ANGULAR TODO LIST LAB PART 1

Task: Create an Angular application that displays to-dos.

What does the application do?

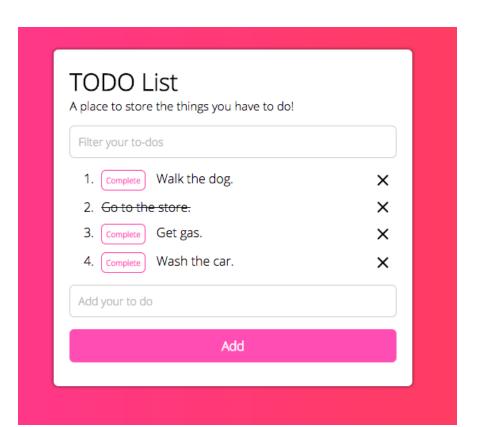
1. Displays a hard-coded list of todos.

Build Specifications:

- 1. Every Angular component must be in a separate script file.
- 2. Must include a module called "todoApp".
- 3. Must include a controller called "TodoController."
- 4. The controller must add an array of sample todo objects to the view model. Each todo object has "task" (string) and "completed" (boolean). Include a mix of complete and incomplete items.
- 5. The **ngRepeat** directive must be used.
- 6. The ngClass directive must be used to add a "completed" class for completed todos.
- 7. The **ngIf** directive must be used to remove the button "complete" button for completed todos.
- 8. "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.

Design:







ANGULAR TODO LIST LAB PART 2

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. Typing into the filter text input uses an Angular filter to filter the items shown.

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

