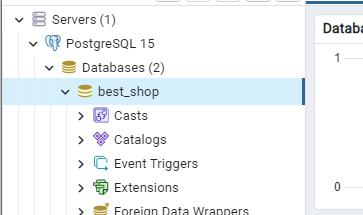
Npm init –y

Npm install express

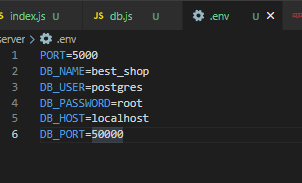
npm install pg pg-hstore sequelize cors dotenv

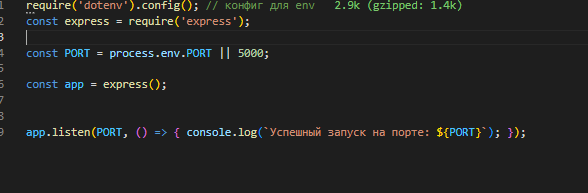
npm I –D nodemon чтобы при разработке при изменение кода не перезапускать сервер каждый раз

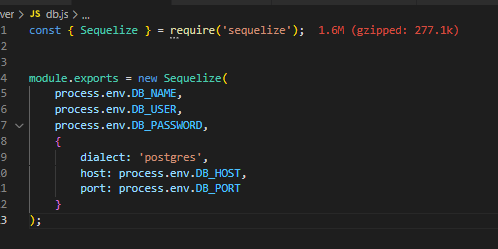
создал бюазуданных в постгресс



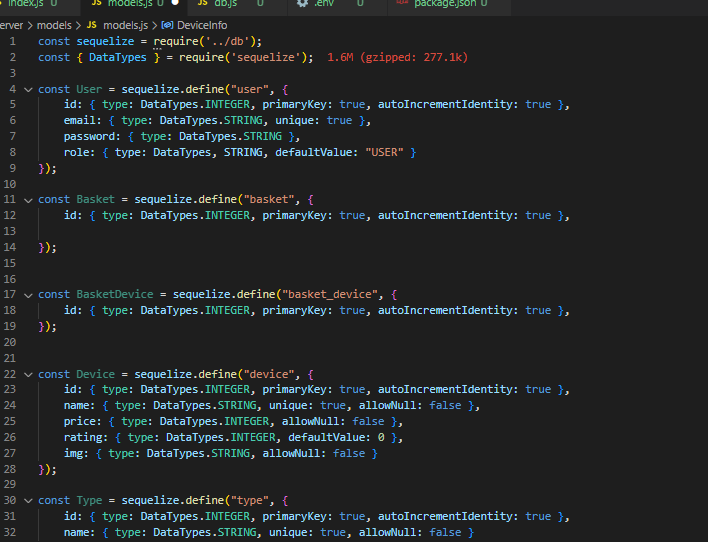
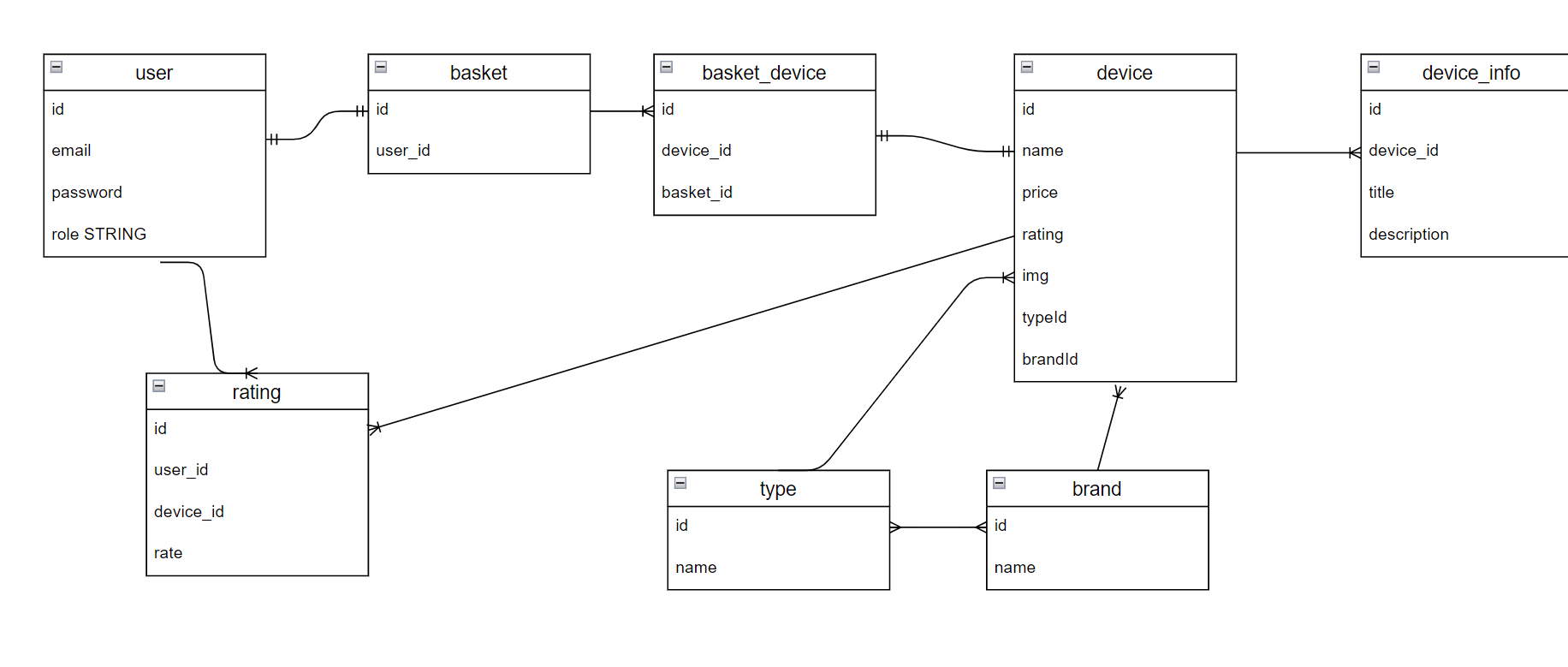
1599 пароль везде порт 50000



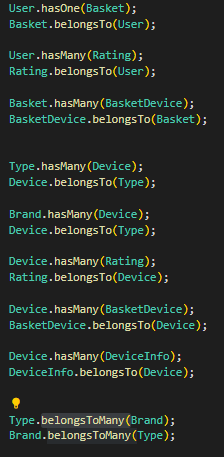




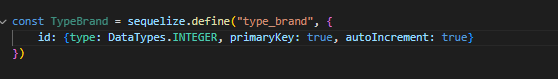
Функция для подключения к базе данных является асинхронной, еще нужно импортировать секвалайз из db

Описал модели

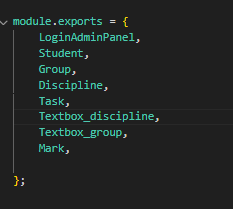
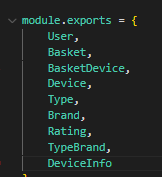
Делаем связь

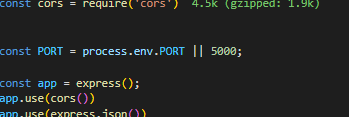


Но для многих ко многим нужно сделать промежуточную таблицу



Он сам для нее остальные поля ключи создаст

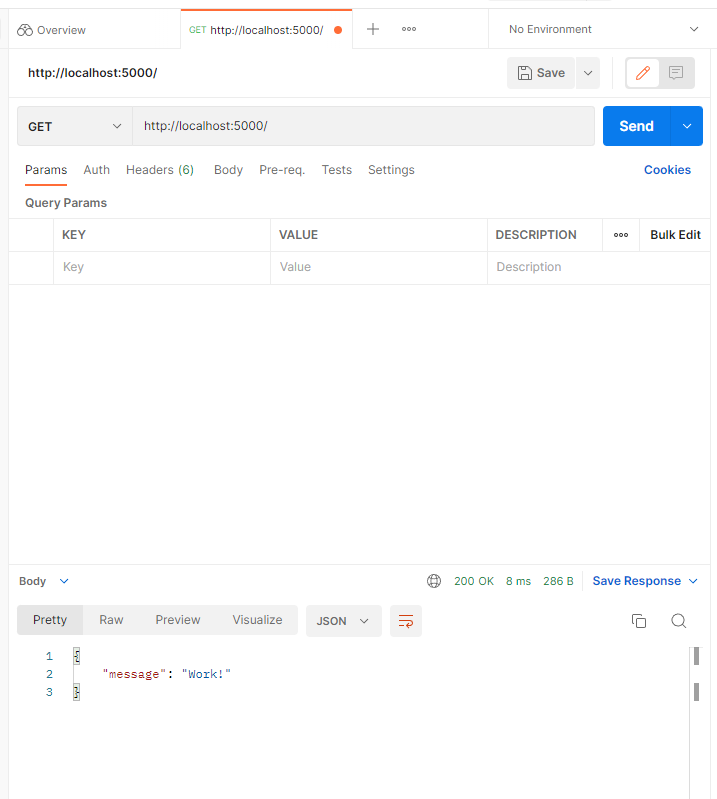
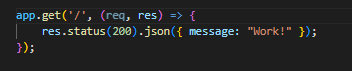
Экспорт 

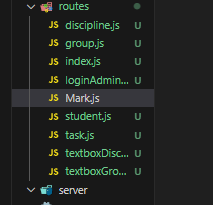


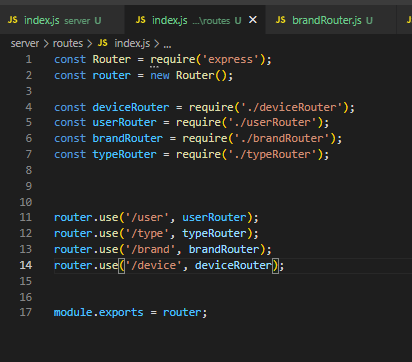
В индекс жс КОРС нужен для кправильной работы с браузерами что-то там потом можно больше изучить

<https://web.postman.co/workspace/My-Workspace~18b59059-98c1-420e-8a90-e4f36eb75e4e/request/create?requestId=498cfdad-175e-4204-bf01-63ffddc6eb13>

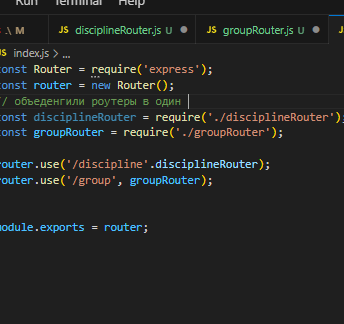
для теста запросов

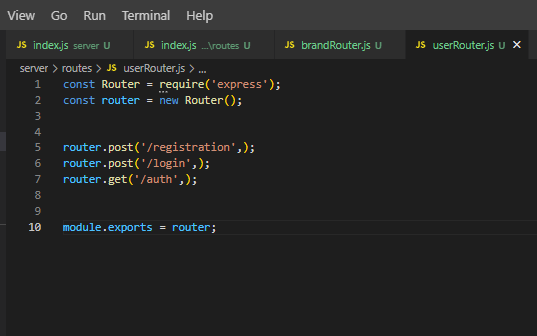


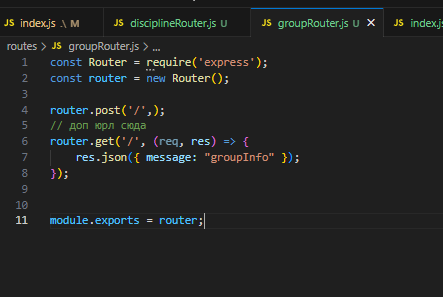
Связующий среди маршрутов для запросов

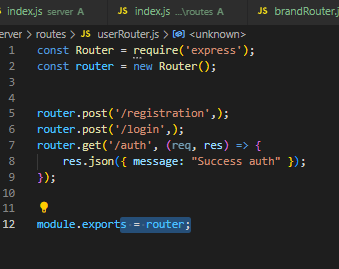


Запросы для пользователя. Надо сделать роутер.делет

Объеденили роутера в один



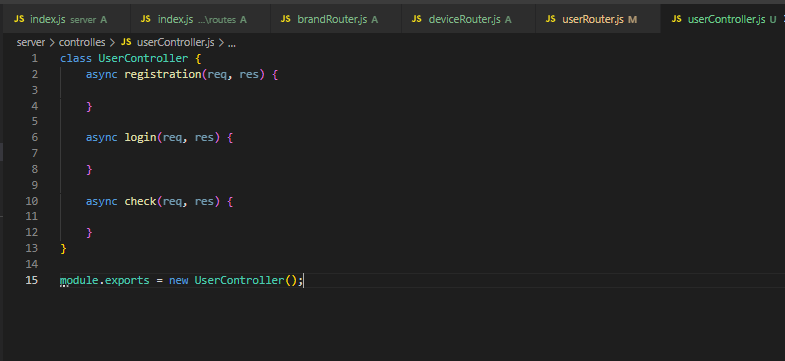




app.use('/api', router);

Через все роутеры по адресу <http://localhost:5000/api/user/auth> мы получаем успешный

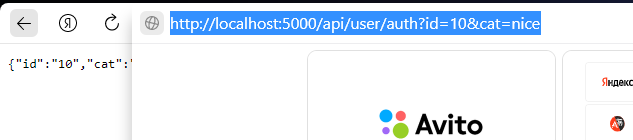
Но так делать не нужно, создадим папку контроллерс, где для каждого маршрута распишем запрос ответы

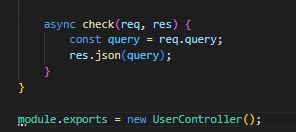


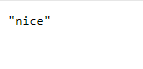
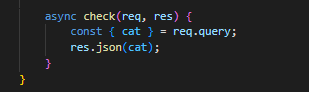
Передаем их в маршрут вторым параметром как объект, не вызываем.

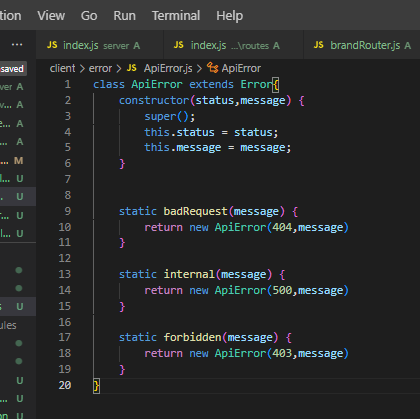


Через query запрос можно брать то что в юрл вводят







Конструктор класса ошибок 

Промежуточный обработчик ошибок

