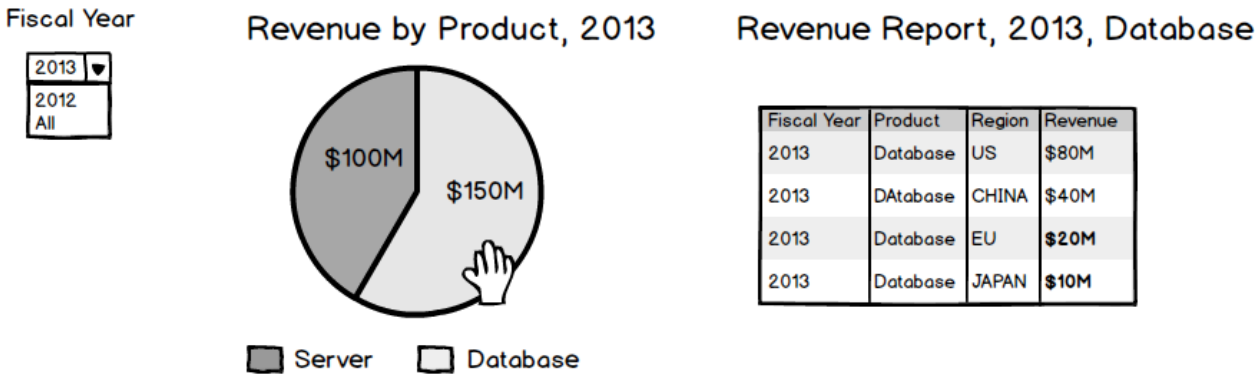
# Glint UI Coding Challenge

These are the Glint coding challenges. We hope that you will find them interesting, and maybe even fun! You are free to choose whichever one you like.

## Option 1) Simple analytics web app.

The app has a single screen with dynamic content (see attached "mockup"). The screen consists of the following:

1. A drop down that allows the user to filter the data by fiscal year. The drop down is always present.
2. A chart that breaks down the revenue data by product.  The pie chart is always present. The user can click on the chart to select a specific product at any time (see below). (You can use any chart type you feel appropriate)
3. A data grid that displays the detailed revenue data for a specific product. The grid becomes visible only after the user chooses a product. Show the revenue figures in bold if they are below average (as indicated in the mockups). Also, use alternating background colors for the rows.




## Requirements

- Implement this feature using a main stream client-side MVC framework. Describe the rationale of you choice.

- Fetch the revenue data by creating a Node.js endpoint using the URL below. Demonstrate (with code) the use of the network/synchronization API with proper error handling.
    - http://aware-ui-test1.s3.amazonaws.com/sample_data.json.txt.gz
- The implementation should work for all major browser platforms (specifically Chrome, Safari, Firefox and IE 8+).

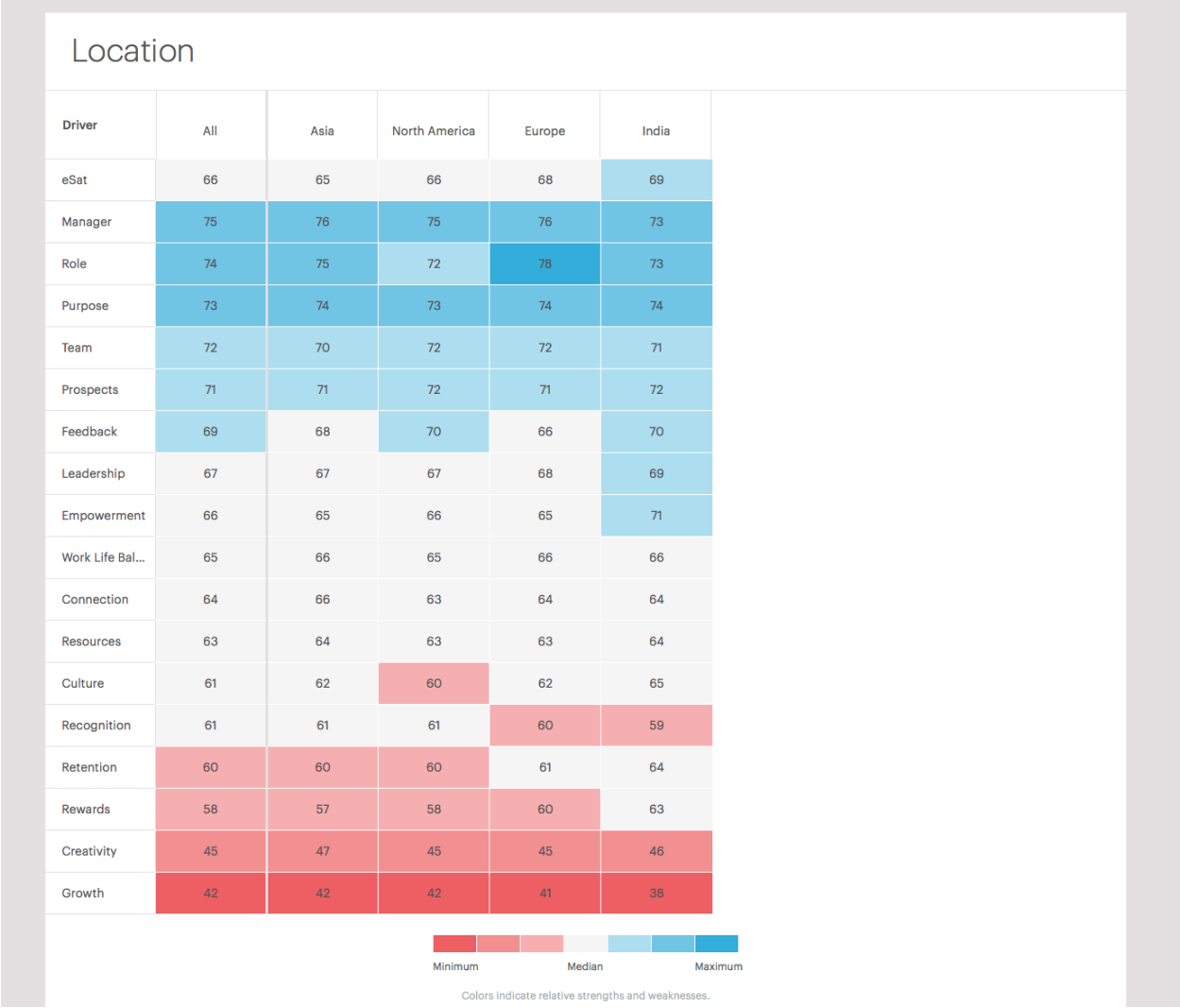
## Nice-to-have (optional)

- Demonstrate/describe your approach to responsive design by making sure the page renders properly in both a desktop browser and a mobile browser.
- Allow the user to edit the grid (e.g., change the revenue figures). The charts should refresh automatically as the data is committed (by pressing enter, e.g.).
- Animate chart content changes (e.g., after choosing a different fiscal year filter)
- Demonstrate how to automate unit testing of the app.

## Option 2) Sample Heat Map

Your task is to create a heat map (a graphical representation of data where the individual values contained in a matrix are represented as colors) of temperature and location for a particular data set.

This is a mockup of what a heat map should look like.




The dataset you have to work with is located at:

http://aware-ui-test1.s3.amazonaws.com/ui-test2/files/heatmap_data_not_sorted.csv

You will be responsible for creating a web page that accomplishes the following tasks:

1. Retrieve the data from the server (e.g. using an ajax http call) and manipulates it to create a heat map.
2. Put the temperature in 5 buckets from min to max value (e.g., 47-55, 56-71, ... etc), find the largest connected region of the same bucket and mark their cells (e.g., with "x").
3. Draw borders around the largest connection region.
4. (optional) Add a UI control to increase/decrease the number of buckets
5. (optional) Assuming you can order the rows and/or columns however you want, how would you determine the order to increase the size of largest connected region?

## General guidelines

- Write clean, properly abstracted code and pay attention to data structure and performance.
- If you think requirement is ambiguous, fill in the blanks and document your assumptions.
- Describe possible optimizations that are not done as part of the exercise.