Angular

1. See the following picture and create the required components, templates

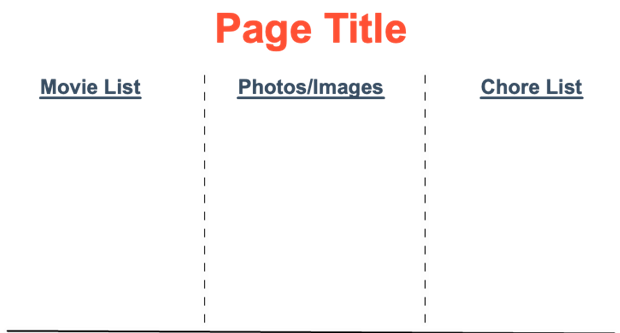


1. Create movie-list component, which should be having the movies list as shown in the above picture.
   1. You can create a movies array to hold the movies as shown below

movies = ['The Manchurian Candidate', 'Oceans 8',’The Incredibles’, ‘Hidden Figures’];

* 1. Change the movie list text by adjusting the code in movie-list.component.css to accomplish the following:
     1. The text for the heading and list items can be any color EXCEPT black.
     2. The movie list should have a centered heading.
     3. The font size should be large enough to easily read.

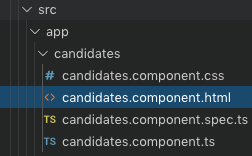
1. Create chore-list component, which should be having the chores list as shown in the above picture.
   1. Change the chore list text by adjusting the code in chores-list.component.css to accomplish the following:
   2. Use a different font, with a size large enough to easily read.
   3. The text color should be different from the movie list, but not black.
   4. The chores list should have an underlined heading.
   5. The chores in the list should be italicized
2. Create fav-photos component, which should be having a single photo as shown in the above picture. ('https://www.launchcode.org/assets/icons/target-00c7a0bc7776901e3c5cdc4c36c465f19e41a249bc7267a595f0bfa14ab944b7.png')
3. You need to get a plan to display total output as shown below



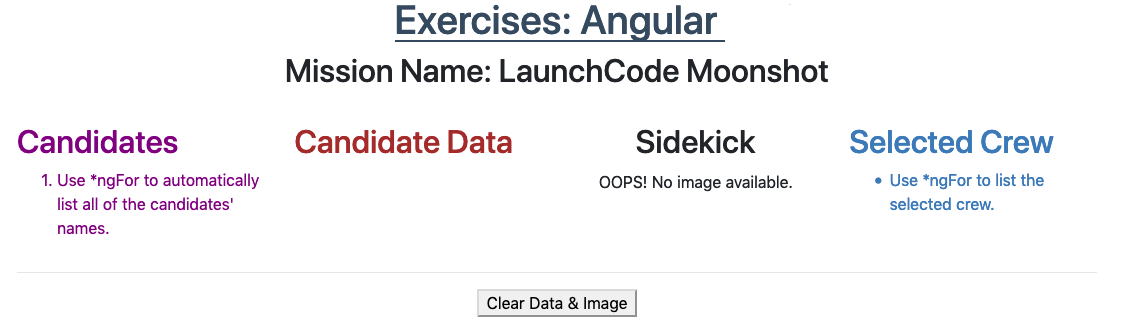
1. Finally you need to include all these components inside app.component.html to get picture shown at start of this document.

**Exercise – 2**

1. Let's build an interactive web page that allows us to review data for our astronaut candidates and select crew members for a space mission
2. Create a folder “candidates” under app folder and do the requirements. Get the help from the picture shown below



1. You have to design your application such that you need to get an html output as shown below



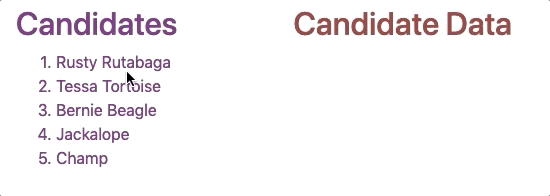
1. **Candidates Column**

Examine the candidates array in candidates.component.ts. It contains one object for each animal astronaut. We want to start by listing the names of the animals in the "Candidates" column of the web page.

Use \*ngFor in the <li> tag to loop over the candidates array and display each name in an ordered list.

We want each name to be interactive. Add a click event to the <li> tag. When a user clicks on a name, set the variable selected to be equal to the chosen candidate.

Properly done, your output should behave something like this



After mouseovering on an item from the Candidates list you should show same name details in Candidate Data column as shown below

Name: Rusty Rutabaga

Age: 5

Mass: 0.75 kg

Sidekick: Blake

1. **Sidekick Image Column**

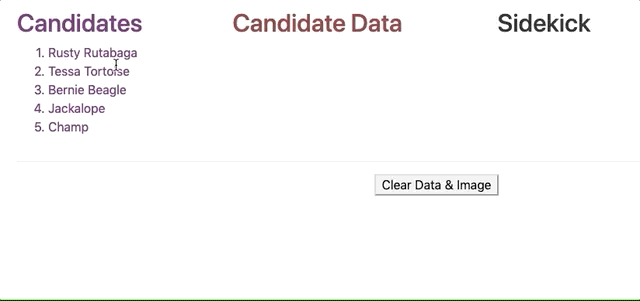
Every good hero needs a loyal sidekick, and our candidates are no exception!

When we click on a candidate's name, we want an image of their sidekick to appear under the "Sidekick" column. If no candidate is selected, we want this area to remain blank.

In the <img> tag, use \*ngIf to check if a candidate has been selected.

Replace the generic {{placeholder}} with the image property of the candidate.

Properly done, your output should behave something like this:



After clicking an item from the candidates, you need show that candidates information as shown above, and also the sidekick picture belonging to the candidates. (I have send that picture “lesson2-exercises-sidekicks.gif” to you, Please refer it.

1. **Selected Crew Column**

Once we select a candidate, we want an option to add them to the crew of the next space mission.

In candidates.component.ts, code an addToCrew function that takes an object as a parameter.

If the candidate is NOT part of the crew, the function should push them into the crew array. Candidates who are already part of the crew should be ignored.

In candidates.component.html, add a "Send on Mission" button next to the "Clear Data & Image" button.

Add a click event to the button to call the addToCrew function. When clicked, pass the selected candidate as the argument.

Under the "Selected Crew" section, use \*ngFor to loop over the crew array and display each name.

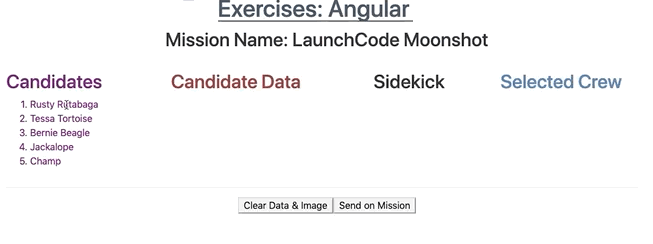
1. **Clear Crew List**

Add a "Clear Crew List" button under the "Selected Crew" list.

This button should only appear when the crew array contains data. Use \*ngIf to make this happen.

Add a click event that clears the crew array.

Properly done, your output should behave something like this:



(See the final output in the picture, “lesson2-exercises-crew.gif”)