**Выбранный стек технологий.**

В выбранный стек технологий входят:

1. Node.js
2. Express
3. Prisma
4. PostgreSQL

В качестве редактора кода будет использован Visual Studio Code. Для удобства работы с PostgreSQL используется приложение pgAdmin.

**Рассмотрим преимущества каждого аспекта выбранного стека технологий.**

1. Node.js

Node.js имеет высокую производительность благодаря использованию асинхронной модели ввода-вывода, что полезно для приложений с высокой нагрузкой. Одной из самых больших положительных черт является экосистема пакетов npm (Node Package Manager) - крупнейший репозиторий инструментов и библиотек для JavaScript, а JavaScript, в свою очередь, это универсальный язык программирования, используемый как на клиентской, так и на серверной части приложения.  
Node.js поощряет использование модульного подхода к разработке. Помимо этого, приложения на этой платформе могут быть легко масштабируемы благодаря асинхронной модели.

1. Express

Express является стандартным каркасом для Node.js при создании веб-приложений и API. Он эффективно использует ресурсы Node, благодаря чему имеет высокую производительность. Ключевой концепцией Express является middleware (связующее программное обесепчение), которое позволяет добавлять дополнительную функциональность к запросам и ответам.

1. Prisma

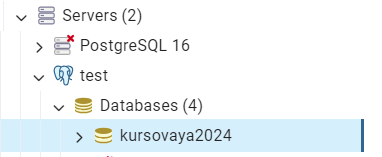
Prisma - это инструмент для работы с базами данных. Prisma отличается простотой в использовании и активным сообществом разработчиков. Кроме того, она поддерживает множество баз данных, таких как MySQL, SQLite, PostgreSQL и другие. Так же, Prisma обеспечивает безопасный доступ к данным благодаря генерации типизированных API на основе схемы базы данных.

1. PostgreSQL

PosgreSQL - объектно-реляционная СУБД, которая является полностью бесплатной с доступным исходным кодом. Обладает модульной архитектурой и обеспечивает надежность и целостность данных. pgAdmin, в свою очередь, сильно облегчает управление БД и мониторинг её промзводительности.

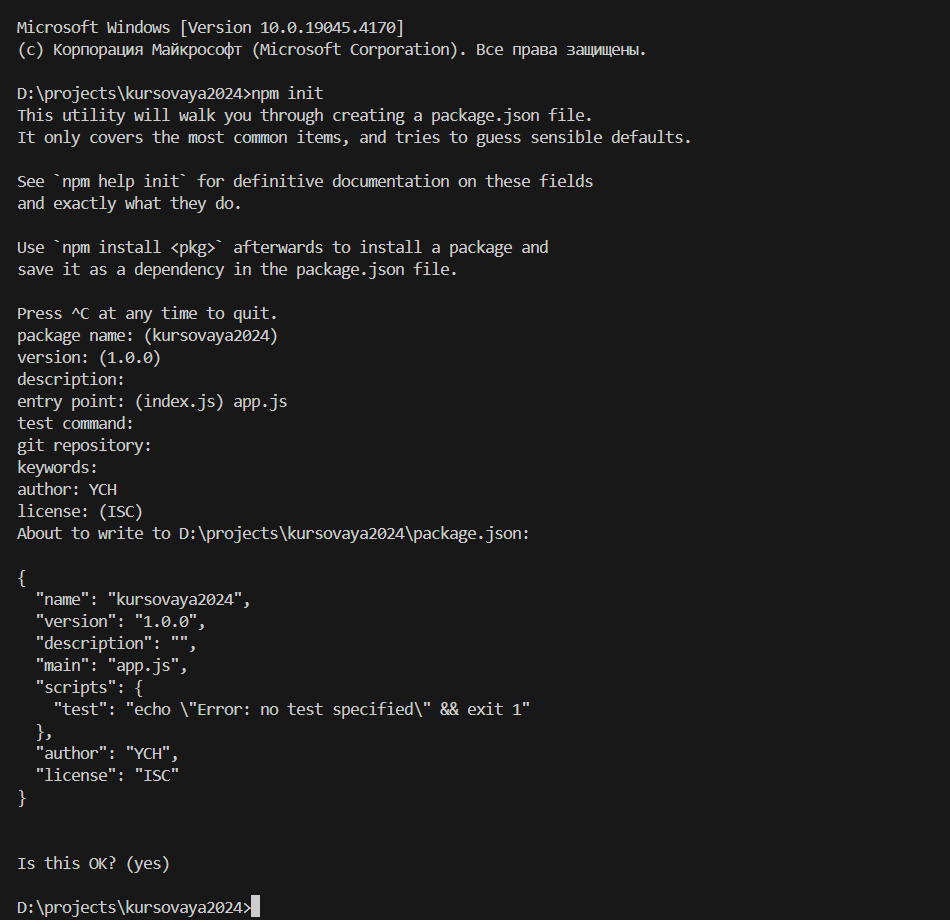
**Настройка выбранного стека.**

**Создание базы данных в pgAdmin**

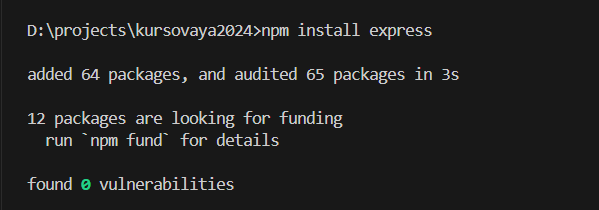


**Установка необходимых библиотек для Node js**

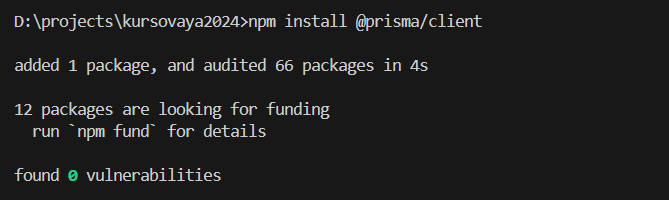
1. Создание проекта Node.js



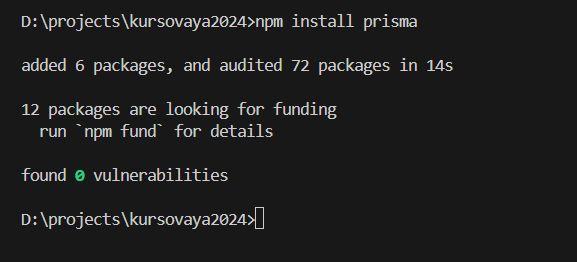
1. Установка Express



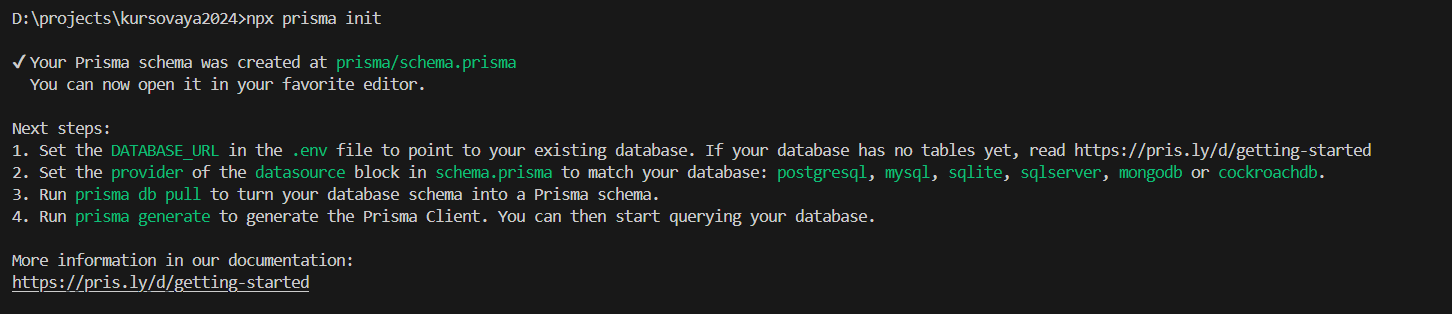
1. Установка клиента Prisma



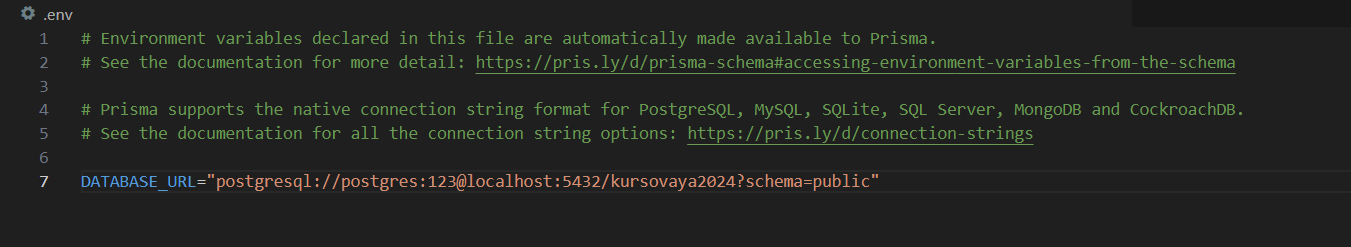
1. Установка Prisma CLI (command line interface)



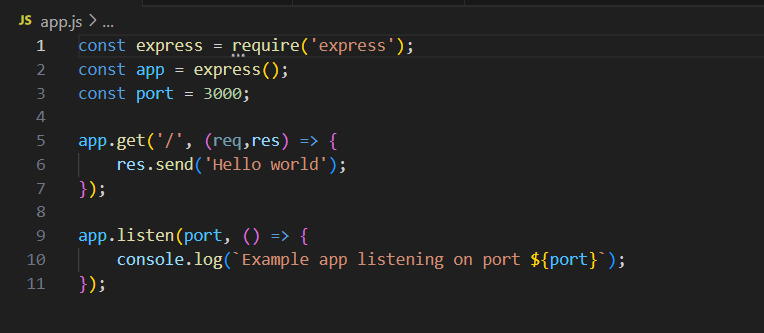
1. Инициализация файла ORM, который в будущем будет содержать набор моделей-таблиц базы данных



1. Настройка переменных окружения



**Простое приложение для проверки работы сервера:**



Запуск приложения

9

Страница в браузере

