FIELDGO PRE-INTERVIEW CODING EXERCISE – "SHAPES"

Overview

The objective of the exercise is to demonstrate basic coding proficiency in front-end development. It is intended to be completed off-line and should take 1 to 2 hours. **Please document how long you spend and include in your final submission.**

The exercise involves implementing a simple single page web app in javascript/typescript. Implementations in a recent version of Angular are preferred, but if necessary, you may use any appropriate javascript framework if you are unfamiliar with Angular.

When reviewing your submission, we will be looking for:

- Correct results
- Well-structured, easily read and easily maintained code
- Robustness to edge-cases and failures
- Good UI design skills
- Good performance characteristics

When implementing your solution, you can make reasonable assumptions about interpretation and edge cases. **Please document any assumptions that you make.**

The Exercise

Implement a web app to parse a JSON documents from a REST API endpoint and display the results.

The JSON document will contain an array of objects, each representing a physical item. They will have the following properties:

- name: string
- height: number (0-100)
- length: number (0-100)
- width: number (0-100)
- mass: number (0-100)

An example API endpoint with some test data can be found at:
http://101.0.97.194:5443/shapes/example.json

The results should be displayed in the following form:

- A panel on the left containing a table of the items, with the following columns:
    - Name
    - Shape
    - Density (mass divided by volume)
- A "details" panel on the right to show all the details of a selected item.

"Shape" is a text description, which should be derived as follows, in order of precedence:

- If the height is both greater than double the width and greater than double the length, "Tall".
- If either of the width or length is greater than double both of the other two dimensions, "Long".
- If the height is less than half of both of the other two dimensions, "Flat".
- If the width or length is less than half of both of the other two dimensions, "Thin".
- Else "Other".

The "details' panel should be populated when an item in the table is clicked, and should display the selected item's name, dimensions, weight, shape, and density.

"Bonus" Features

Optionally, you can add bonus features to your web app, such as making the table sortable, or a filter box for the table. If you do so, please document what features you have added, and roughly how long you spent on them.

Submitting Your Solution

Please submit your solution as a zip or tarball containing:

- Source code
- Any build output needed to run the web app
- A README containing any implementation notes or instructions for running the web app

Please email your final solution along with the time spent to kayla.crook@fieldgo.com.