

Links needed:

<https://waaiez.software/apis/dsw-backend/api/data/tenants> (<https://waaiez.software/apis/dsw-backend/api/data/tenants>)

<https://stackoverflow.com/questions/21987909/how-to-get-the-difference-between-two-arrays-of-objects-in-javascript> (<https://stackoverflow.com/questions/21987909/how-to-get-the-difference-between-two-arrays-of-objects-in-javascript>)

<https://www.geeksforgeeks.org/mongoose-save-function/> (<https://www.geeksforgeeks.org/mongoose-save-function/>)

So you need need to create 2 routes which will be used by the app

These 2 routes will go in the `data.routes.js` file

- `/api/data/new-shops`
 - This is a GET route
 - This will show new shops by comparing our current tenant list and the list that was uploaded before
 - This doesn't need to be a full implementation, I will help you implement when the time comes
 - Your task for this route is to **create a new project** and **experiment with comparing 2 arrays of objects**
 - This is because the tenants list we upload gets turned into json and pushed into an array
 - For this you need to check the format of the data we return, use the link I gave above, it outputs an array of objects and each object has a `shopName` field.
 - I have attached a stackoverflow link that will help
 - Reminder that this isnt required for the mvp so take your time
- `/api/data/report`
 - This is a POST route
 - You create this in `data.routes.js`
 - The logic will be handled in `data.controller.js`
 - This will be used to report any wrong data
 - The frontend will have a form like interface where the user will enter any required data and you need to access the data on the server
 - The body will include details of the report
 - Fields will include:
 - `type`
 - `title`
 - `description`
 - `reportDate`
 - You can look at the `auth.controller.js` file and the `signUp` method in that file to see how you pull in fields from the body
 - On the second line which says `const { tenant: Tenant } = db;`
 - You need to change this to `const { tenant: Tenant, report: Report } = db;`
 - This is the Report model where the data gets saved
 - See the geeksforgeeks link on how to save to the database
 - Again this isnt needed for the mvp, just take your time with it, ask questions and i'll try help as much as i can

PrivateBin is a minimalist, open source online pastebin where the server has zero knowledge of pasted data. Data is encrypted/decrypted *in the browser* using 256 bits AES. More information on the project page (<https://privatebin.info/>).