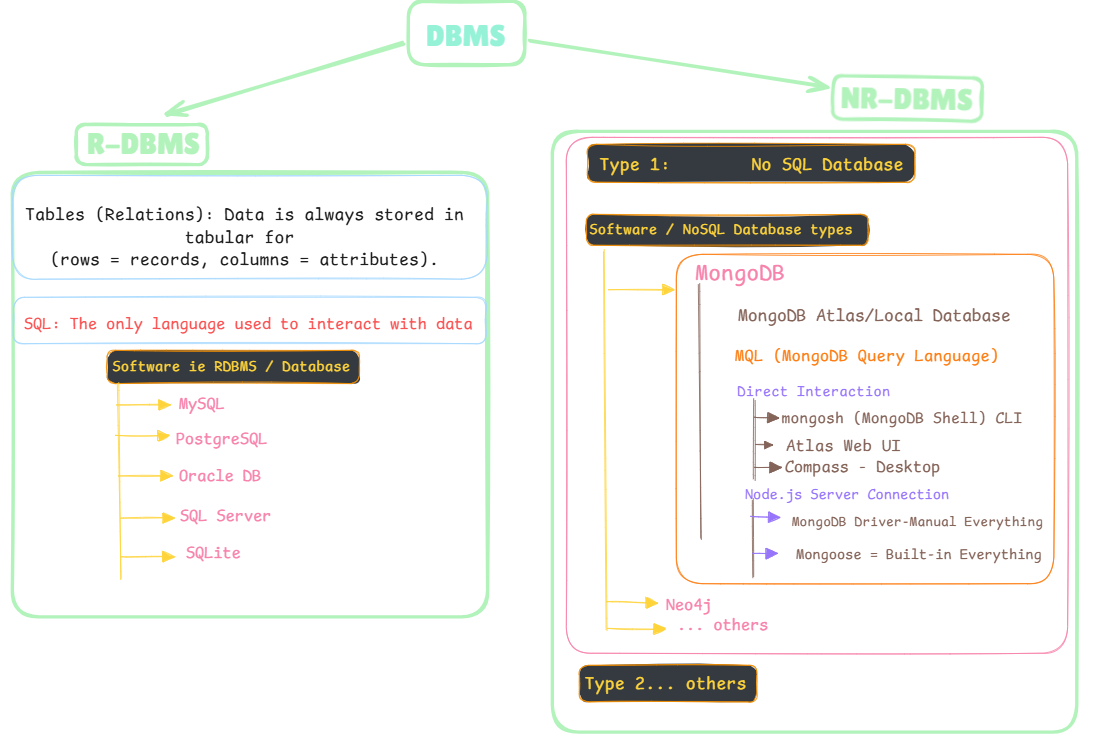
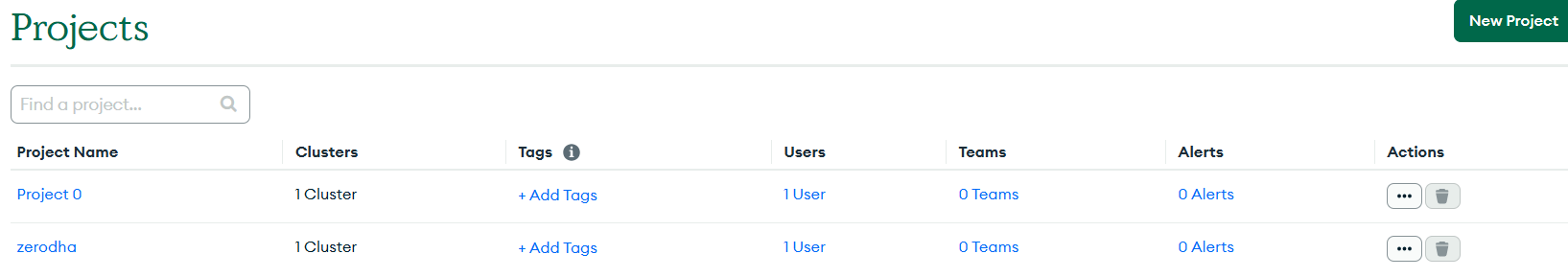
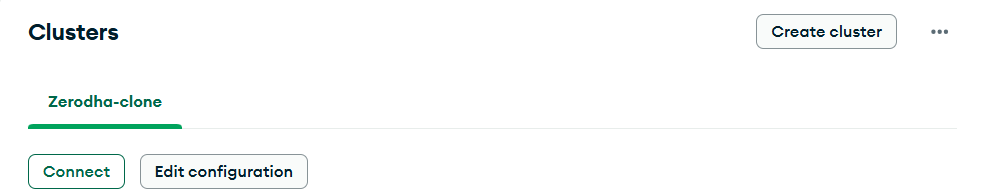
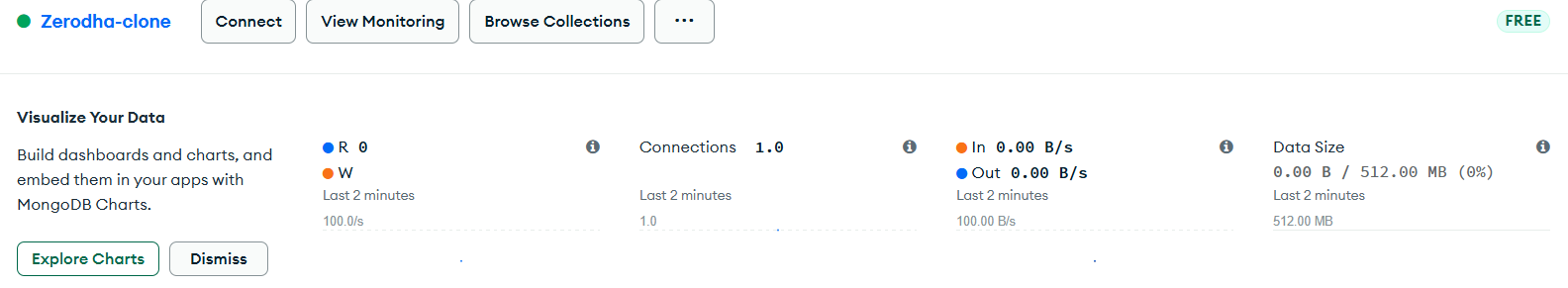
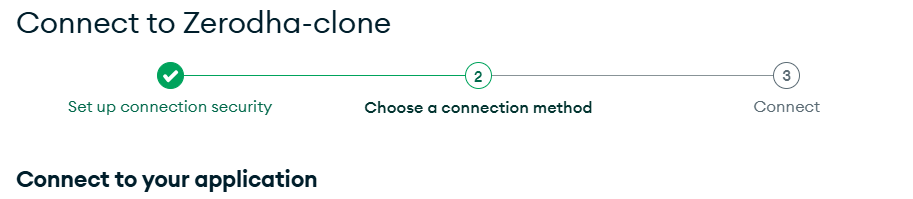
**ZERODHA BACKEND**

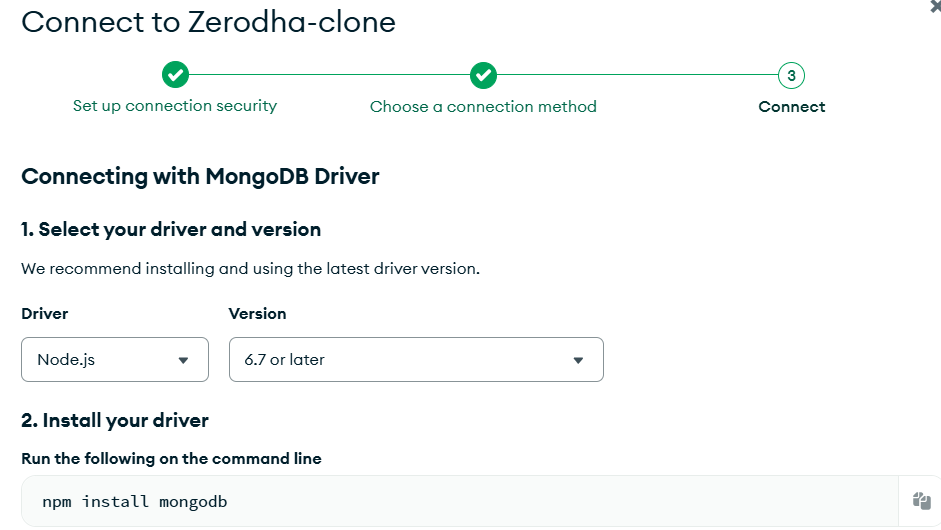
****

****

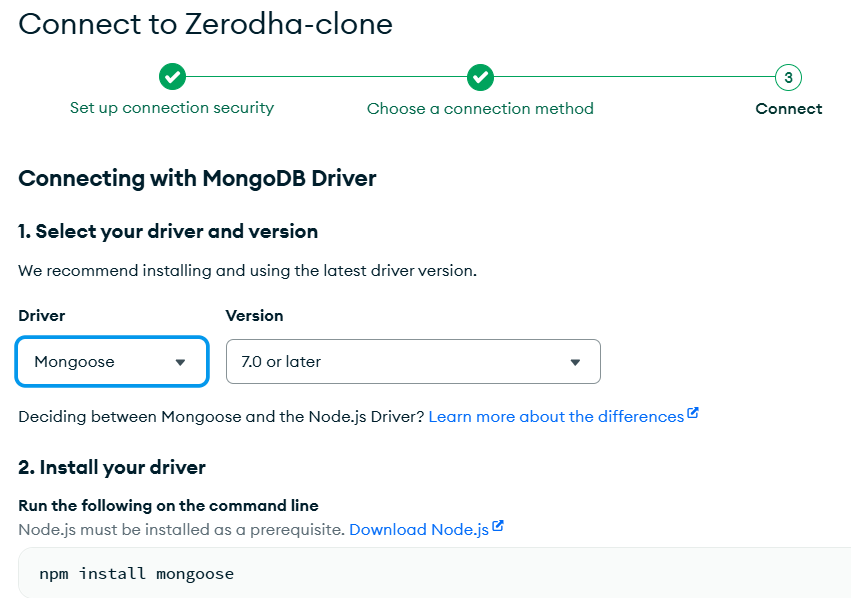
****

****

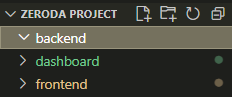
****

****

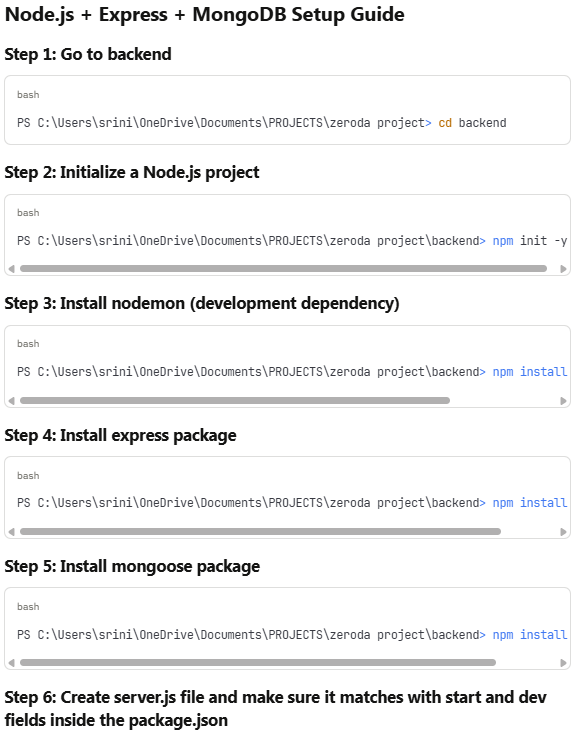
Going with Driver:Node js all manual work   
so go with mongoose driver.

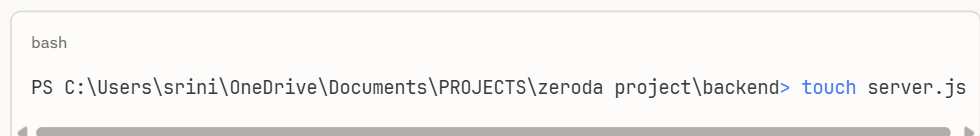




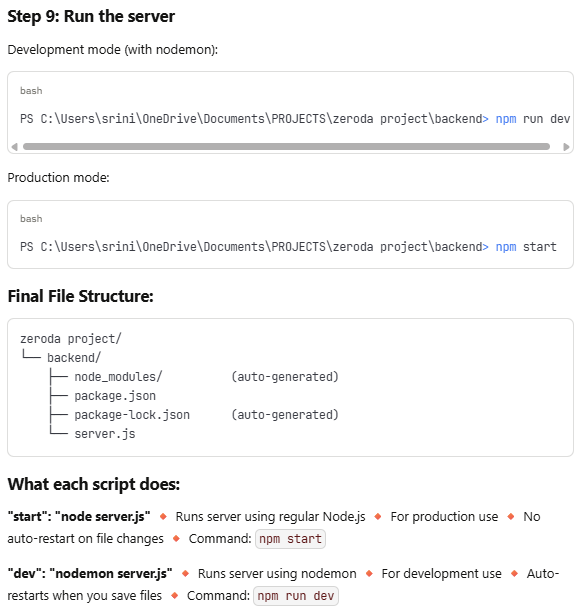


Refer frontend and backend set up notes

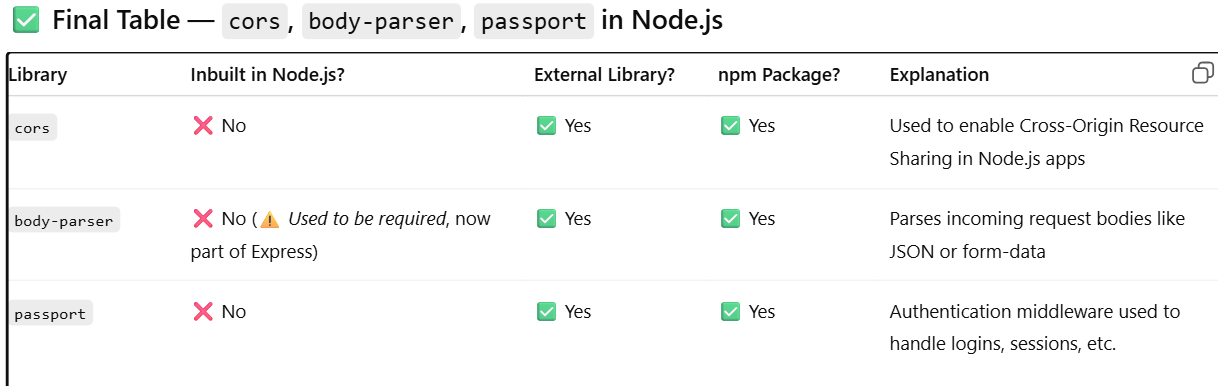




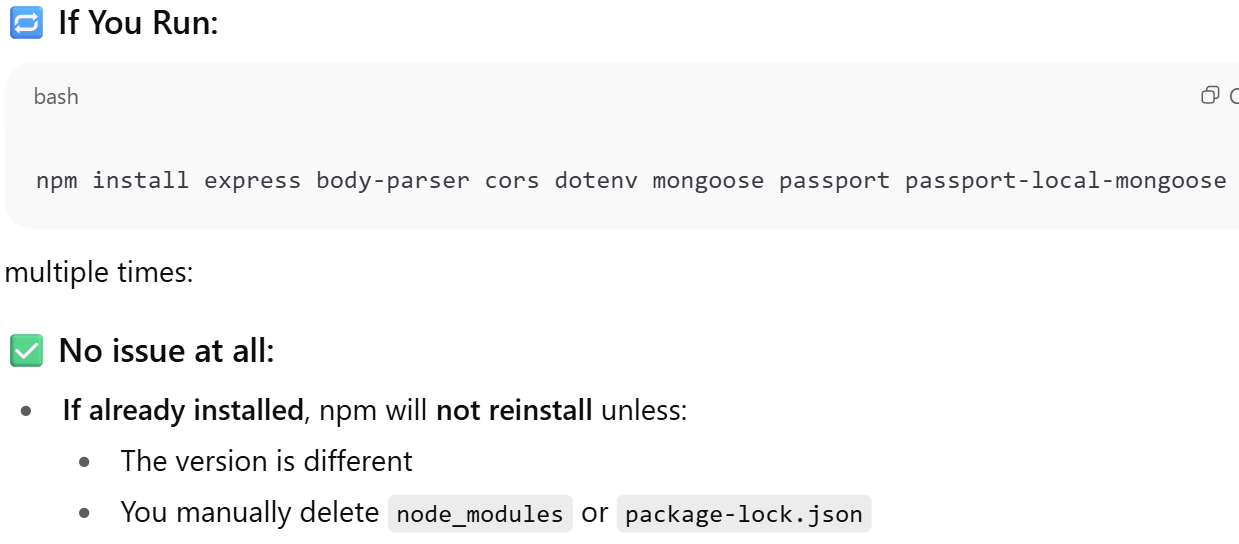




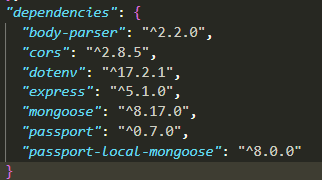
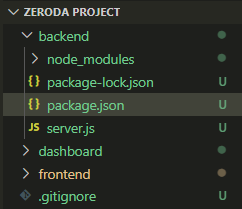
Now:



Please do refer CORS file …

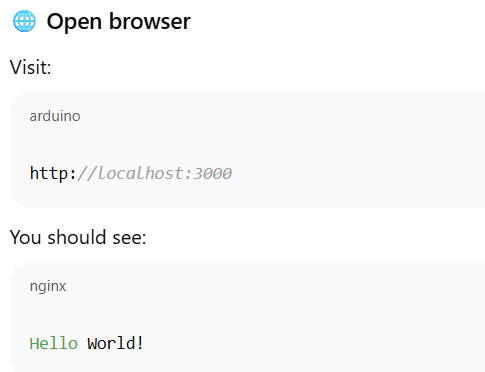


Note:u can use i or install for above

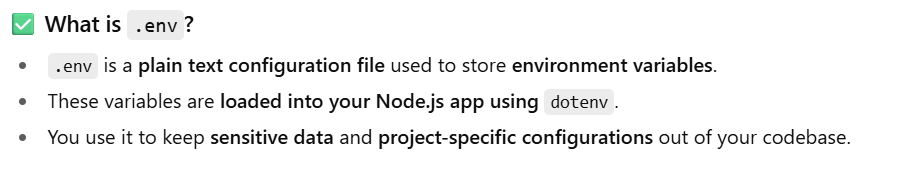


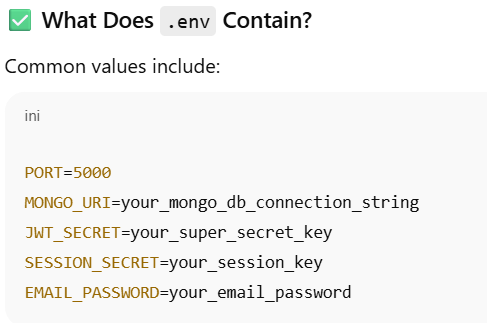
Basic setup for checking if your Express server is running.

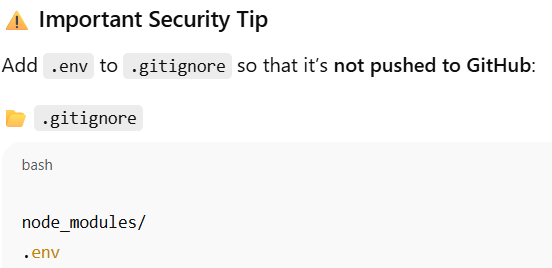




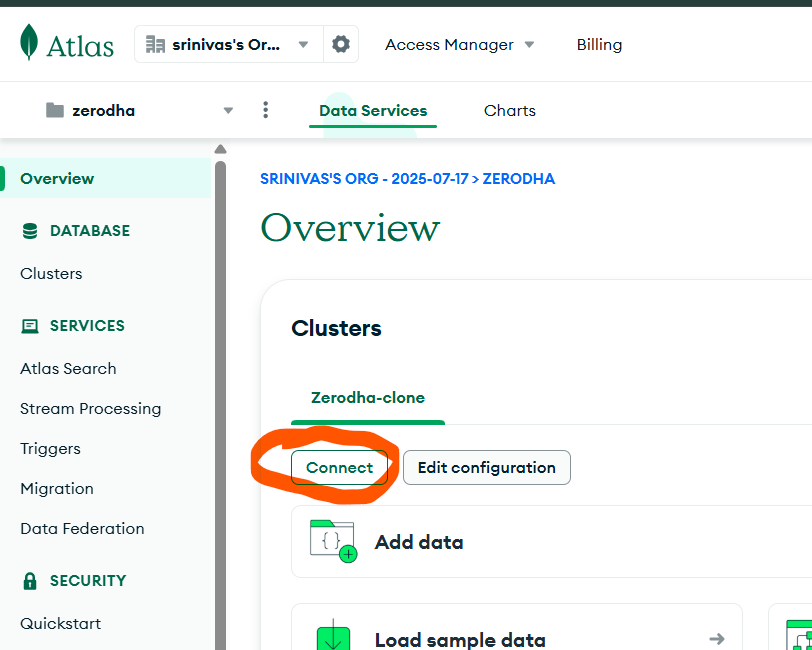
Now u can gradually add other middlewares and routes hander

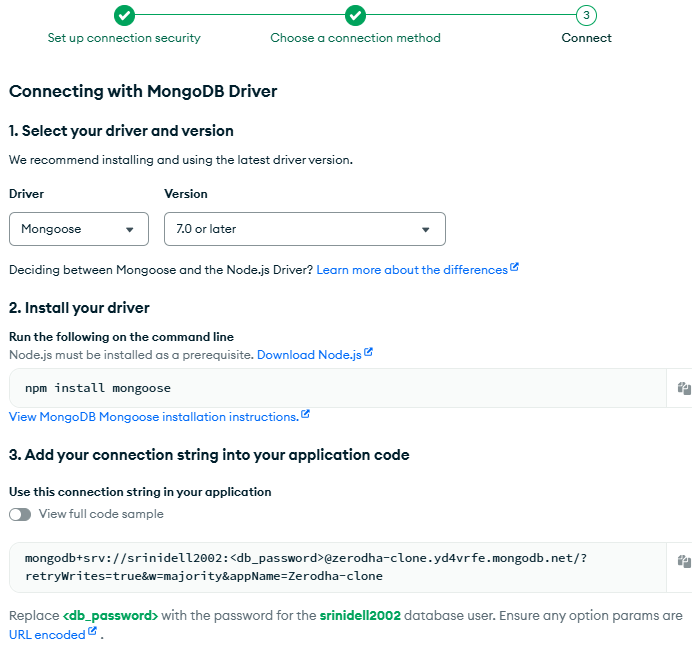




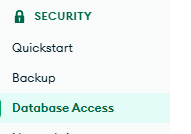


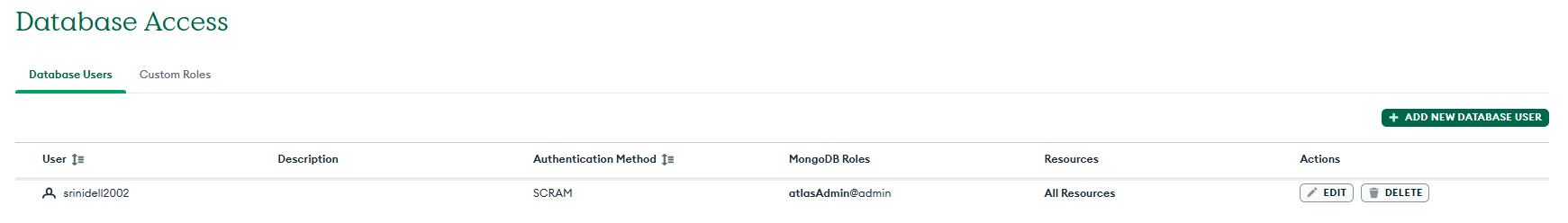
Only Mongoose for Now

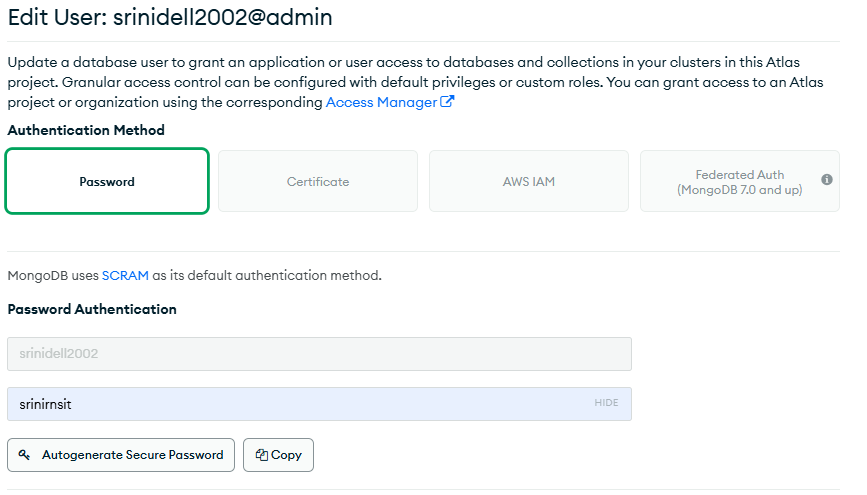


make sure driver is mongoose if u choose nodejs then u have implent manulaly all

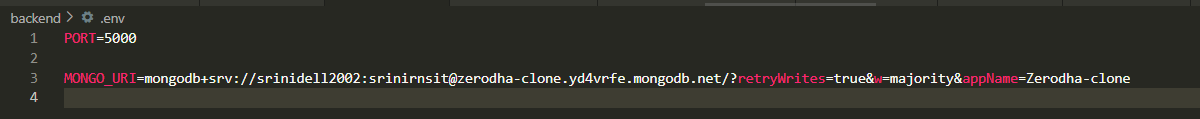
To access password :



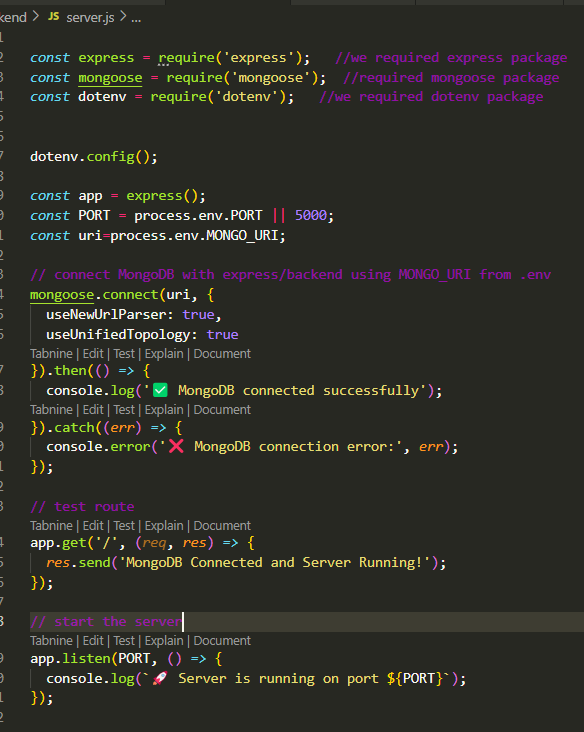


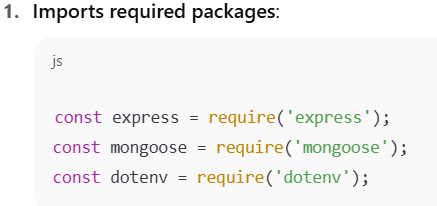


Update ur [ .env] file :

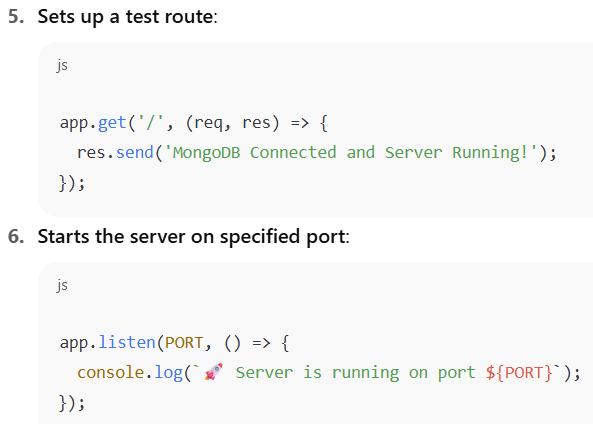


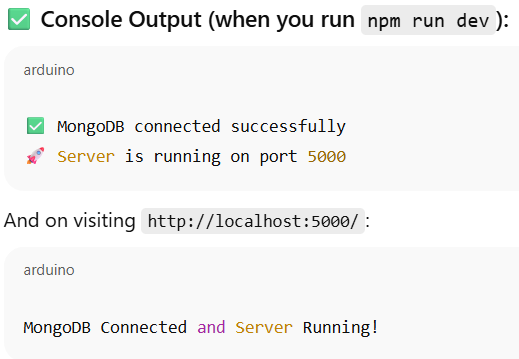
Update the server.js

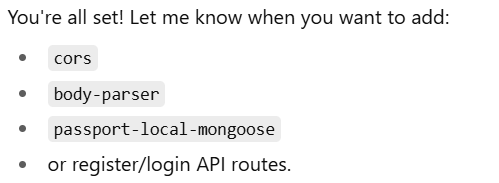




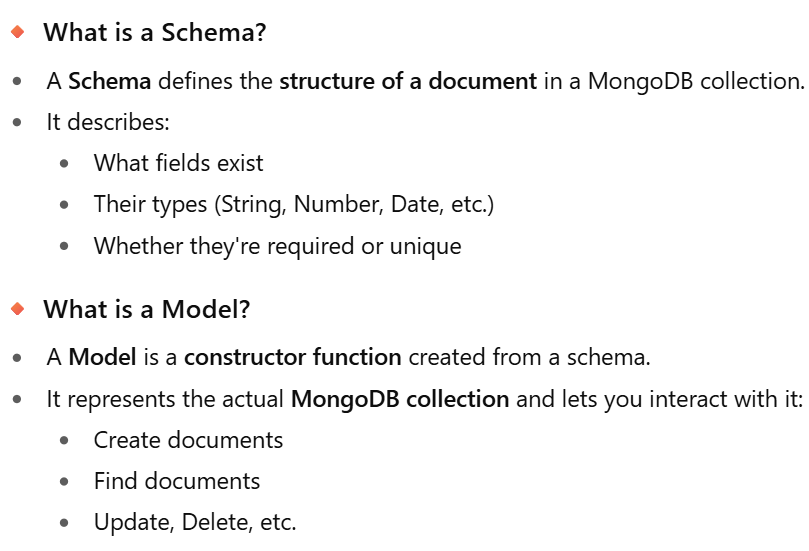


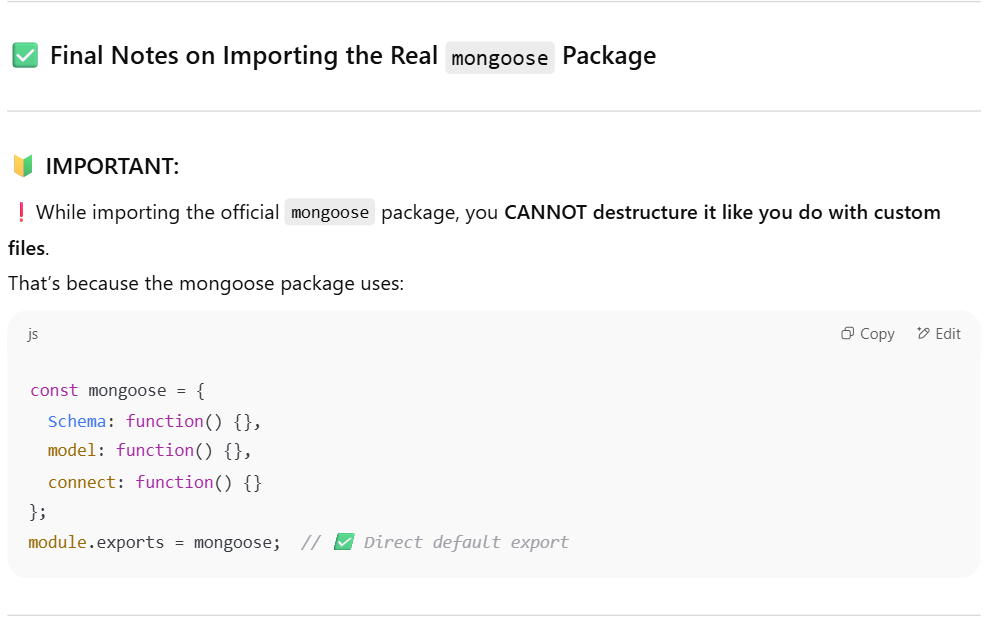


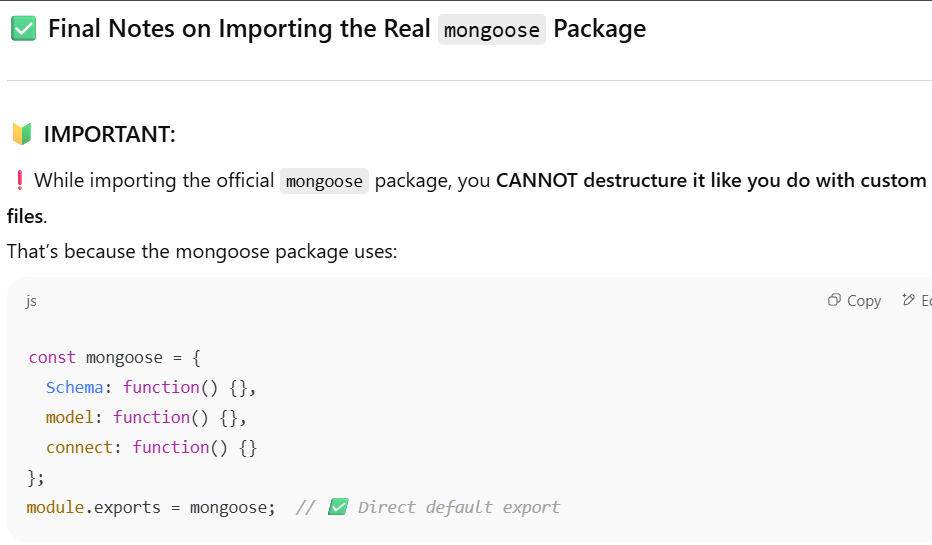




Creating schemas and models :

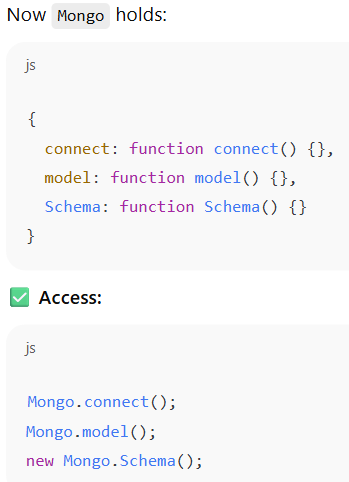






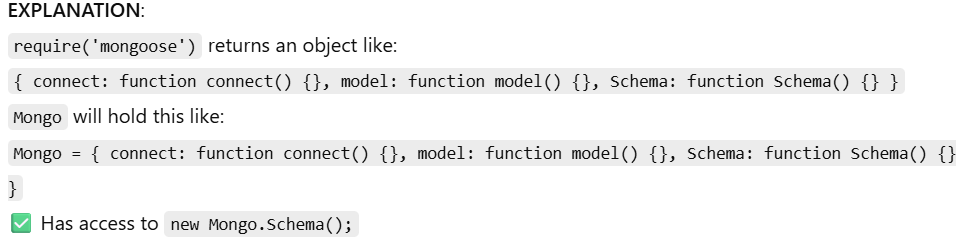


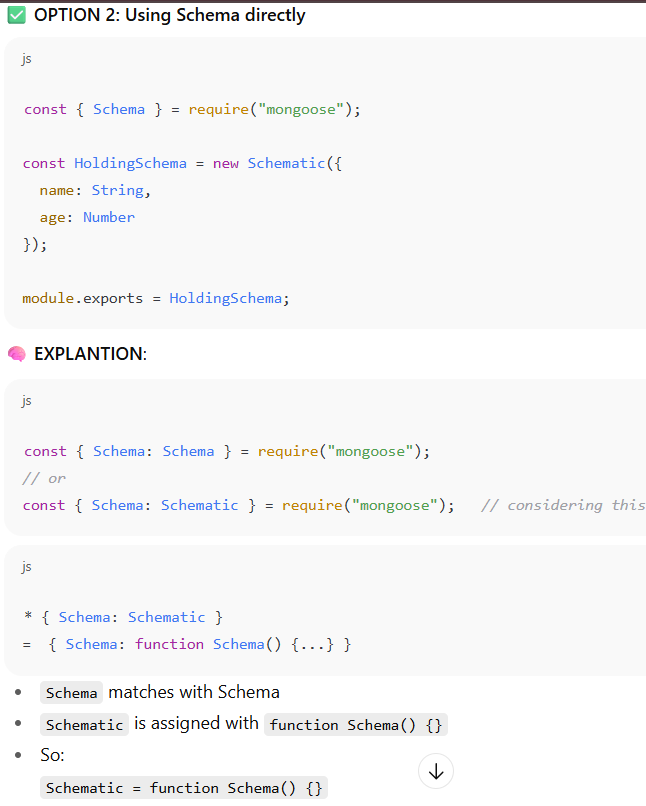




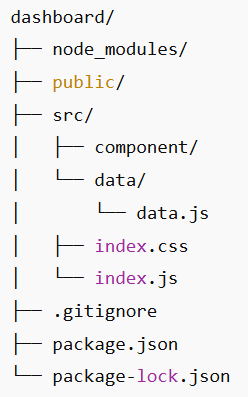
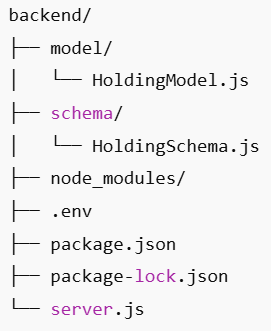


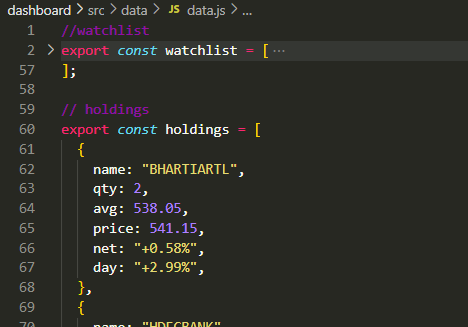


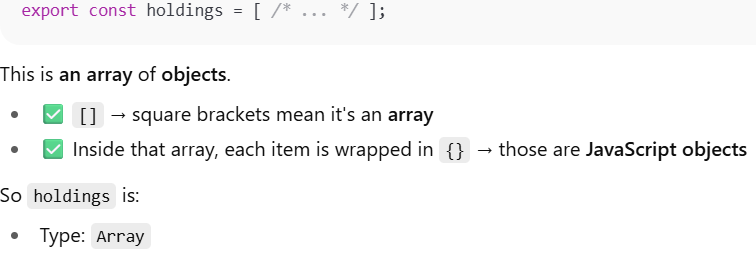


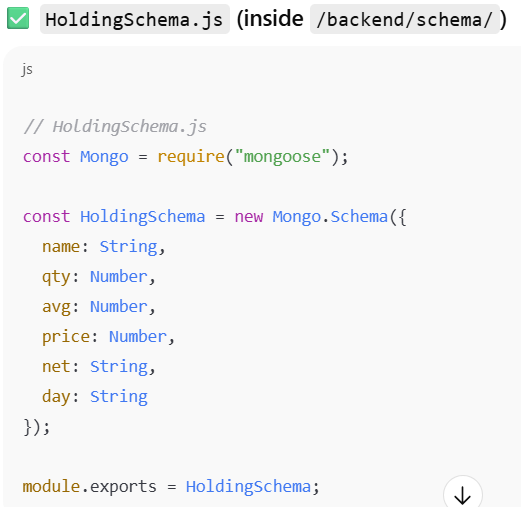








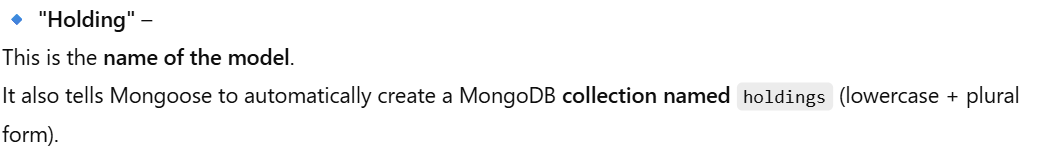
****

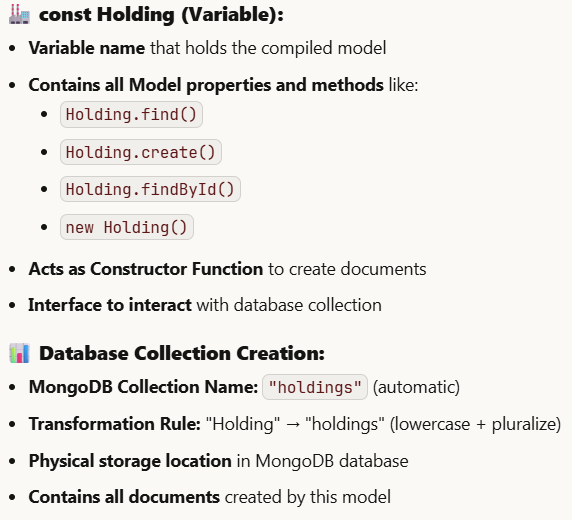
****

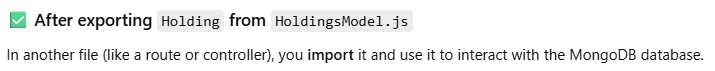
**Schema is based on data.js**

****

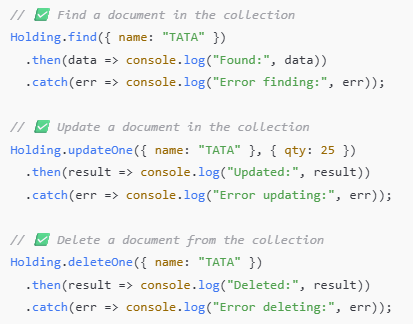
**Model is based on Schema**

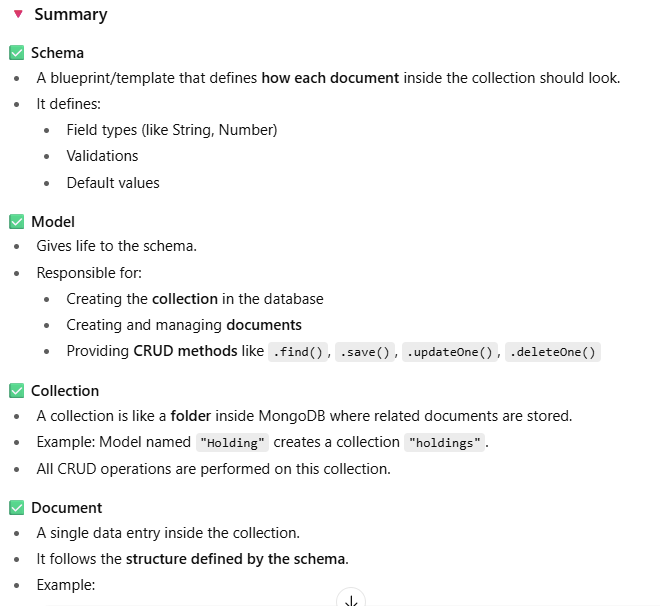
****

****

****

****

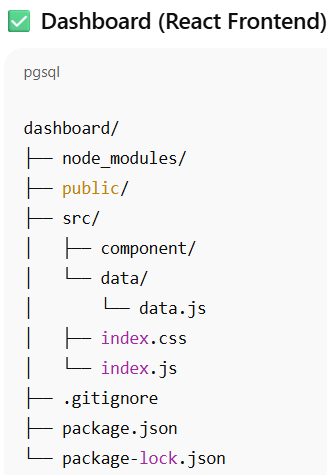
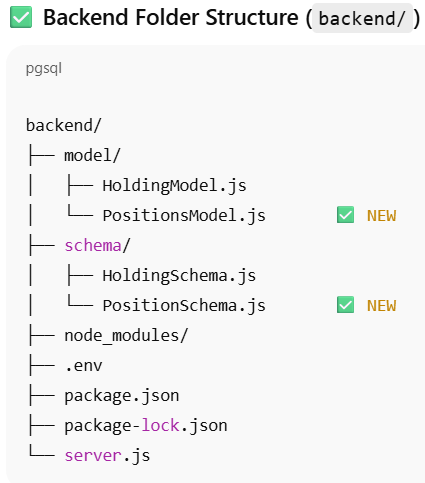
****

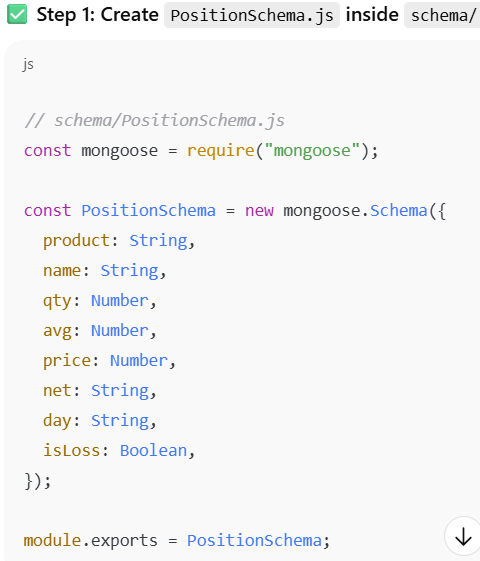
****

****

**Similarly for positions array inside the data.js**

similalry create create schema based on the positions array inside the data/data.js and besed on schema create a model.

****

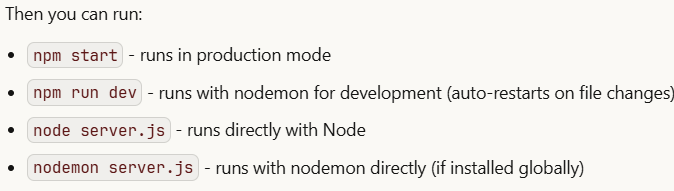
****

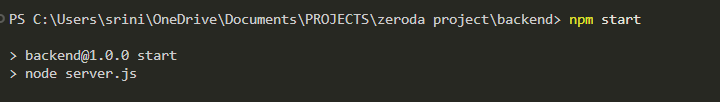
****

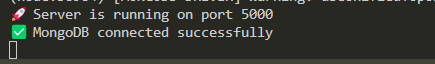
Similarly for orders but this is not from data.js

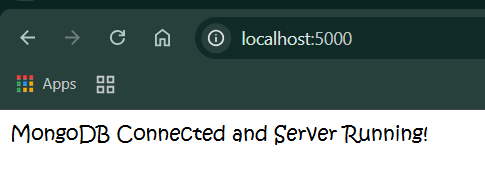
****

Lets check whether server and mongoDb is working or not .

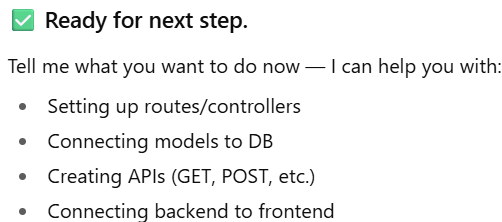




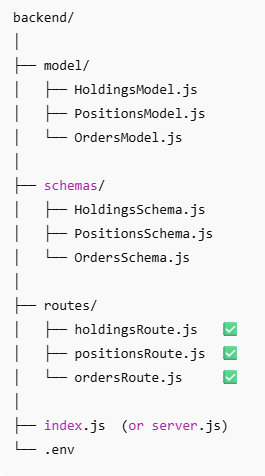
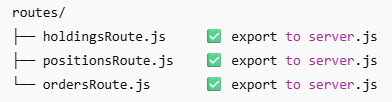


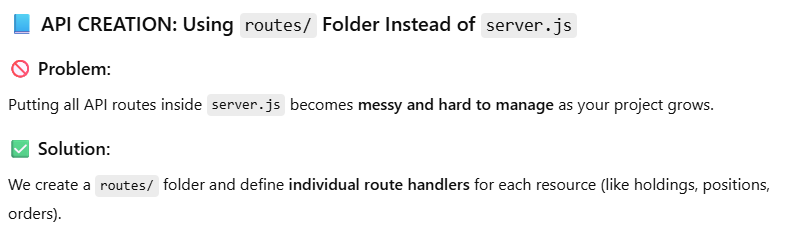


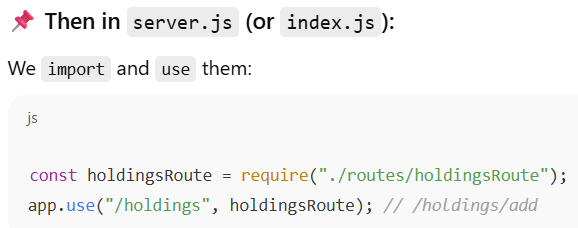
Also update server.js with CORS and json Parser   

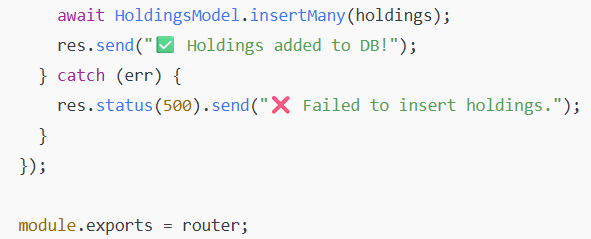
API CREATION :

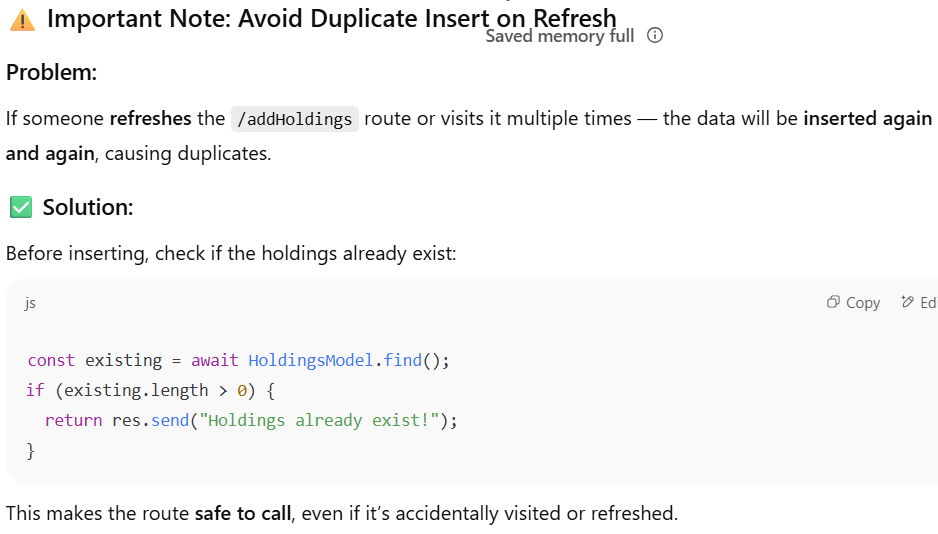
Instead of creating in server.js it creates messy so   
 

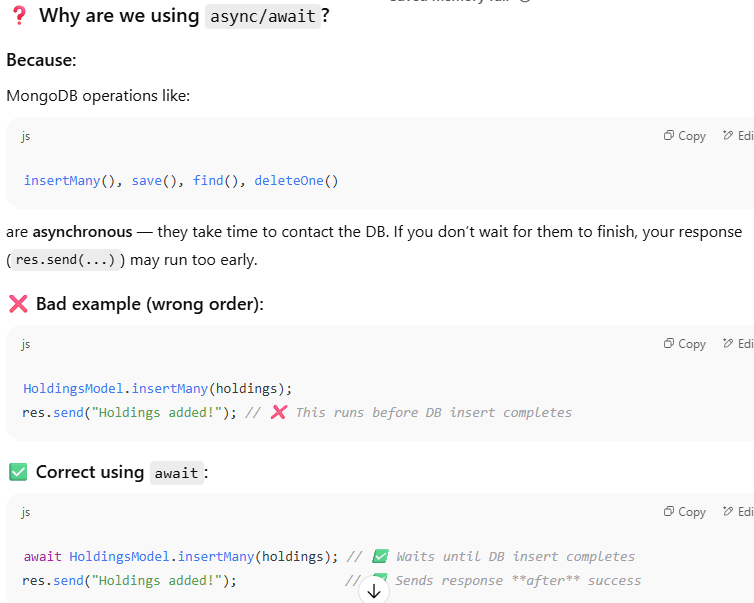




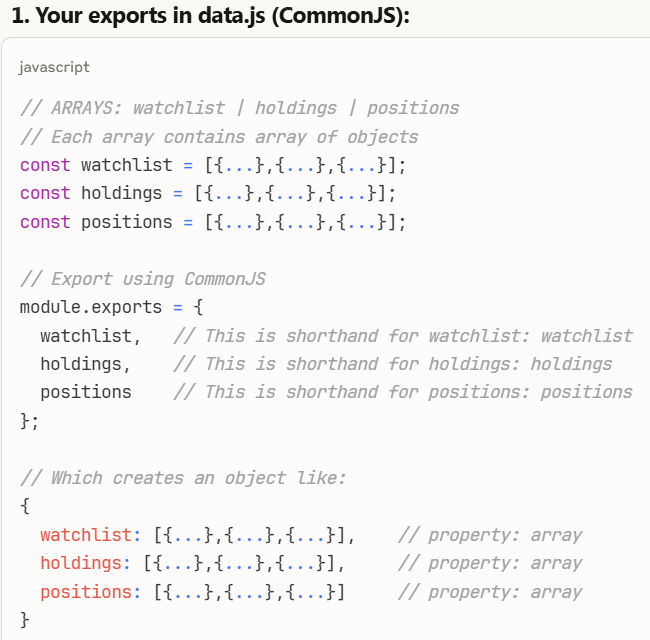


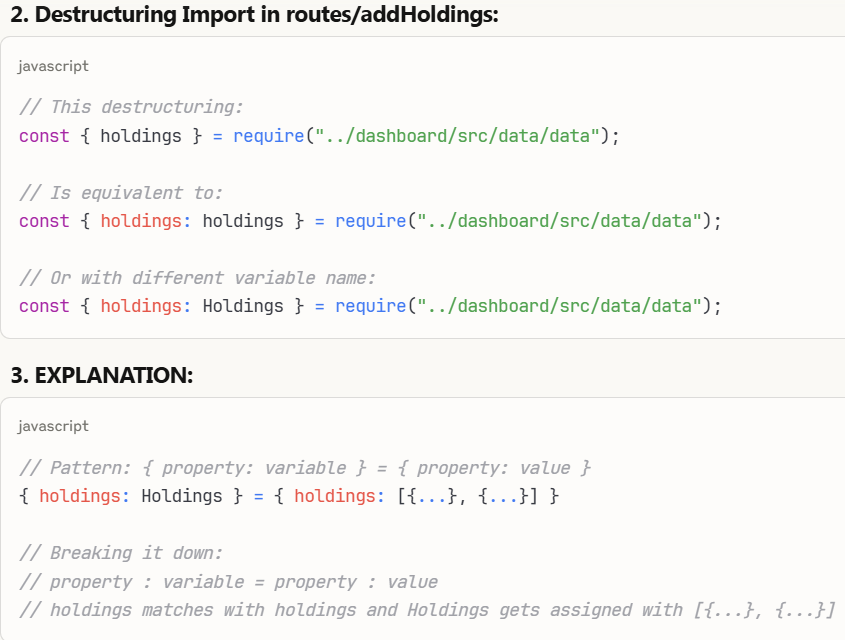






Confusion :



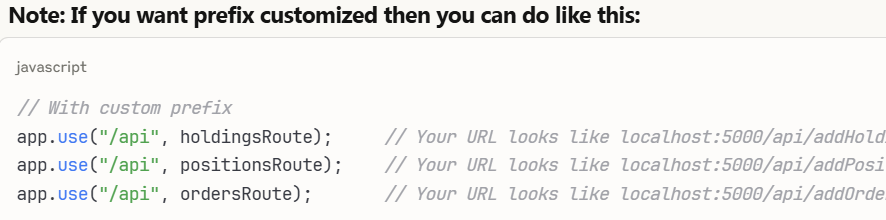


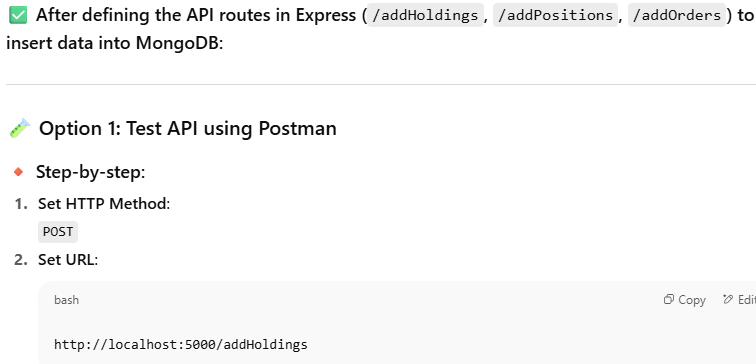
Now similarly for orders and positions ;

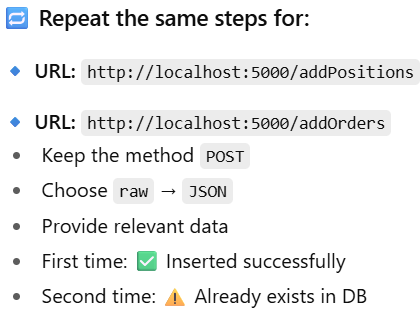
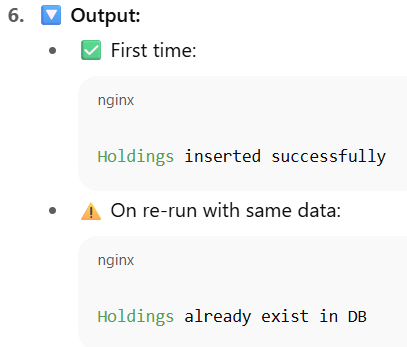


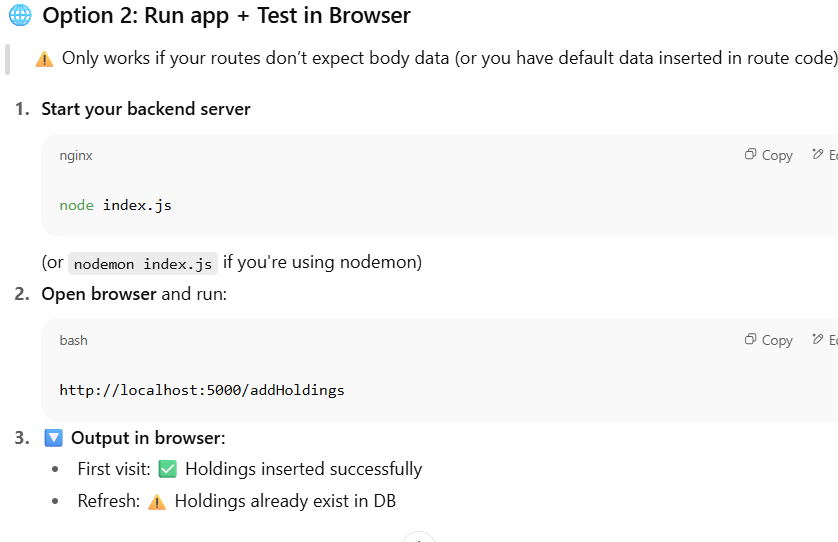
Note: add all routes in the server.js

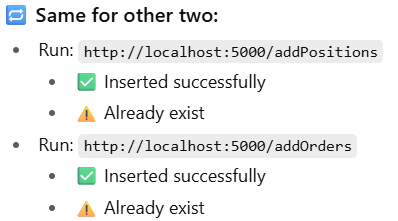


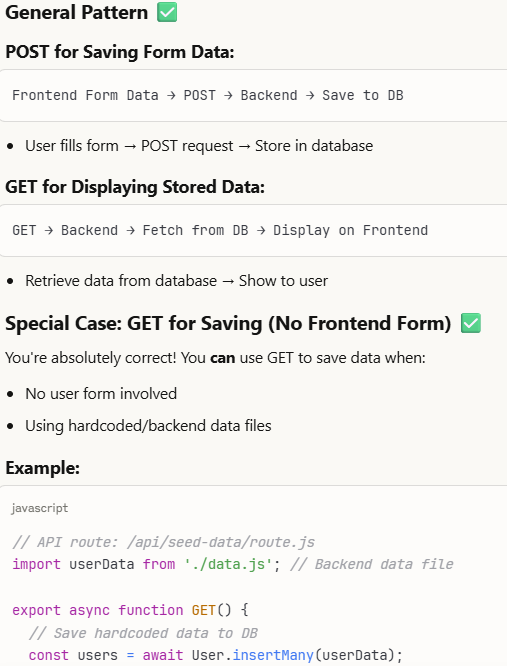






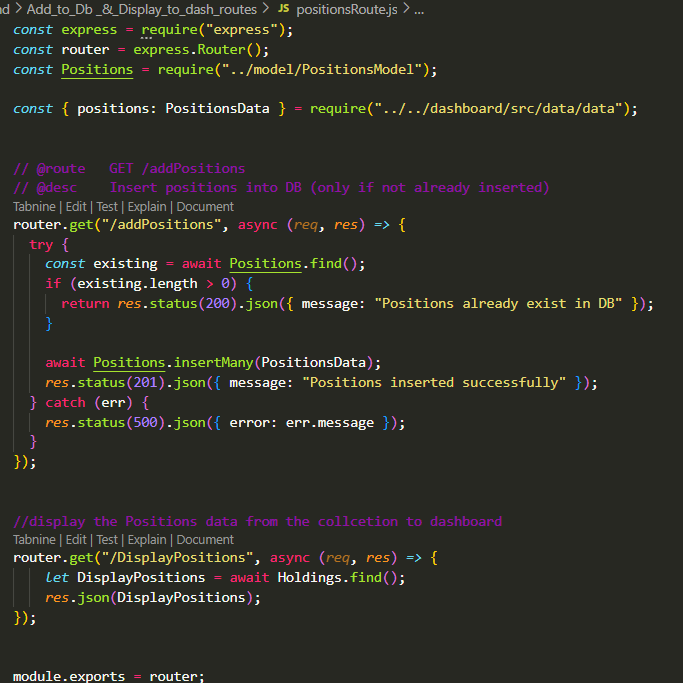




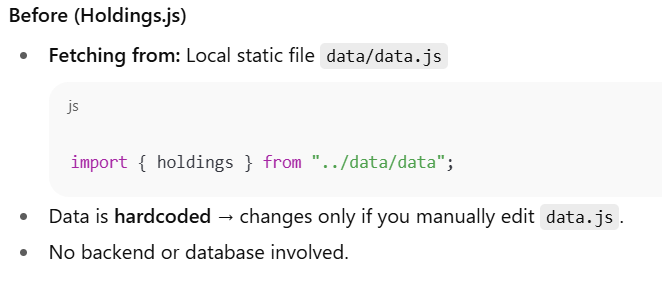


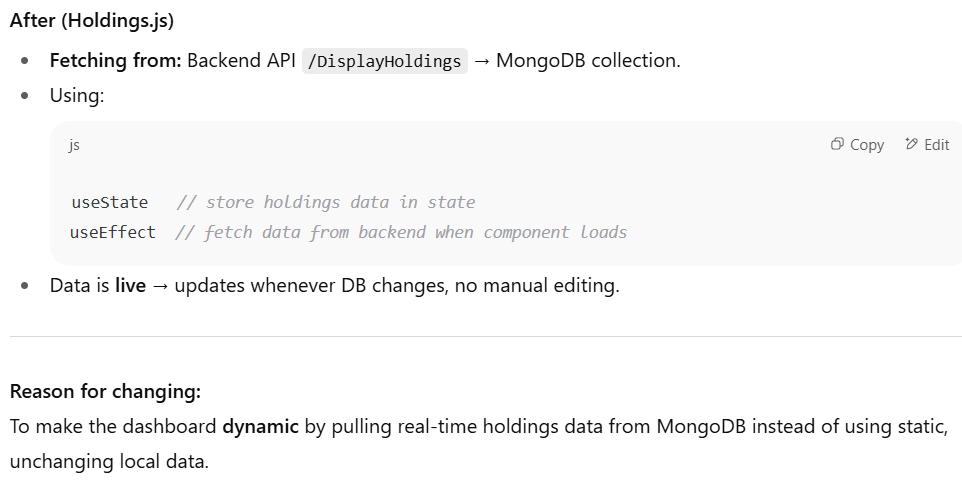
Now How to Display The holdings/orders/positions data in dashboard:

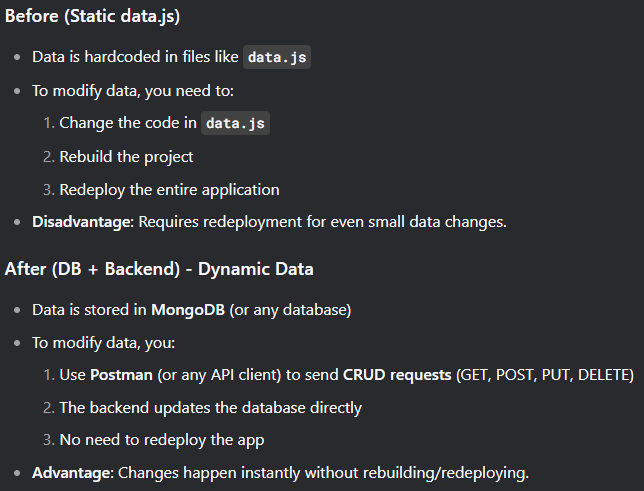
* Backend → Reads from MongoDB and sends JSON
* Frontend dashboard → Fetches JSON from backend and renders it

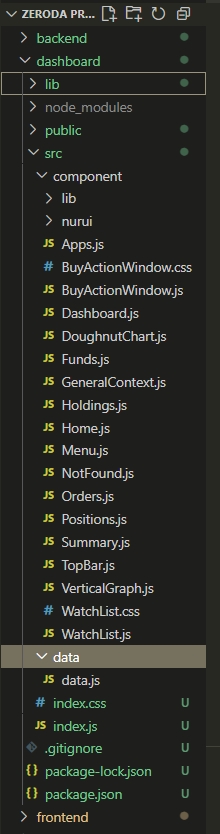


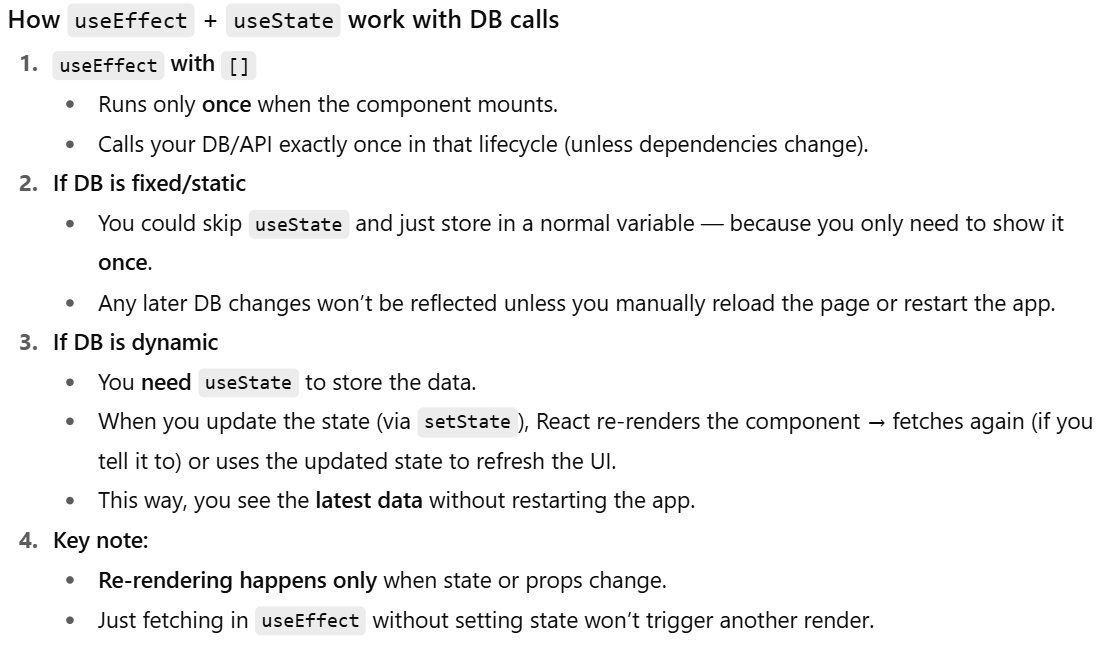
Similalrly for others also :

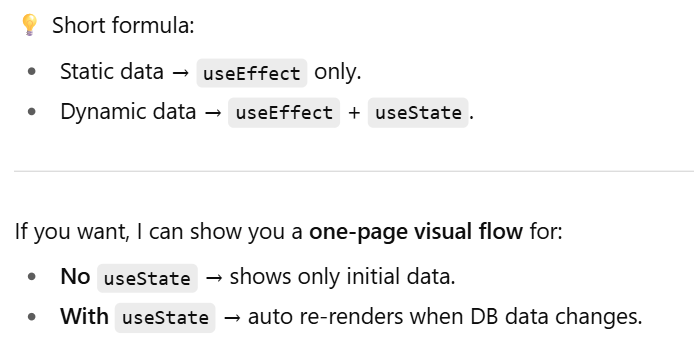
Now in the Dashboard {React Folder} :  




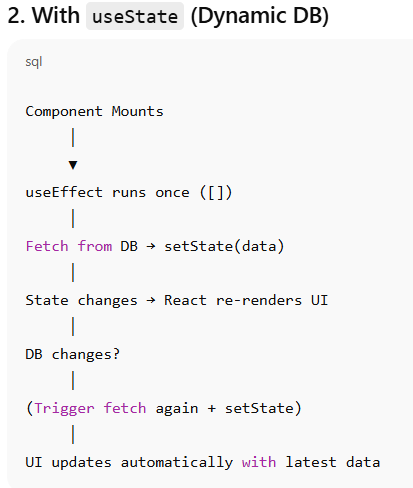


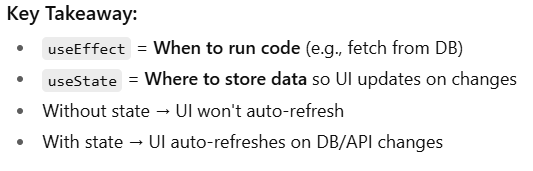
refers use state and useeffect pdf







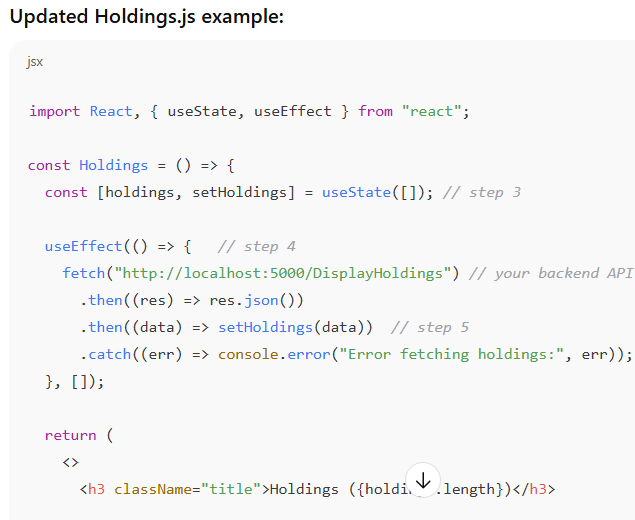


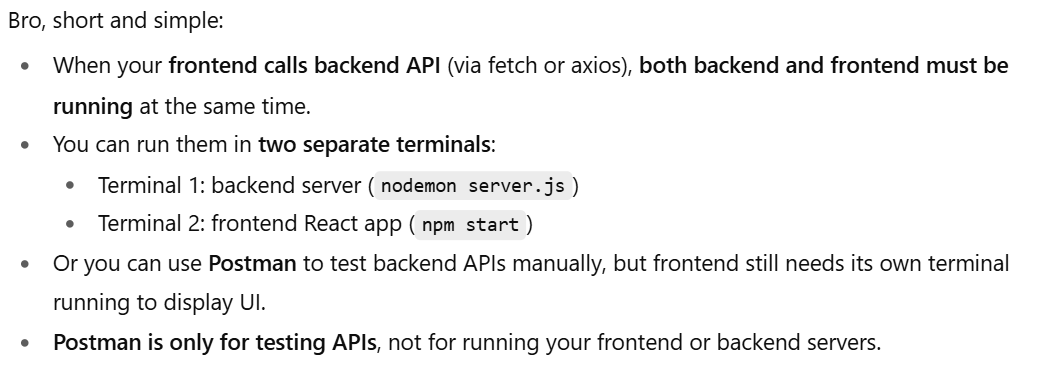


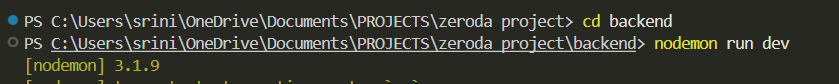
So its optinal to use the usestate ;

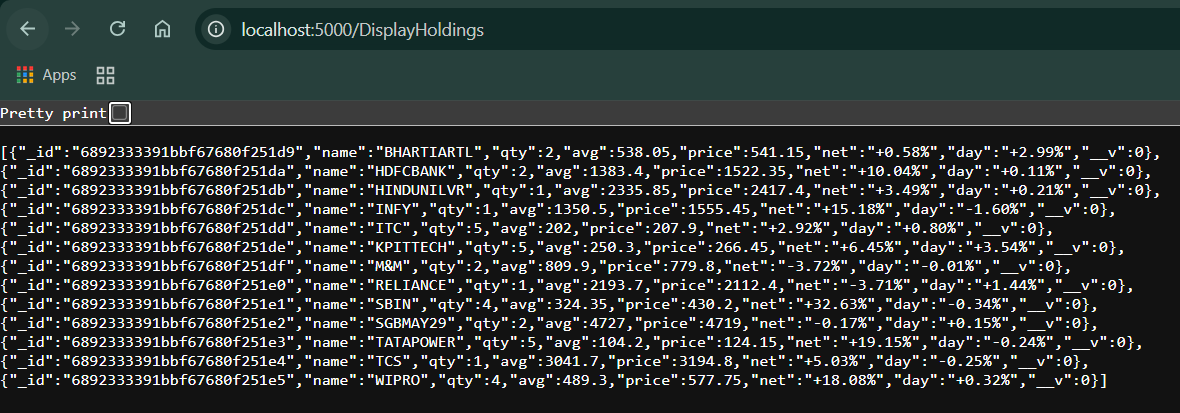


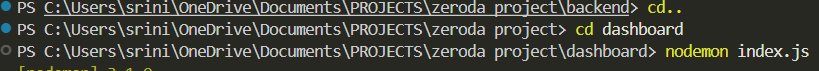
To

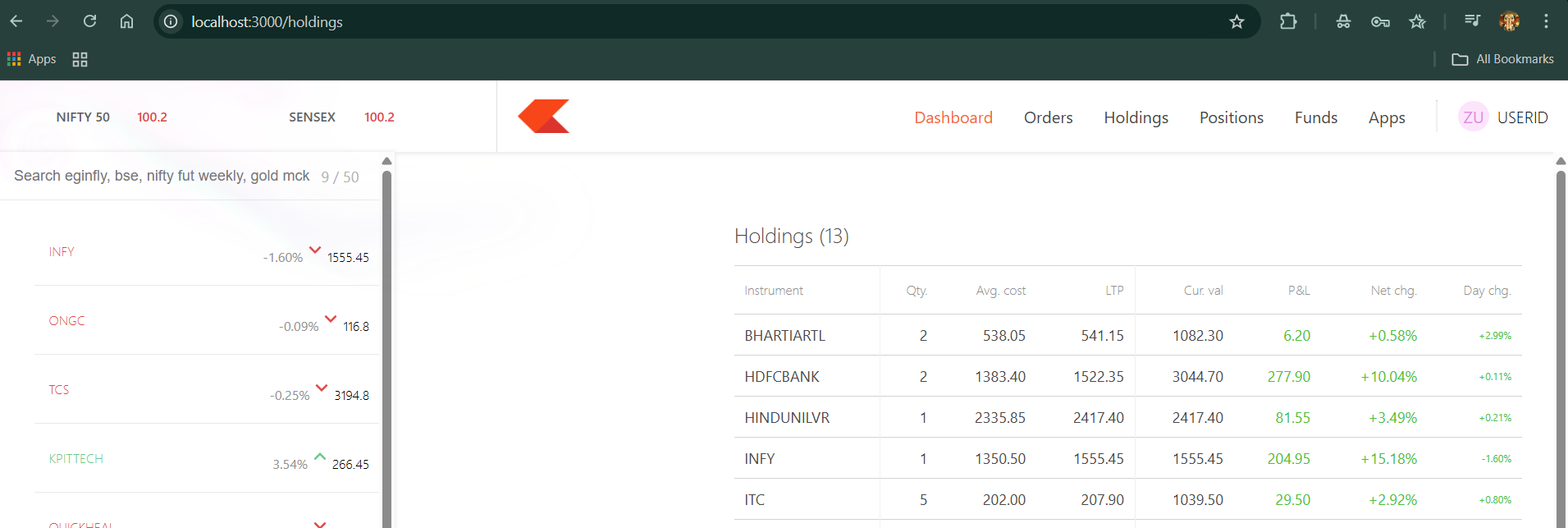


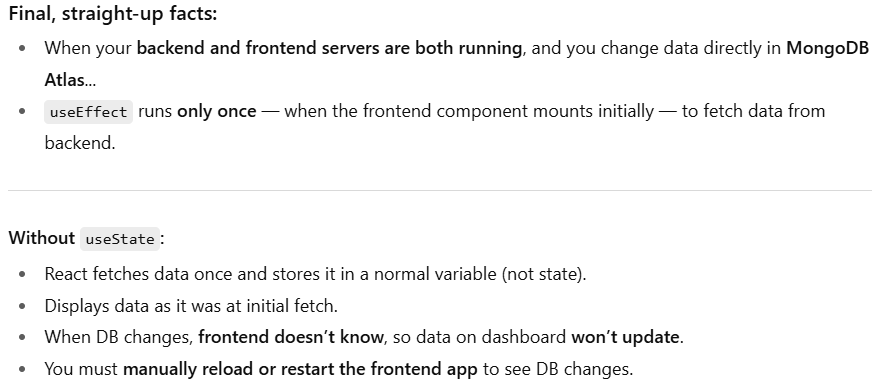


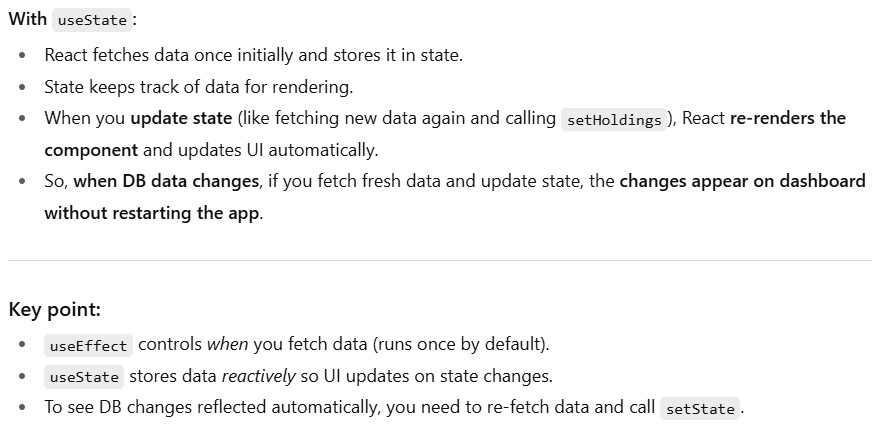




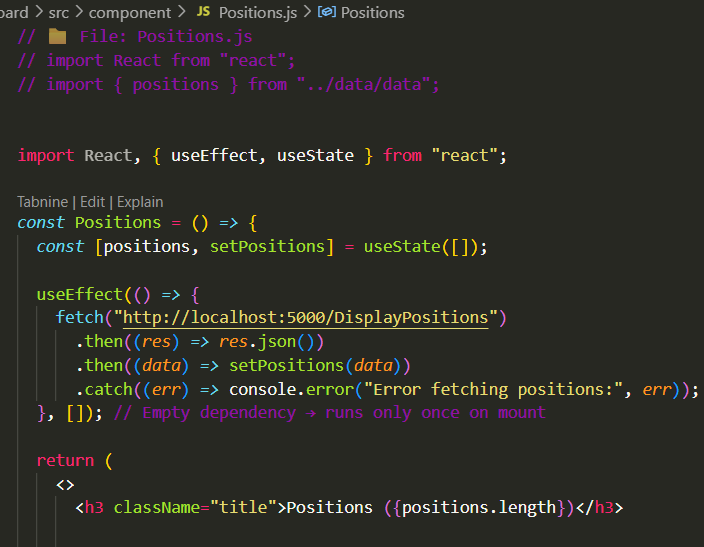


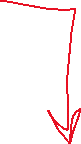


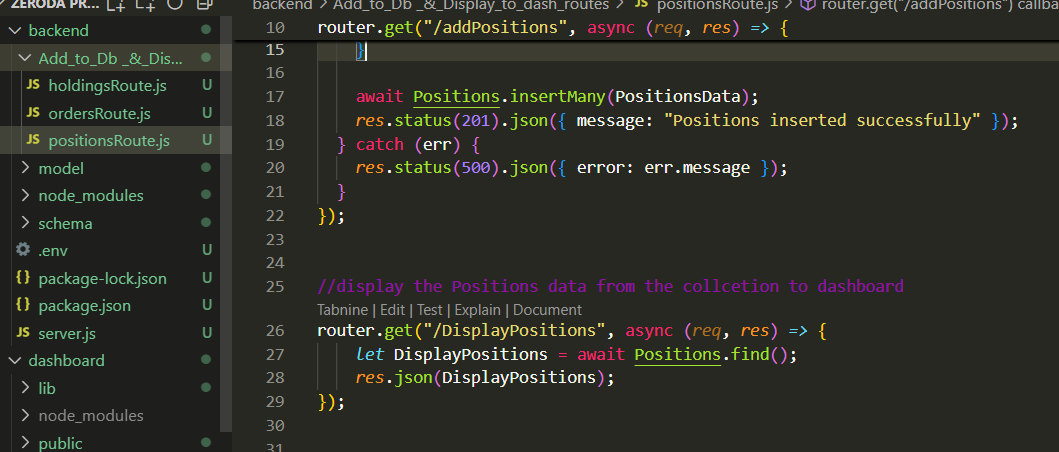




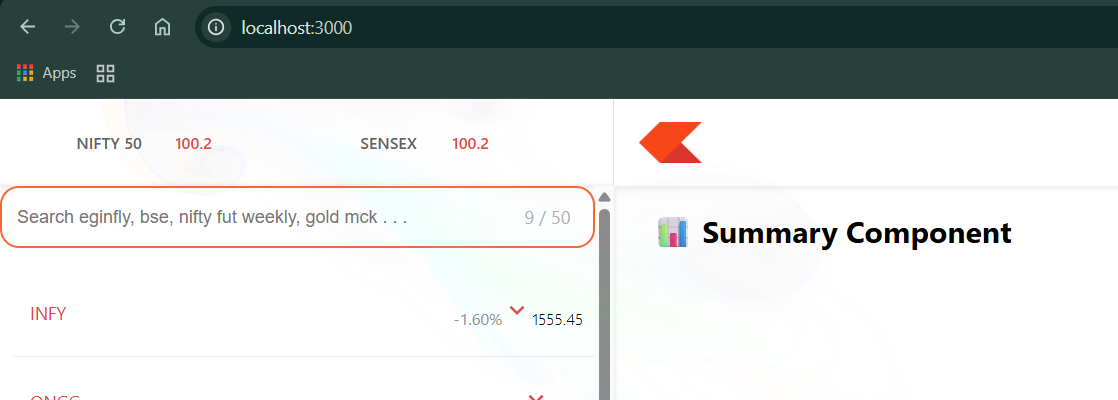
Simialry for postions data



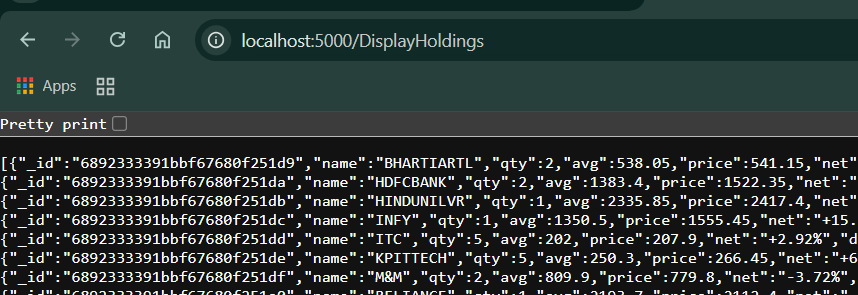


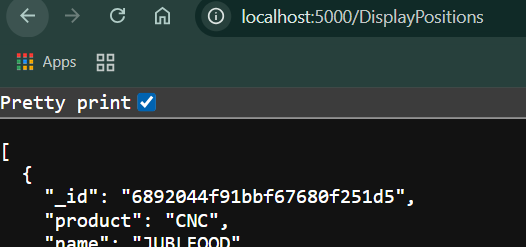


Terminal 1: run dashboard :npm start   
opens



Terminal 2: run backend :nodemon npm run

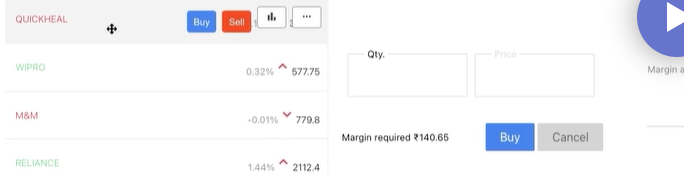
Opens 



Now lets work on crud from the dashboard:

Open dashboard





Refers create conext padf

GeneralContext.js