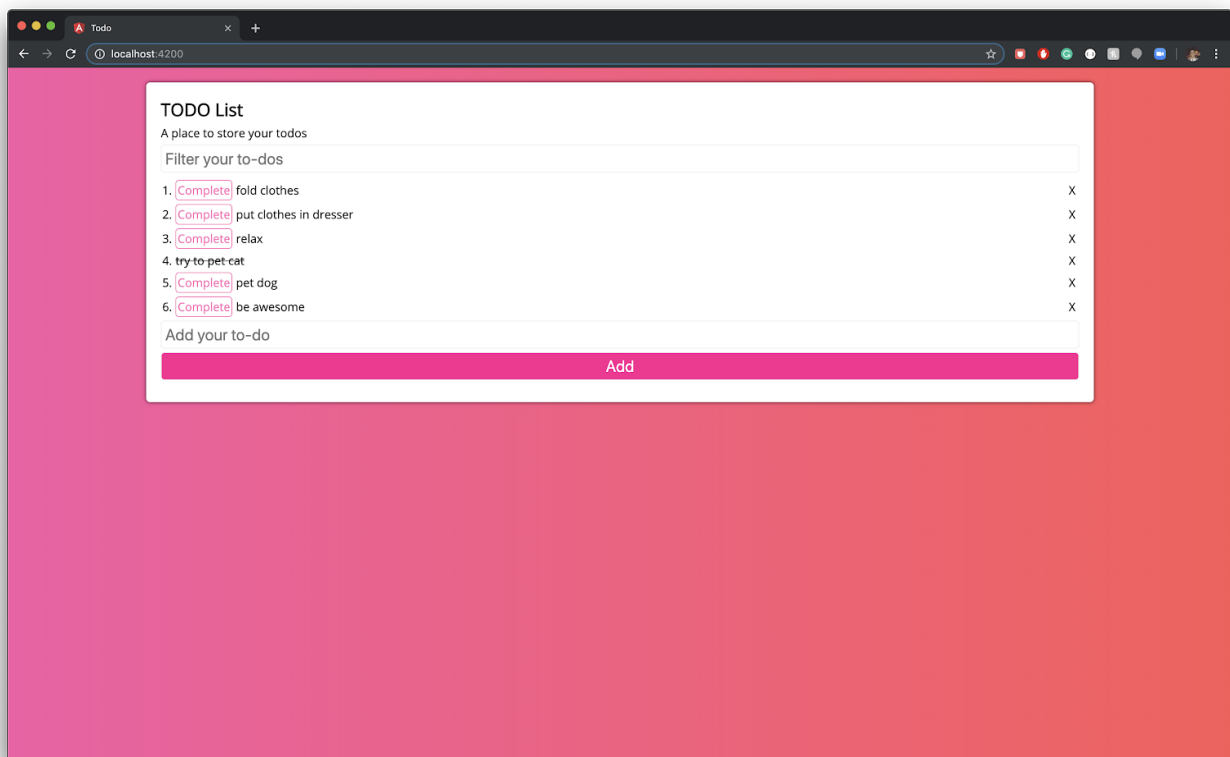# ANGULAR LAB 1 PART 1 - TODO LIST

Task: Create an Angular application that displays to-dos. The end product for this lab will not be functional, as this lab just focuses on displaying a hardcoded list of to-dos.

Build Specifications:
- Must include a component called "todo"
- Must include a Todo Interface, which consists of a "task" (string) and a "completed" (boolean).
- The component must add an array of sample todo objects to the view. Each todo object should match the Todo interface. Include a mix of complete and incomplete items.
- The ngForOf directive must be used.
- You must be able to dynamically style todo items (strikethrough when an item is completed)
- The ngIf directive must be used to remove the button "complete" button for completed todos.
- "x" buttons, "complete" buttons, the two inputs, and the "Add" button are all non-functional for now. Add them to the HTML and style them. We will make them work in the next lab.

Extended Challenge: Display a message such as "Nothing to do!" if the list of todos is empty.
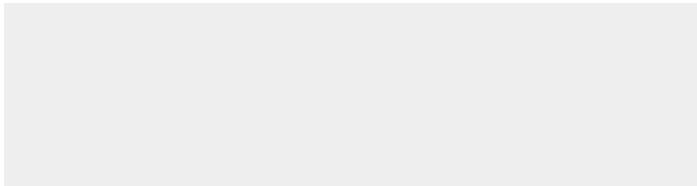
Style guide:

- Font-family: Open Sans;
- Font-weight: 300;
- Body background: linear-gradient(to right, #f857a6, #ff5858)
- Icons: Font Awesome or Material Icons
- Box-shadow: box-shadow: 0px 0px 5px 0px rgba(0,0,0,0.75);
- Colors:

#bbb

#eee

#ff0000

#ff69b4

#ff1493

ANGULAR LAB 1 PART 2 - TODO LIST

Task: Add interactivity to the todo list.

What does the application do?
1. The user can add items to the bottom of the list.
2. The user can filter the list.
3. The user can remove items from the list, by clicking the "x".
4. The user can mark items complete by clicking the "complete" button.

Build Specifications:
1. Submitting the form calls a function named addTask that adds a new todo to the array with the complete property set to `false`.
2. Clicking the "x" on an item executes a function named removeTask that removes that item from the array.
3. Clicking the "complete" button executes a function named completeTask that sets the task's `completed` property to `true`.
4. The text input at the top is for filtering. It filters as you type.

Extended Challenge: Give users the ability to edit the text of a todo.