JS Evaluation Application:

Description:

This document outlines the features, technical specs, and technical debts of the JS Evaluation for

360 cloud solution.

Features:

The UI for the small application consists of the following:

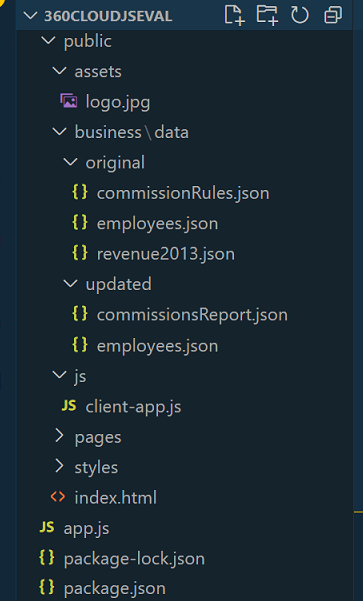
**UI:**

1. Home Page – (uses only as placeholder)
2. Processing page– (is the page where operations can be take place) ref images below
3. Data menu – (display the raw json data) and it contains
   1. Original page – (used to display the original unmodified JSON files)
   2. Update page - (this is only a place holder)

**Technical Design:**

The application was built using the following technologies:

1. Node as the server side and version (14.18.1)
2. Express, lodash and nodemon were added only for developing
3. VSCode is the IDE
4. File structure



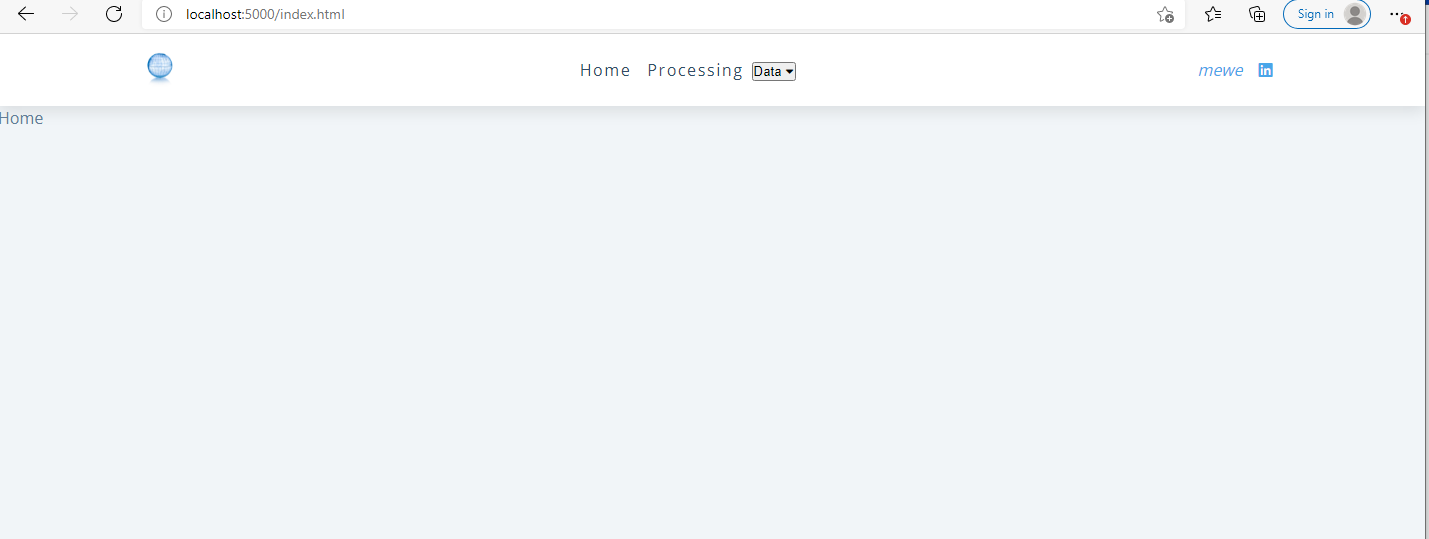
1. All the logic reside on the app.js file for the server and handles the following
   1. Routing
   2. CRUD for the database (commissionRules.json, employees.json and revenue2013.json)
2. The public/js/client-app.js
   1. Handles all the logic to execute (CRUD functionality) consuming from the server

USAGE

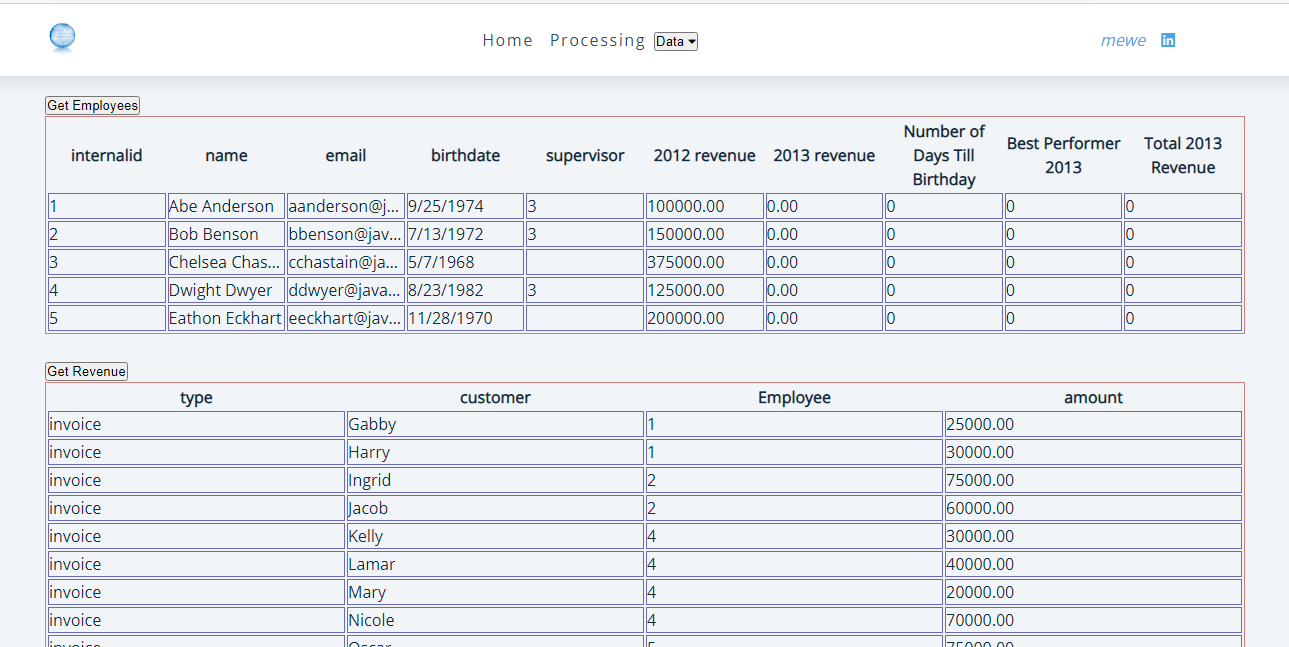
1. When launching the app on the node root by entering
   1. >npm start
2. Than navigating to the browser and entering <http://localhost:5000>

You will be presented with the following screen, which is the home page

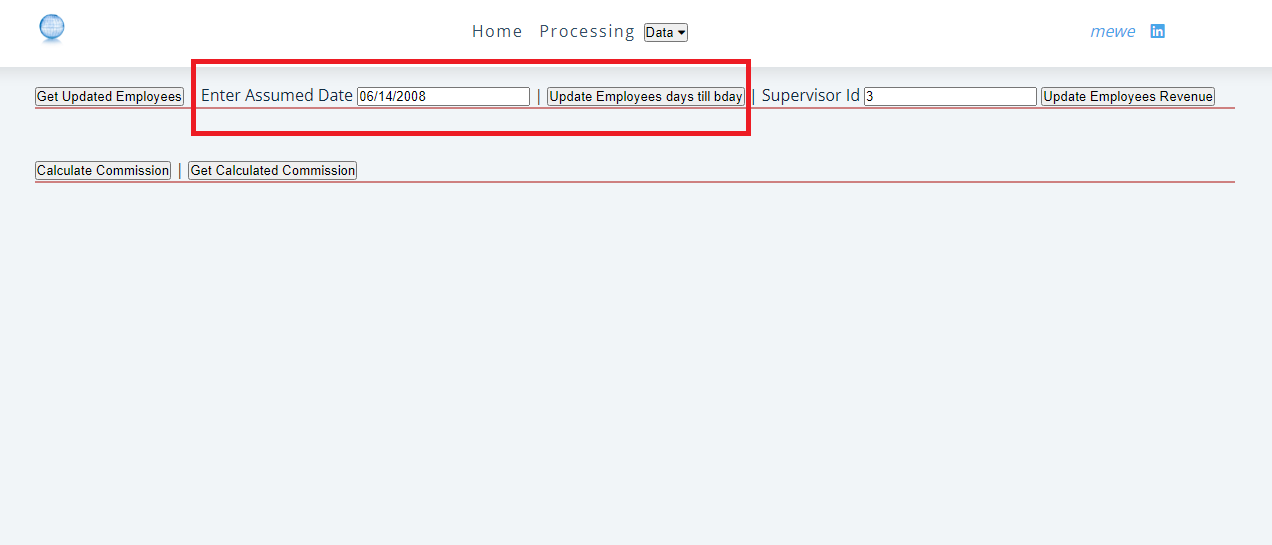
Home Screen



::Data->Original (will display the original json files)

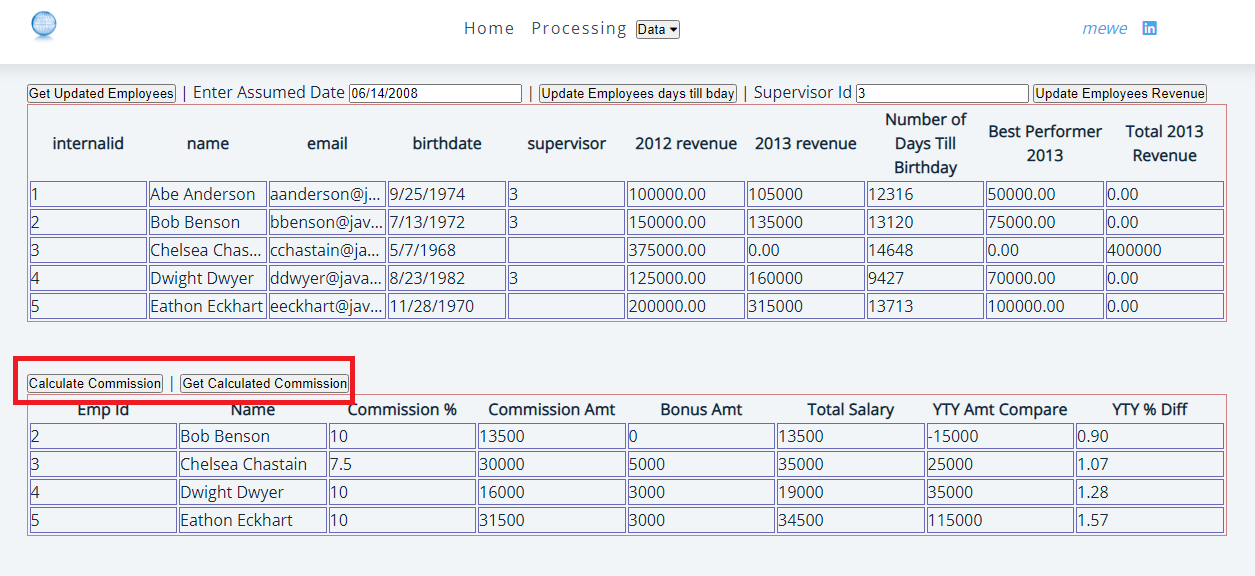


::Processing (from the top Nav) - you will be presented with the following screen



1. Initially you click on the [Update Employees days till bday] , this will add the [number of days till birthday] to the employees.json file, here you can also change the [target assumed date]
2. Now you can view the results, by clicking on the [Get Updated Employees]
3. Click on the [supervisor Id] enter the supervisor in this case there is only one, then click on [Update Employees Revenue]
   1. [Best Performer 2013] and [Total 2013 Revenue] Fields will be created and calculated
   2. Again to view results click on [get Updated Employees]
4. Click on [Calculate Commission] button (will calculate the commissions and bonuses)
   1. This creates a new data file [CommissionReports.json] (which display the commissions)
   2. Click on the [Get Calculated Commission] to view results

Reference the below screenshot



Technical Debt:

There are many thing that could be improved on this JS Eval program

1. Create different modules to separate concerns (it was like that originally, but all of it sudden node, started actin on me to the point that, it would not return data
2. An implementation of DRY approach need to be implements
3. Better ui design
4. Loading the display automatically after an action is executed (ie after calculating commission, you should not have to click on the [Get Calculated Commission]

Remarks

I added the [Year to Year] Amount compare and [Year to Year Percent] Comparison

Installation:

1. Download and install node
2. NPM command
   1. >NPM Install (will install the dependencies) i.e node\_modules