Para inicar uma api com prisma express e ts

Comandos

1 ) yarn init -y

2) yarn add @prisma/client express

3 ) yarn add –dev @types/express @types/node nodemon prisma ts-node typescript

PRISMA

4 ) npx prisma init --datasource-provider sqlite

5 ) vai nor arquvi prisma crire as tabelas

// This is your Prisma schema file,

// learn more about it in the docs: https://pris.ly/d/prisma-schema

generator *client* {

  provider = "prisma-client-js"

}

datasource *db* {

  provider = "sqlite"

  url      = env("DATABASE\_URL")

}

model *Autor* {

id  *String* @id  @default(uuid()) //auto imcrement do prisma

name *String*

livros *Livros*[]

}

model *Livros*{

id  *String* @id  @default(uuid()) //auto imcrement do prisma

autor *Autor* @relation(fields: [autorId] , references: [id])

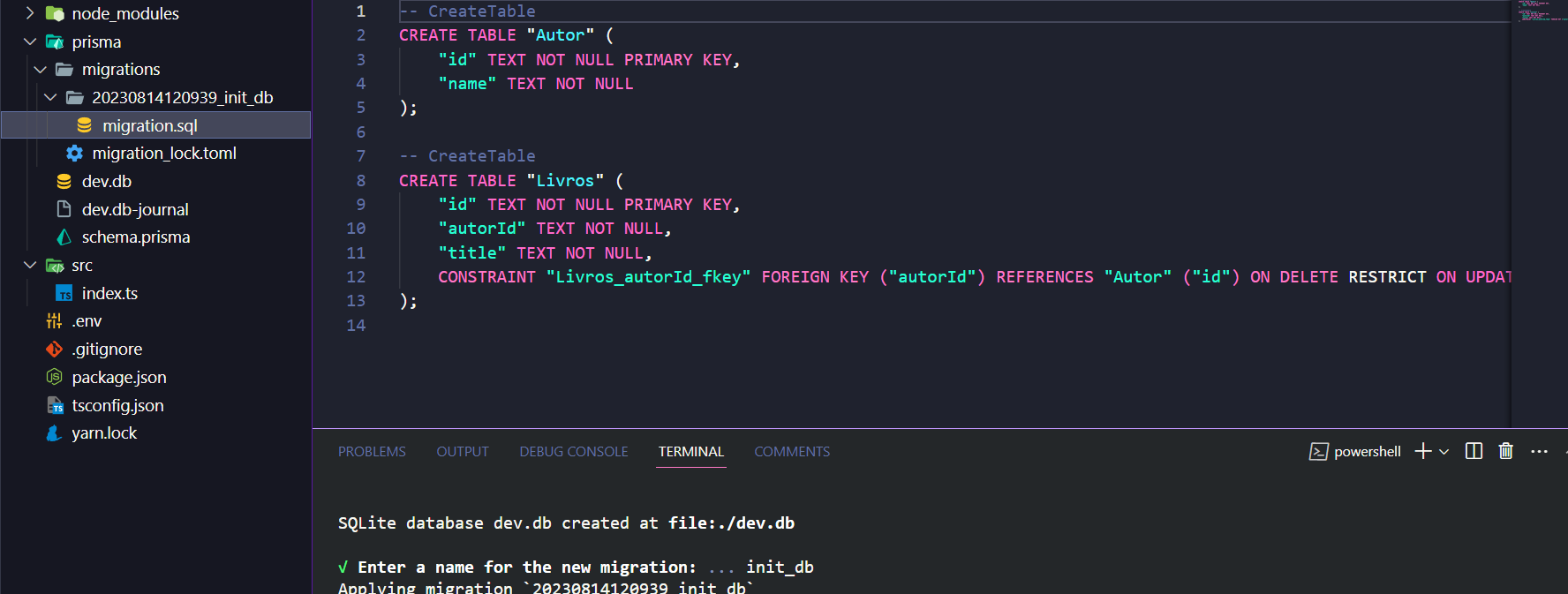
autorId *String*

title *String*

}

6 passo após o criamento delas temos que fazer uma migration para aplicar isso no banco

7 ) yarn migration ( esse migration é um script que criamos na pasta json)  
8 ) dar um init\_db ai ele vai criar uma pasta migation aode vai ter um endPoint da alteração sql feita



Após isso prisma configurado

EXPRESS

1 ) ciar uma pasta src com um arquivo index.ts dentro com um certo código

import express, { Request, Response } from "express"

const app = express()

const port = process.env.PORT || 8080

app.use(express.json())

app.get('/ping',(req : *Request*,res : *Response*)=>{

res.status(200).send("running")

})

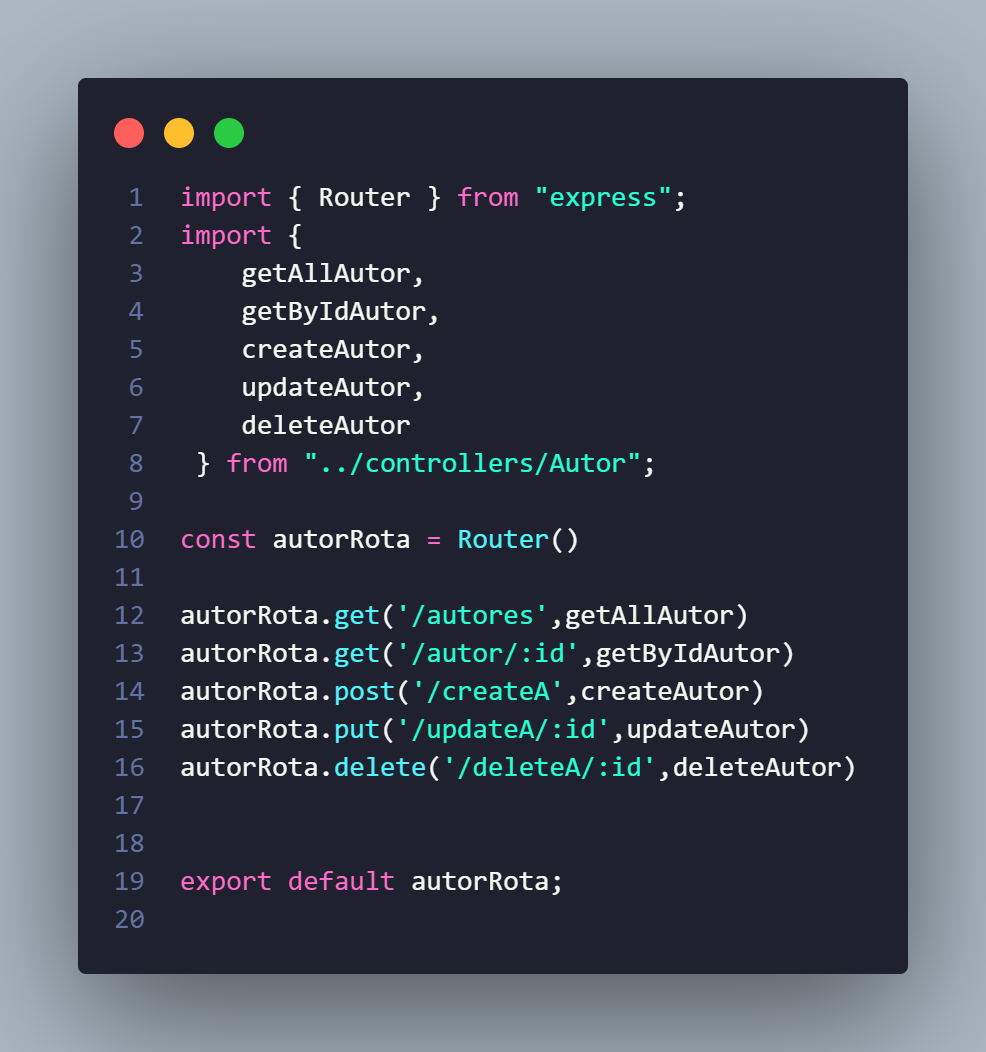
app.listen(port,()=>{console.log(`server rodando na porta: ${port}`)})

2 ) passo criar 2 pasta uma de controller e outra de routes

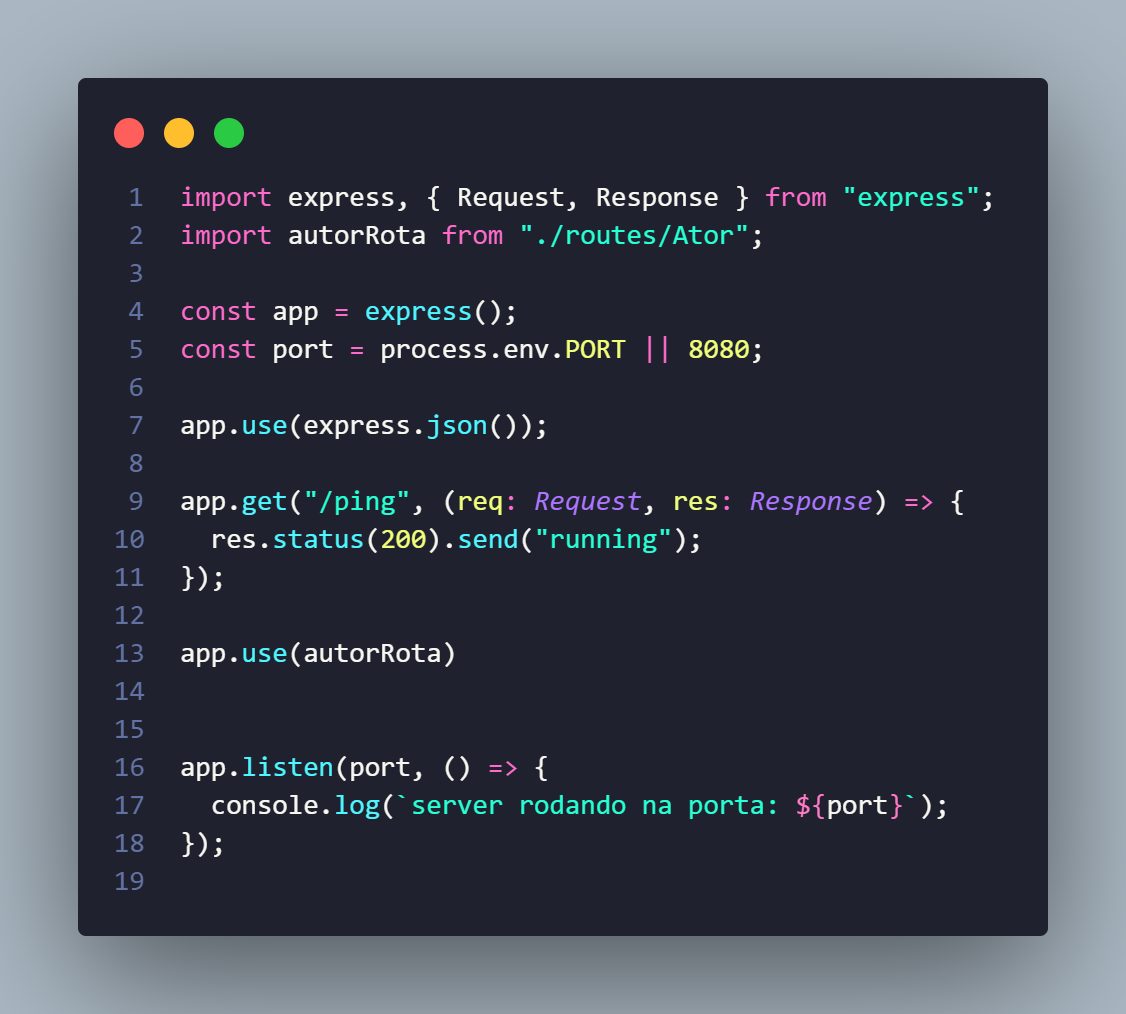
3) na pasta de controller é aonde vai ficar os métodos relacionados com o banco usando o prima



4 ) próximo passo fazer arquivo de rotas:



Após isso fazer a config do arquivo principal (index.ts)



IMPORTANTE

export const createLivro = async (req: *Request*, res: *Response*) => {

  try{

    const livroData = req.body

    const livro = await Livros.create({

      data : {

        title : livroData.title,

        autor : {

            connect : { id: livroData.autorid}

        }

      }

    })

    res.status(201).json({data : livro})

  }catch(err){console.log(err);}

}

Aonde vc coloca esse conect é que ele vai fazer a conexão e puxar dados de uma outra tabela