**Actividad Práctica: Creación de una API de CRUD con Node.js y MariaDB, y desarrollo de un Front-End utilizando Docker**

**Criterios de evaluación para la actividad práctica:**

* *Documentación en archivo de Word y capturas de pantalla: Se requerirá que el estudiante realice todo el trabajo práctico dentro de un archivo de Word. Deberán incluir capturas de pantalla por cada punto completado, junto con una descripción clara y concisa de cada paso realizado. Se evaluará la correspondencia entre las capturas de pantalla y las acciones realizadas en cada punto del trabajo práctico.*
* *Se evaluará la prolijidad y coherencia en la organización del código, documentacion y en los archivos, así como el respeto de los datos utilizados, utilizando nombres descriptivos y siguiendo buenas prácticas de seguridad en la manipulación de la información.*

**Consignas:**

# Parte 1: Configuración del entorno y creación de la API

1. Configura un entorno de desarrollo Docker en tu máquina.
2. Crea un contenedor de Docker utilizando la imagen de MariaDB.
3. Crea una API básica de CRUD utilizando Node.js y Express.
4. Configura una conexión a la base de datos MariaDB desde la API.
5. Implementa las siguientes rutas en tu API:
   * Ruta GET: /productos - Esta ruta debe devolver todos los productos almacenados en la base de datos.
   * Ruta POST: /productos - Esta ruta debe permitir la creación de un nuevo producto en la base de datos. Los productos deben incluir un nombre y un precio.
6. Crea un archivo “Dockerfile” y copia los archivos de la API dentro del contenedor.

# Parte 2: Configuración de los servicios

1. Crear un archivo docker-compose.yml y configurar los distintos servicios creados anteriormente.

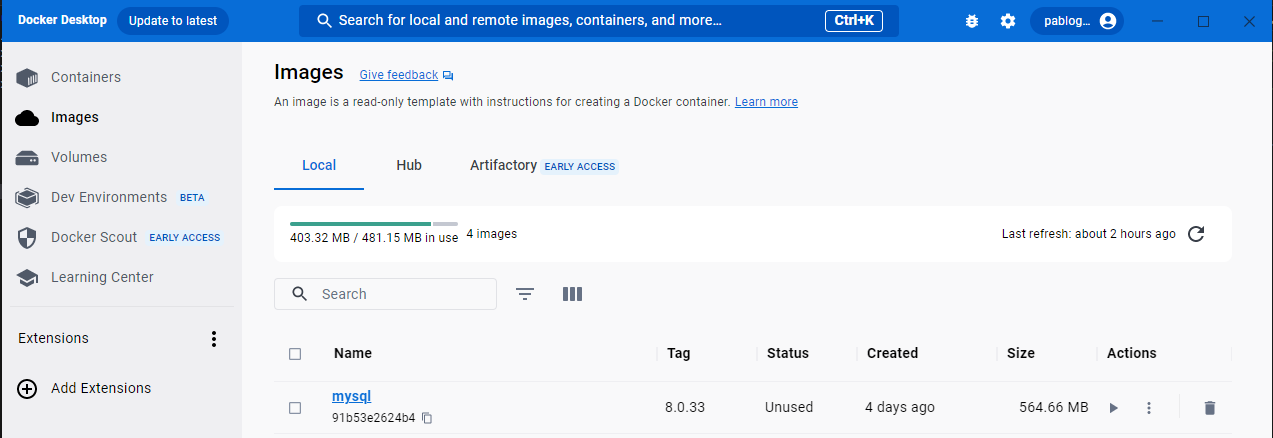
Parte 4: Pruebas:

1. Verifica que la conexión entre la API y la base de datos funciona correctamente. Asegúrate de que los productos se guarden correctamente en la base de datos.

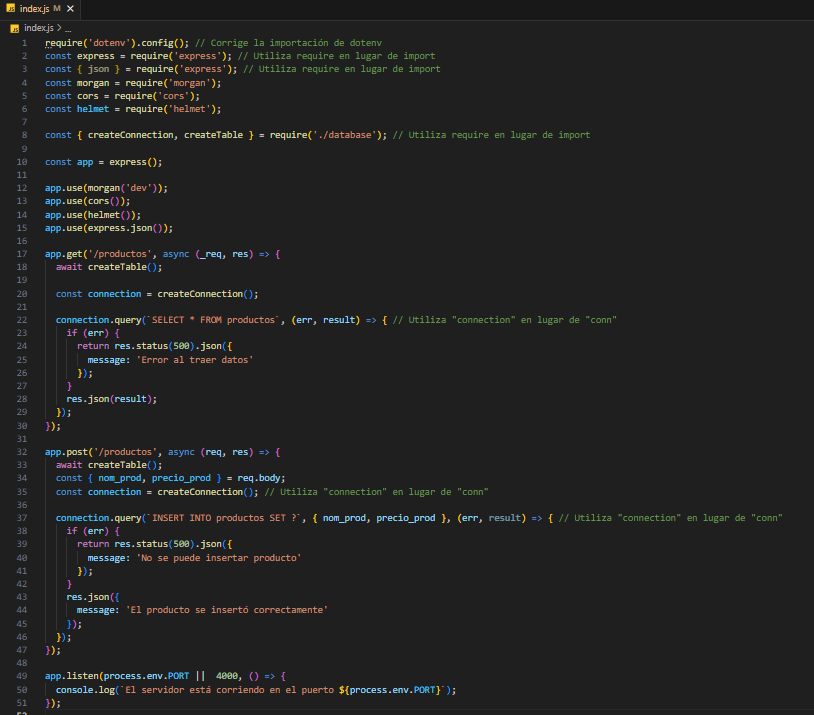
**Desarrollo**

Parte 1:

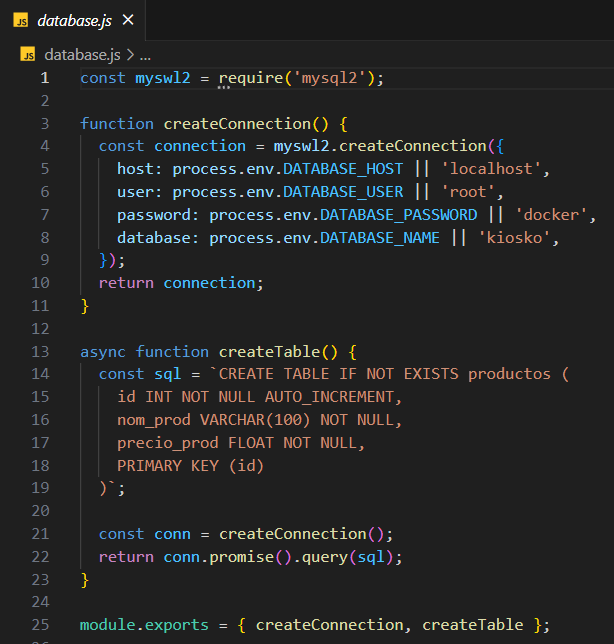
1. Ya creé mi entorno de Docker en el practico anterior.
2. Creé un contenedor Docker utilizando **mysql** con la versión 8.0.33.



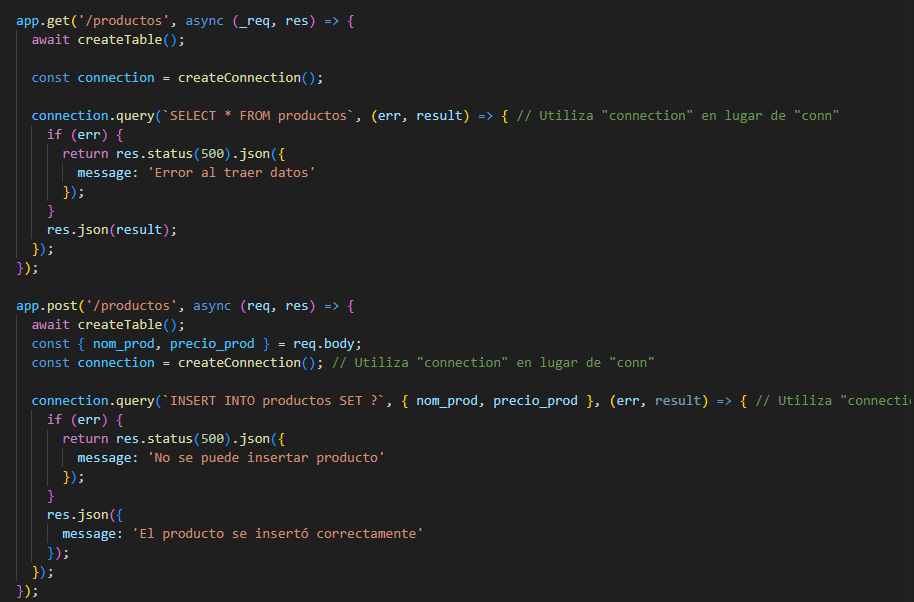
1. Hice un servidor API REST con Node.js y Express.



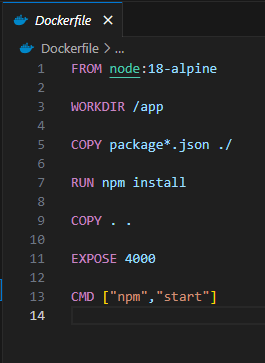
1. Configuro la conexión a una base de datos en mysql.



1. Implemento la ruta Get/productos y la ruta POST/productos.

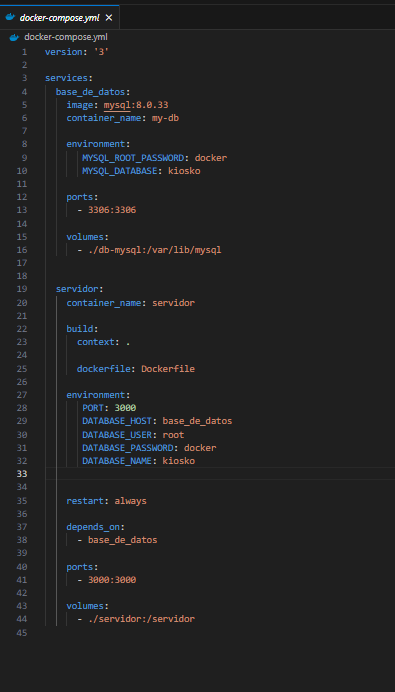


1. Creo el archivo Dockerfile.



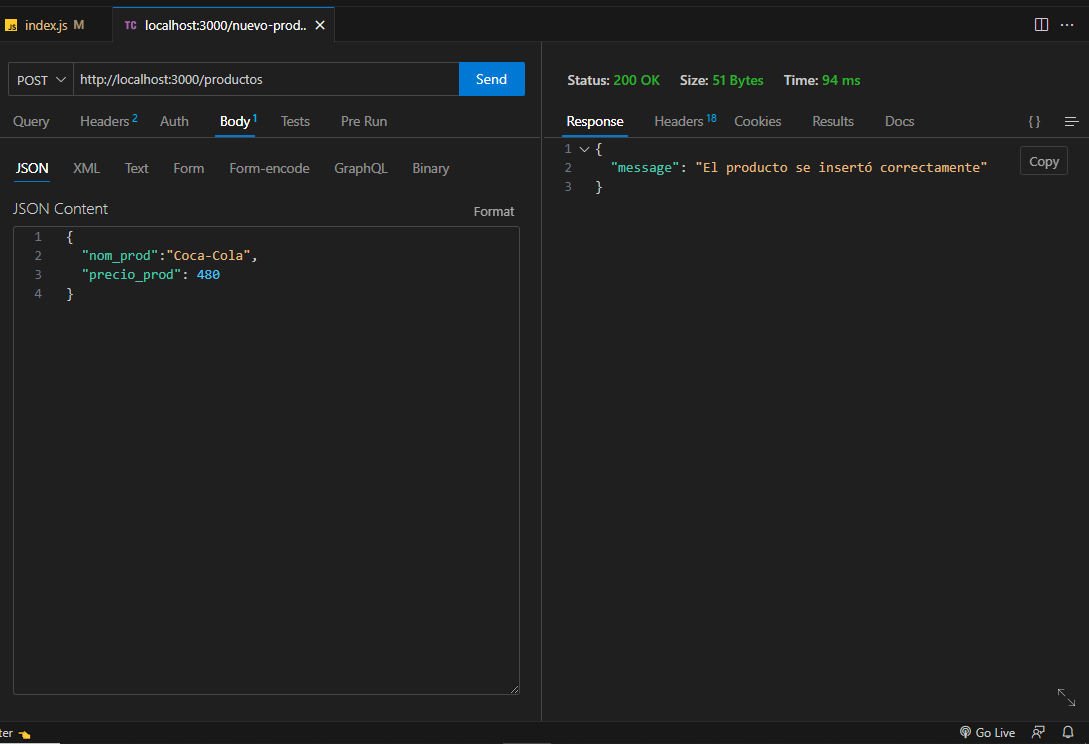
Parte 2:

1. Creo el archivo docker-compose.yml con toda la configuración del los servicios.

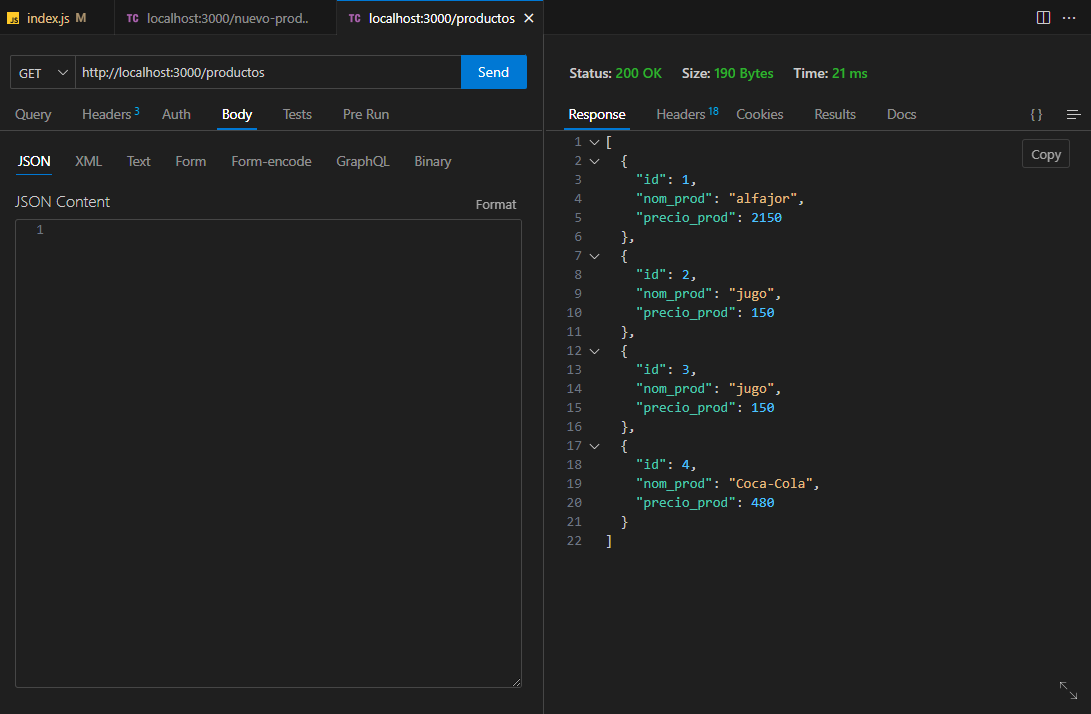


Parte 4:

1. Corroboro que funcione la ruta Post.



Corroboro la ruta Get.



Mi servidor y la base de datos corriendo en Docker.

