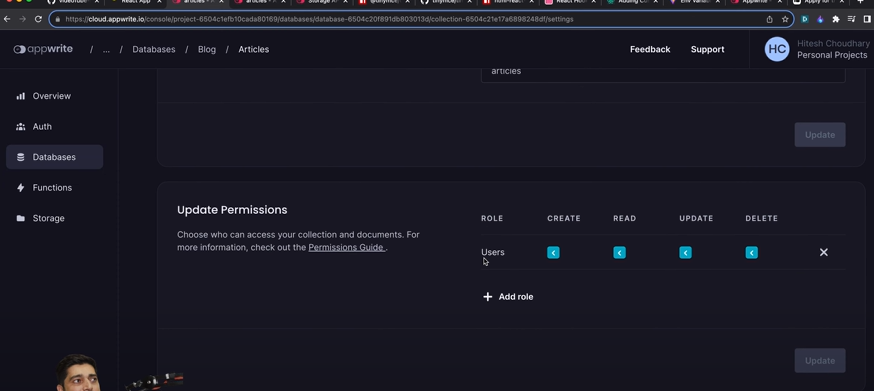
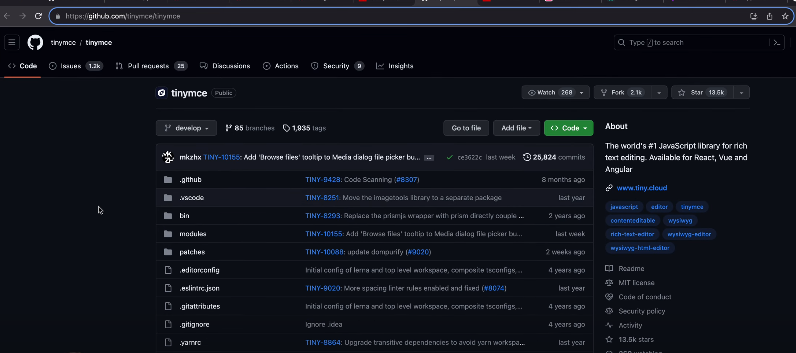
Mego Project

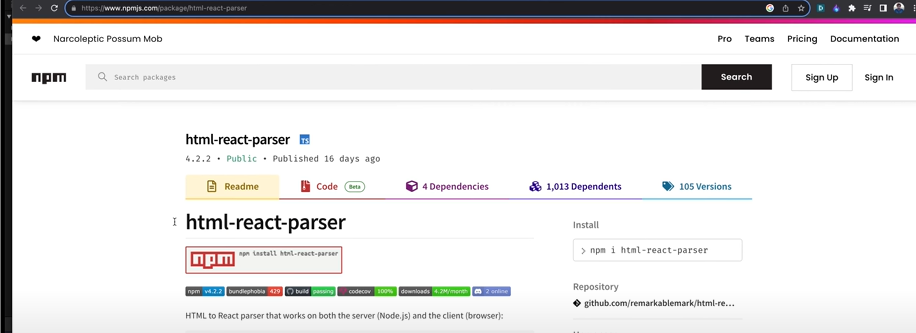
1. Appwrite (backend who will authenticate user,update,delete )



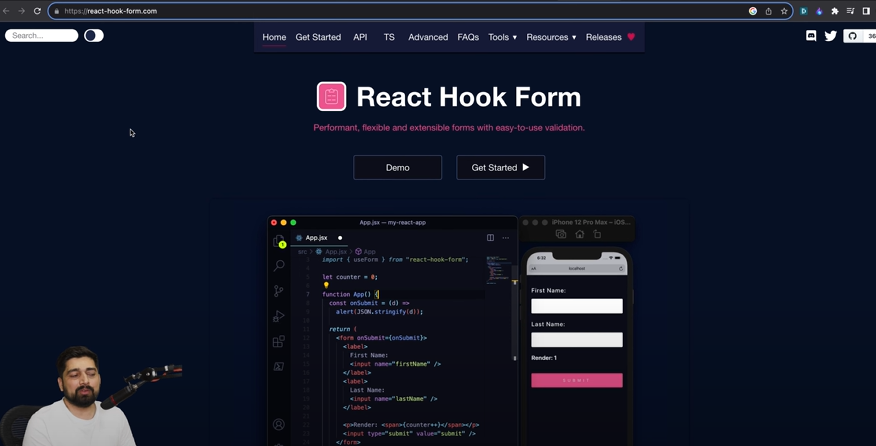
1. Tinymce(text want to bold,italic…)



3.html -raect-parser(as we can get html directly so we want html parser)



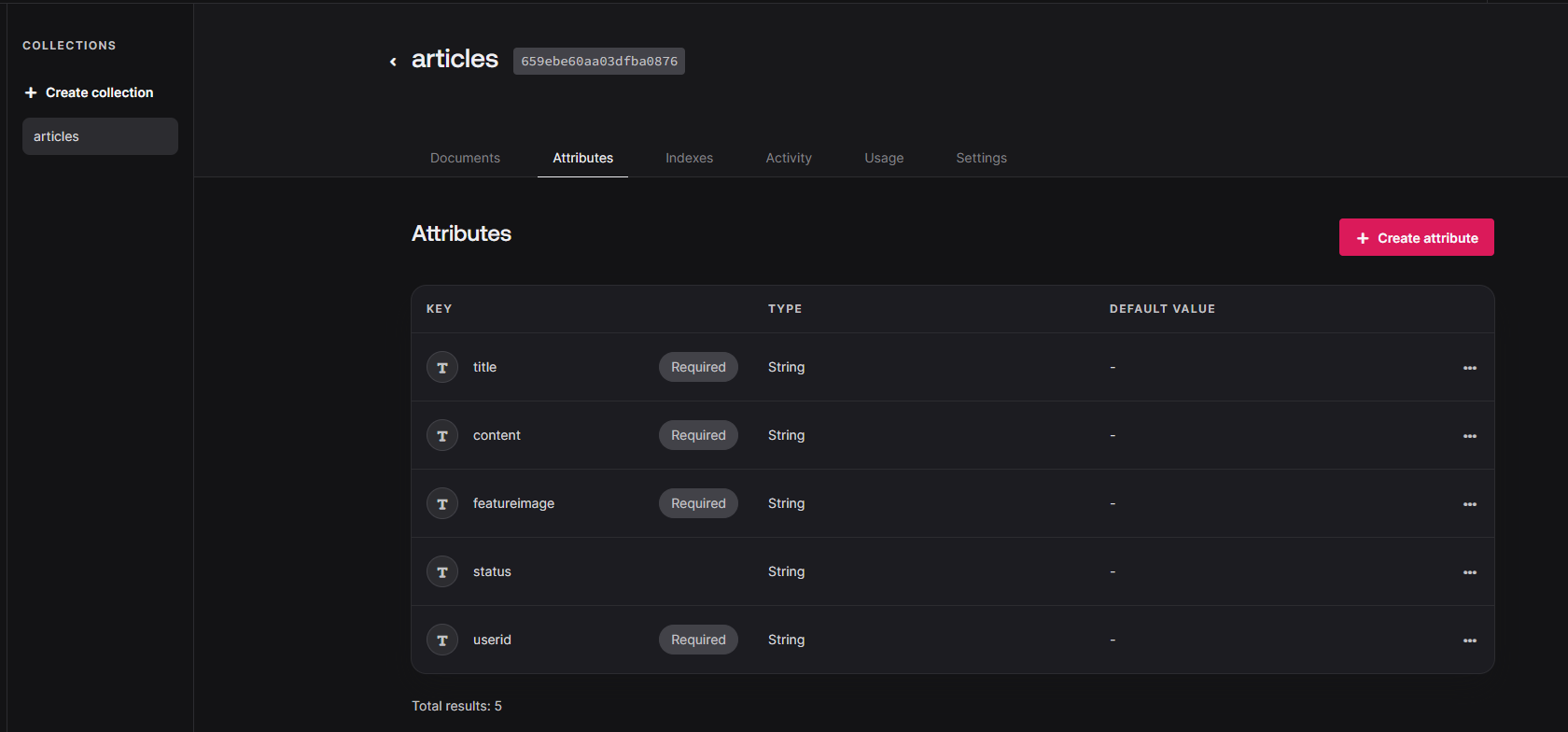
4.React hook Form



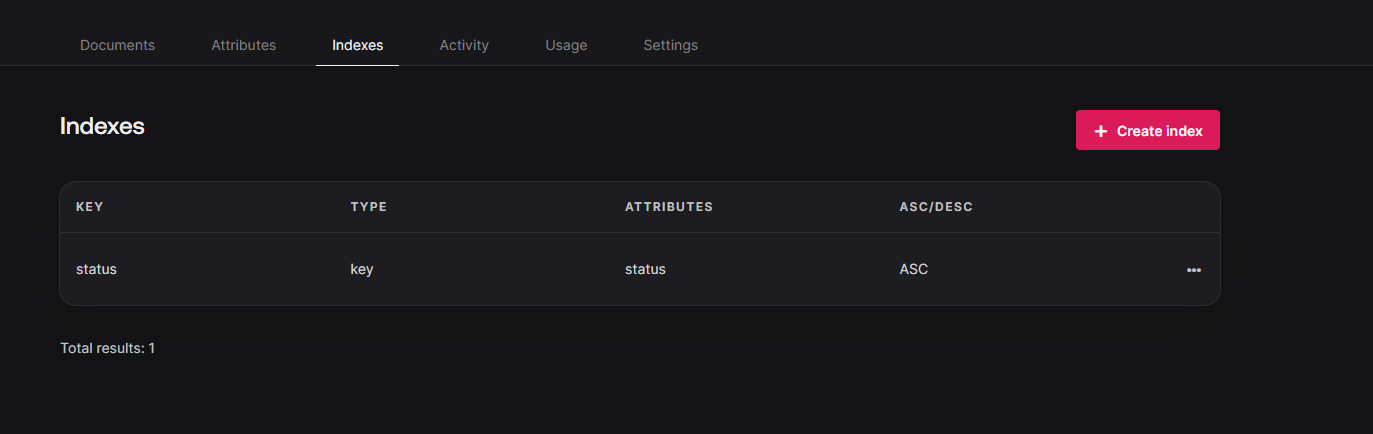
Now create the Project

1. Appwrite

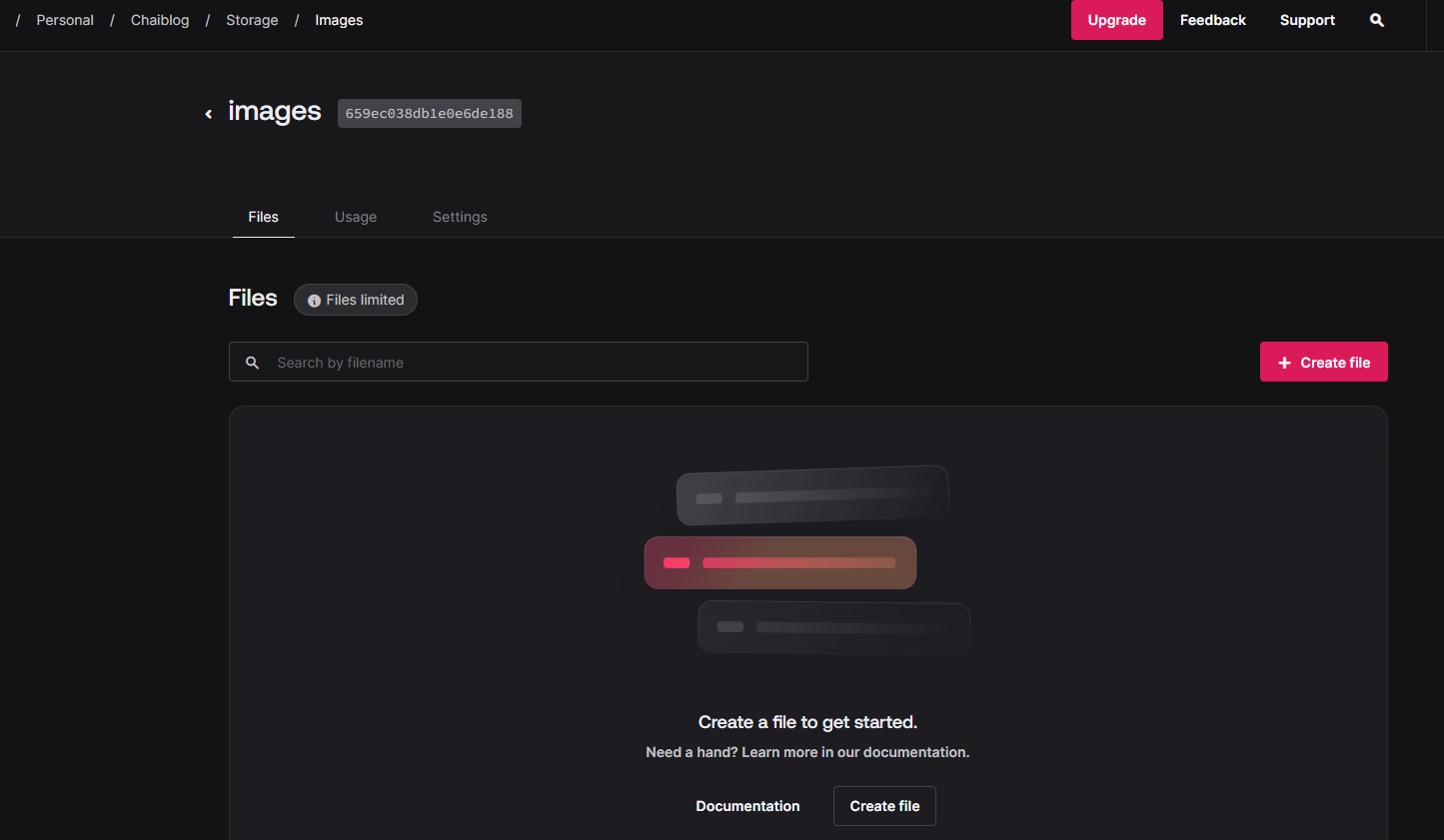
Create database – then craete collection 🡪 then attributes insert



Collection name- articles

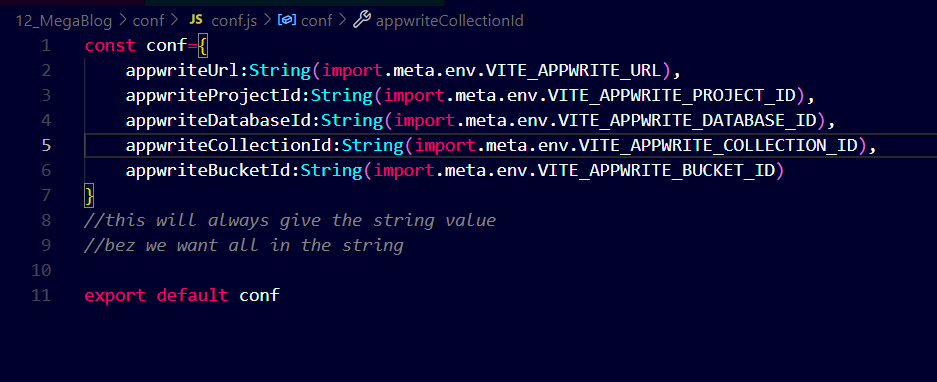


Give indexes and setting have permission

1. 

Storage

1. Env done in another file so no console.log()



Do this and then use in app.jsx so that string() values will be getten only-🡪

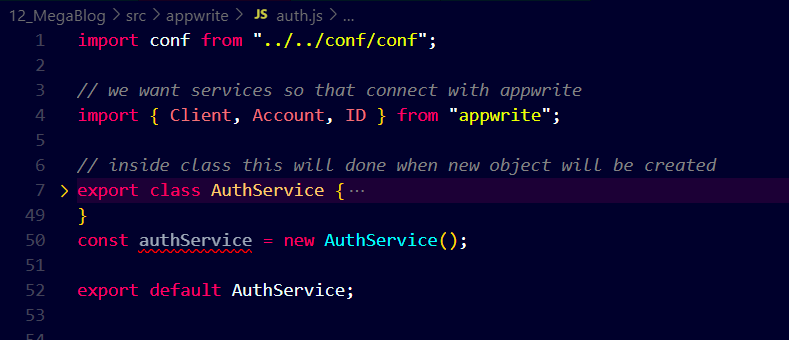
Env varibale always want string()

20th video---------------🡪>>>>>

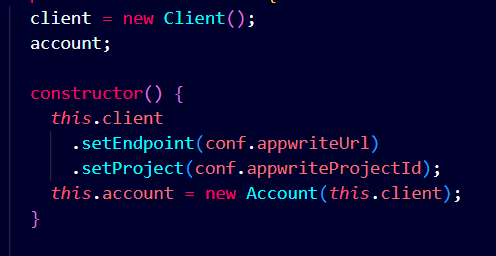
1st what are the things for the connection -🡪client,Account,ID

Then all the things signup,login,singout wrap in the class

Then make new object of this class and then export that class

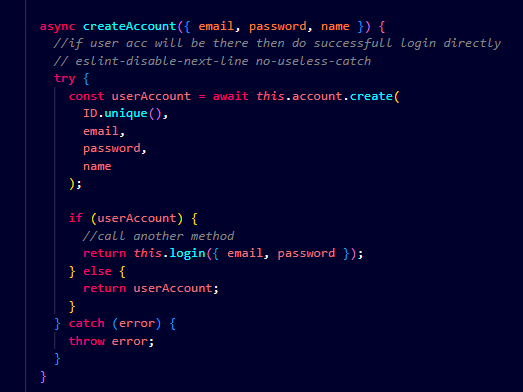


Needed

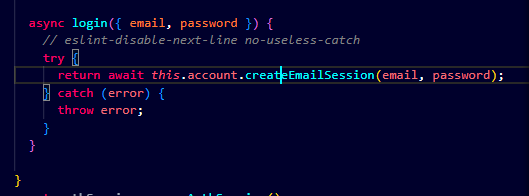


Then create acc🡪

By try and catch take all the things if useracc present then do login else return useracc



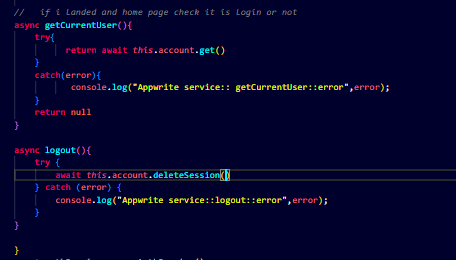
Then login



Get current user and logout

Account.get 🡪 give all the session

Session deleted as there are many code and methods are there



Here in 1st configuration we see:  
authentication,createaccount, login,getcurrentuser,logout

(want url,projectId)

Now we want another services configuration:-- related to post

(url,projectId)

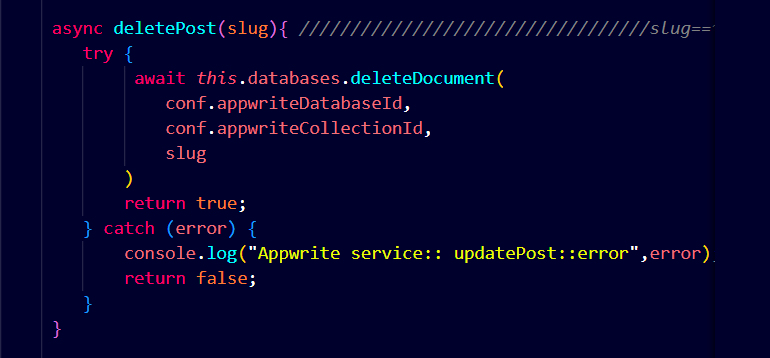
1.createPost -🡪 databaseId, collectionId ,attributes



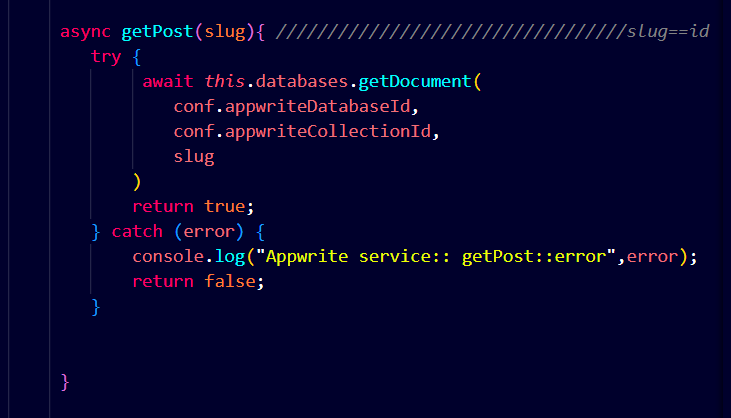
2.Update post --- > here attributes will be updated

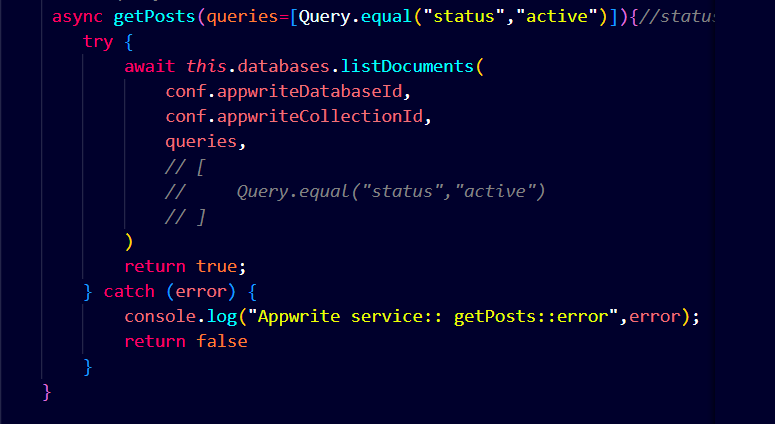


1. Delete post

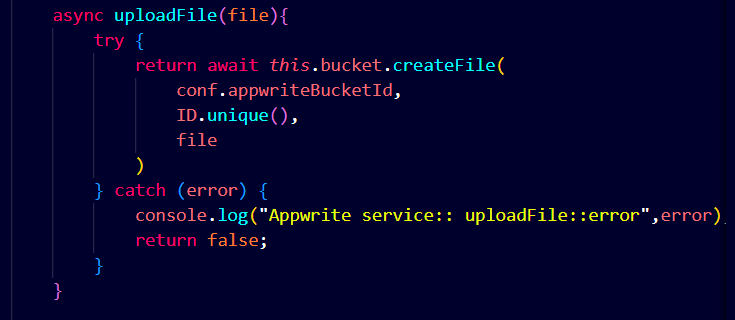


1. Getpost





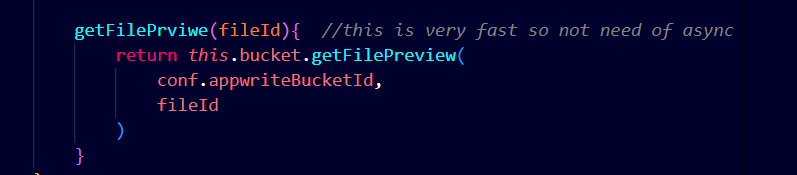
1. Upload file



1. Delete file -🡪 get fileId for delete using Bucket



1. getFilePreview



Post update delete get by -🡪 database, collection inside that slag is there where attributes are there and there is post

File-🡪bucketId inside that this will be present with fileId

22🡪

Redux store will be there and we want to config this

Redux have store --- reducer ---dispatch--- selector

1.store



2. reducer



3.give provider



4.App.jsx--- dispatch



If statement get userData fron getCurrentUser()

If 🡪 data is then do login

Else🡪 logout

Finally 🡪 loading false so go to landing page

Components



Use this in tha app.jsx

