last week ...



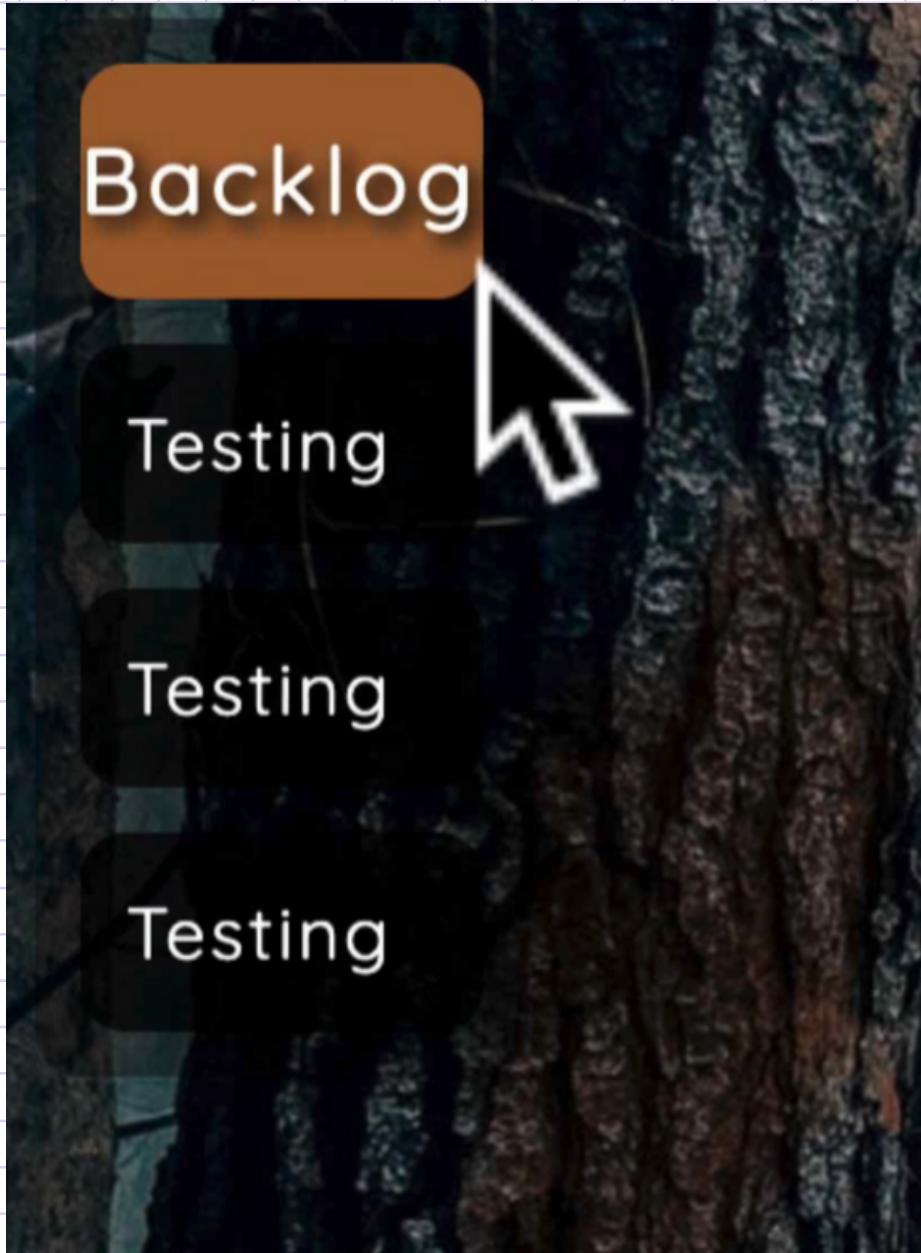
# Building Columns

# Our first column!

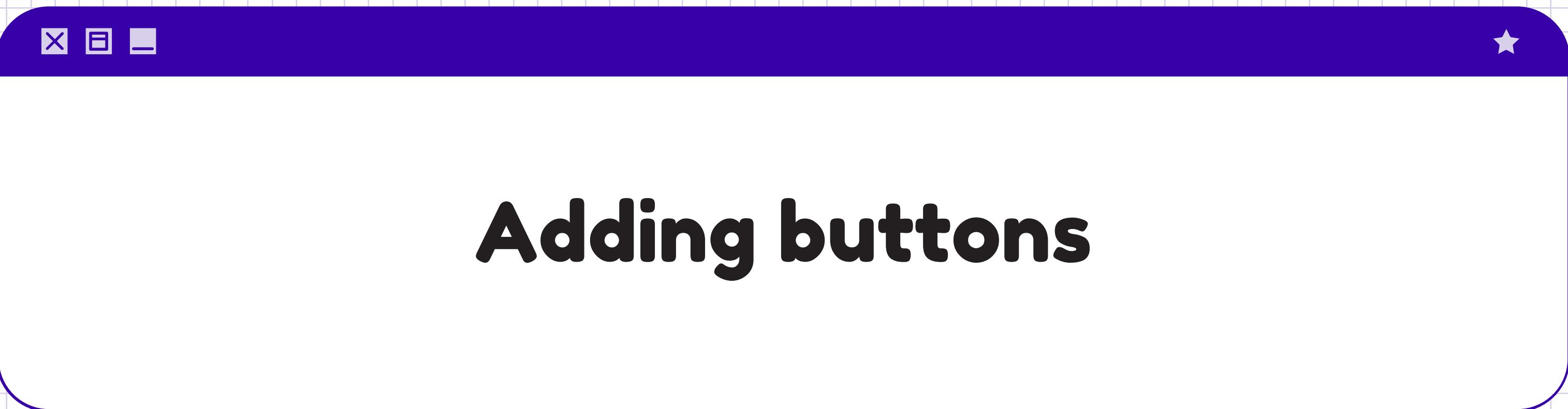


- Below our header, create a division with class drag-container to hold our columns.
  - Inside our container, create a list of class drag-list.
- Create our backlog column:
  - use classes drag-column and backlog column
  - create a header using <span> and add a name using <h1>
- Now for the content
  - Create a parent div with id "backlog-content"
  - Create an unordered list inside the div with id "backlog-list"
  - Add list items: <li class = "drag-item">Testing</li>

# Any problems?

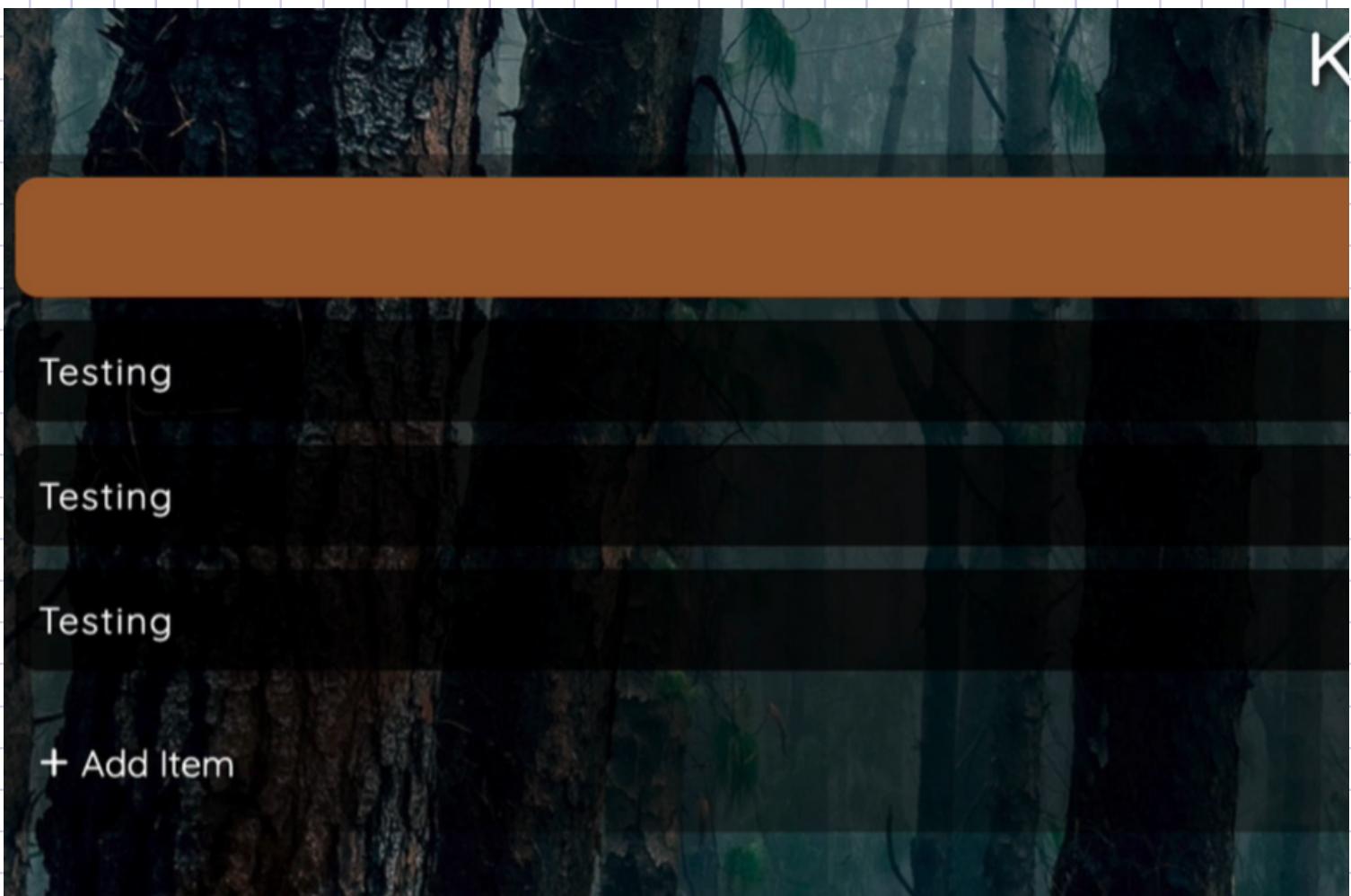


- We want our columns to fill the space in the web page.
  - This means we have to change something in our CSS file...
- 
- Find the class: `.drag-column`
  - This is the class that is on every column in our board!
  - We have to add `flex:1;`
    - Evenly divide space between any drag-columns!



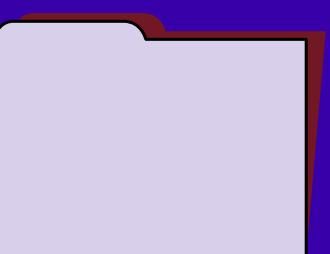
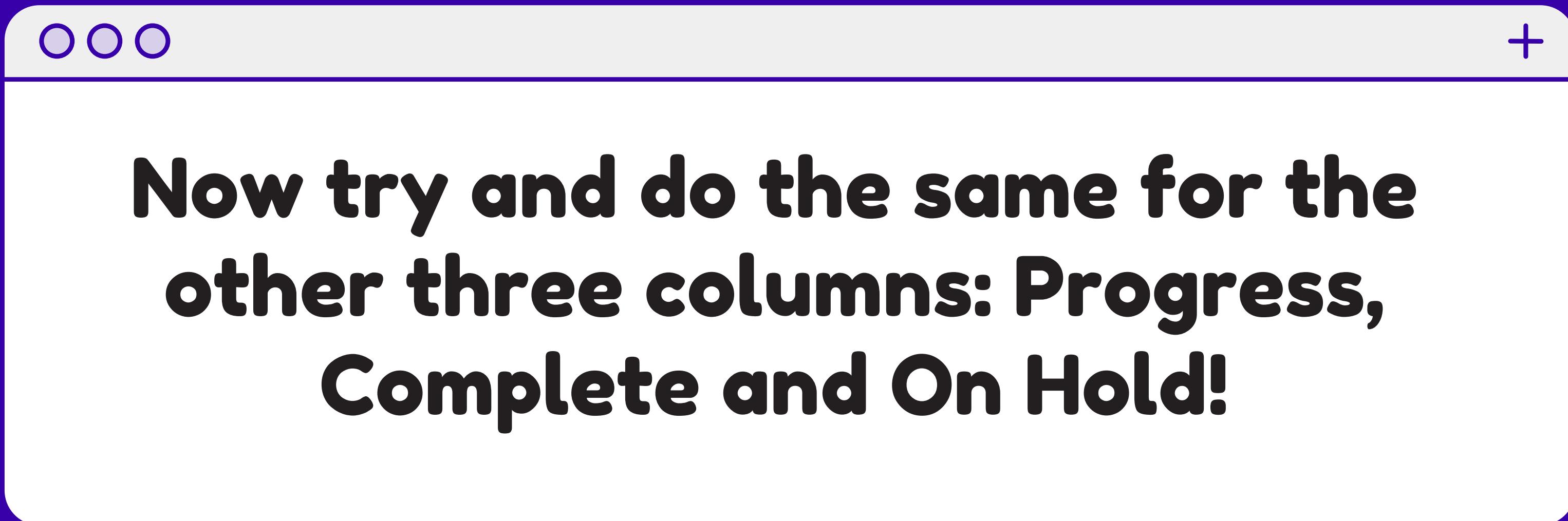
# Adding buttons

# Adding buttons



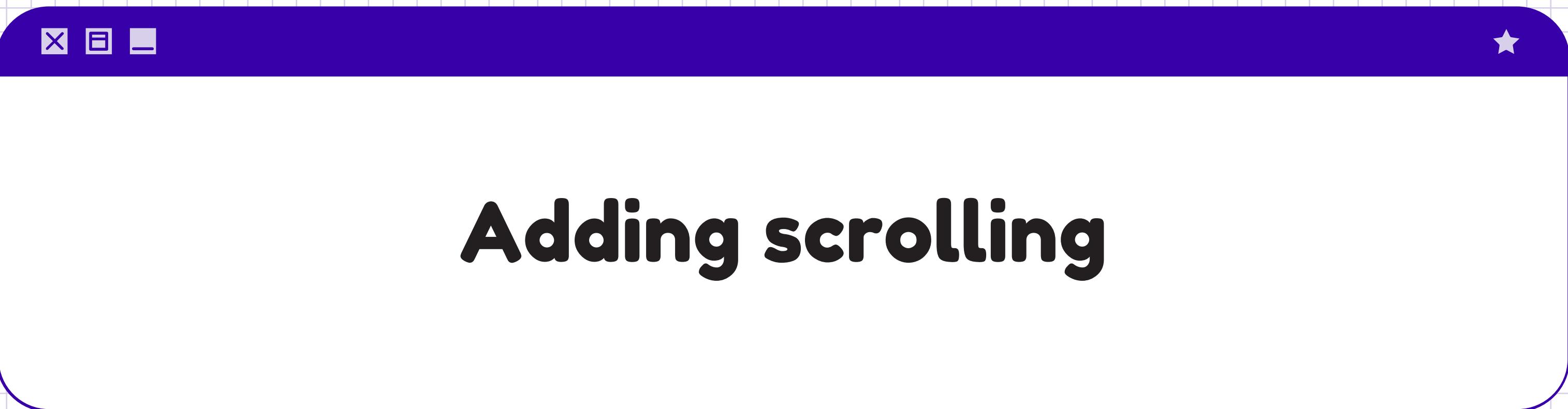
- We want to add a button group.
- Create a div class "add-btn-group"
  - Then, add another div for "add-btn"
    - Create a span with class "plus-sign"
    - Create another span with text: Add Item
    - That's it for our add item!
  - Create a new button div with class: add-btn solid.
    - Create a span with text: Save Item.
  - Create a new div class "add-container"
    - Inside this div, create a new div with class "add-item"

We hope you are having fun! ;)



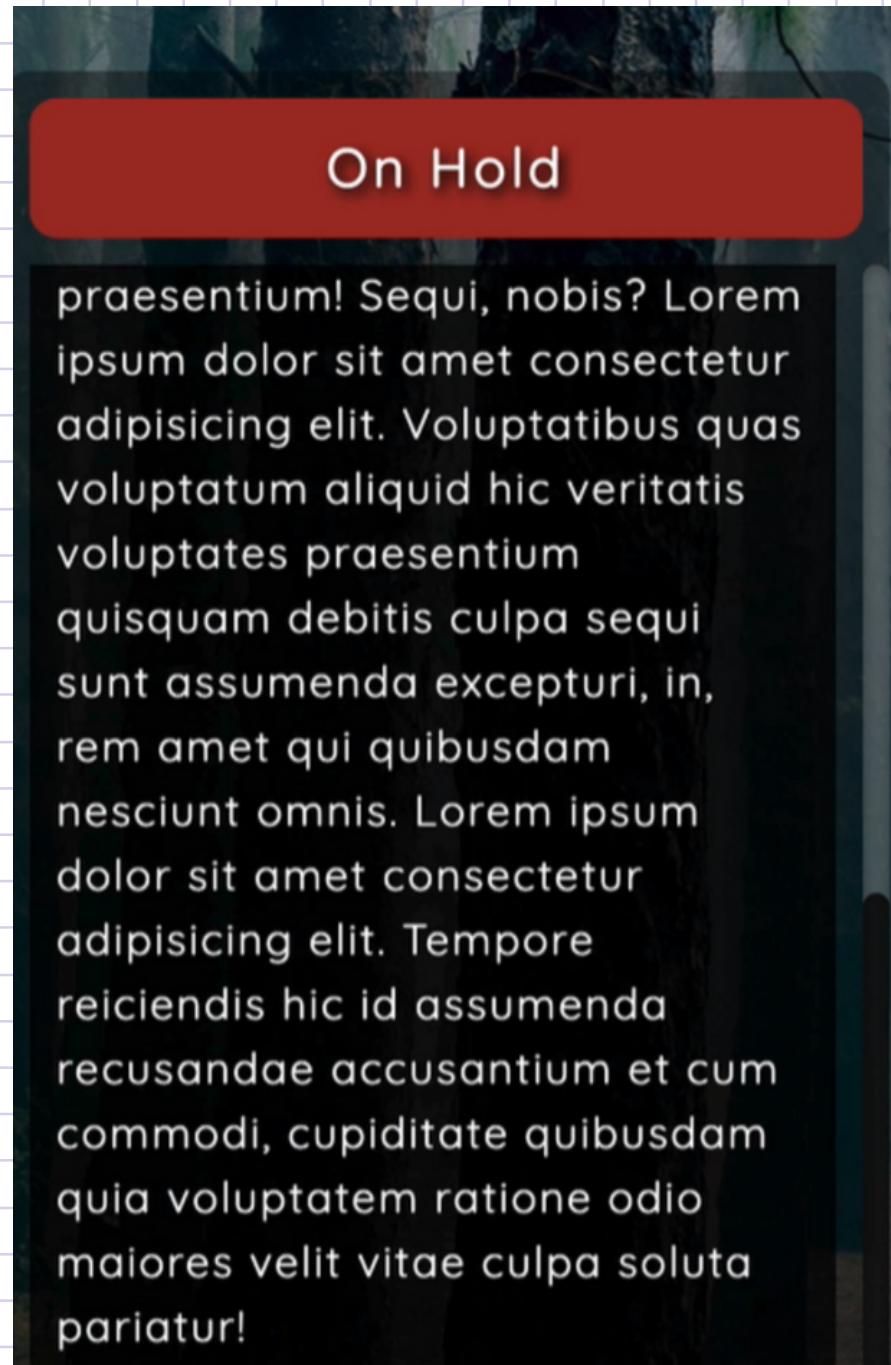
**You can copy-paste to make it a bit easier!**





# Adding scrolling

# Scrolling



- Try adding a **VERY** long text to one of your test items! What happens?
- To solve this, we want to use a custom class from our CSS file called **.custom-scroll!**
- Where do you think we need to add this? Try it out!
- Hint: We want to have scrolling **within** our columns. What **<div>** holds our unordered list of items?



**Your turn: play around with the colors in the CSS file and personalize your UI!**