**Parcial Bases de Datos Api**

**Edwin Camilo Rodriguez Arredondo**

**ID:841424**

**Bases de Datos Masiva**

**6° Semestre Ingeniería de Sistemas**

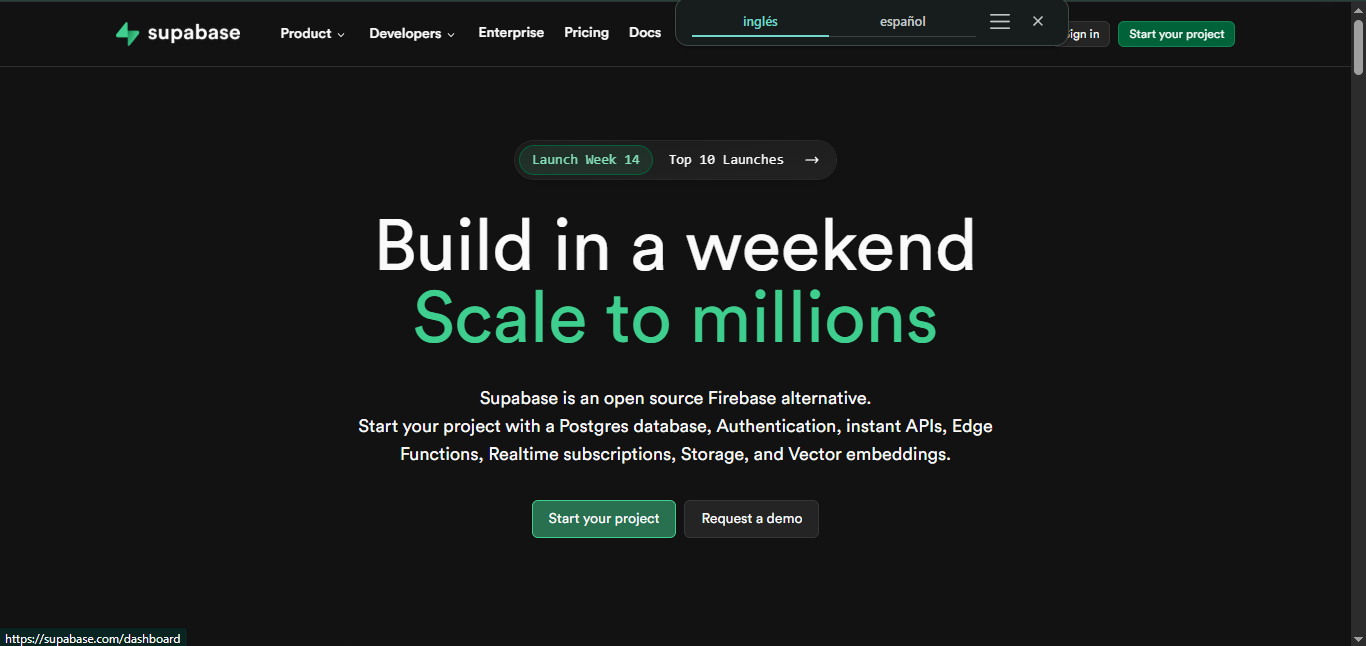
**Ingeniería de Sistemas**

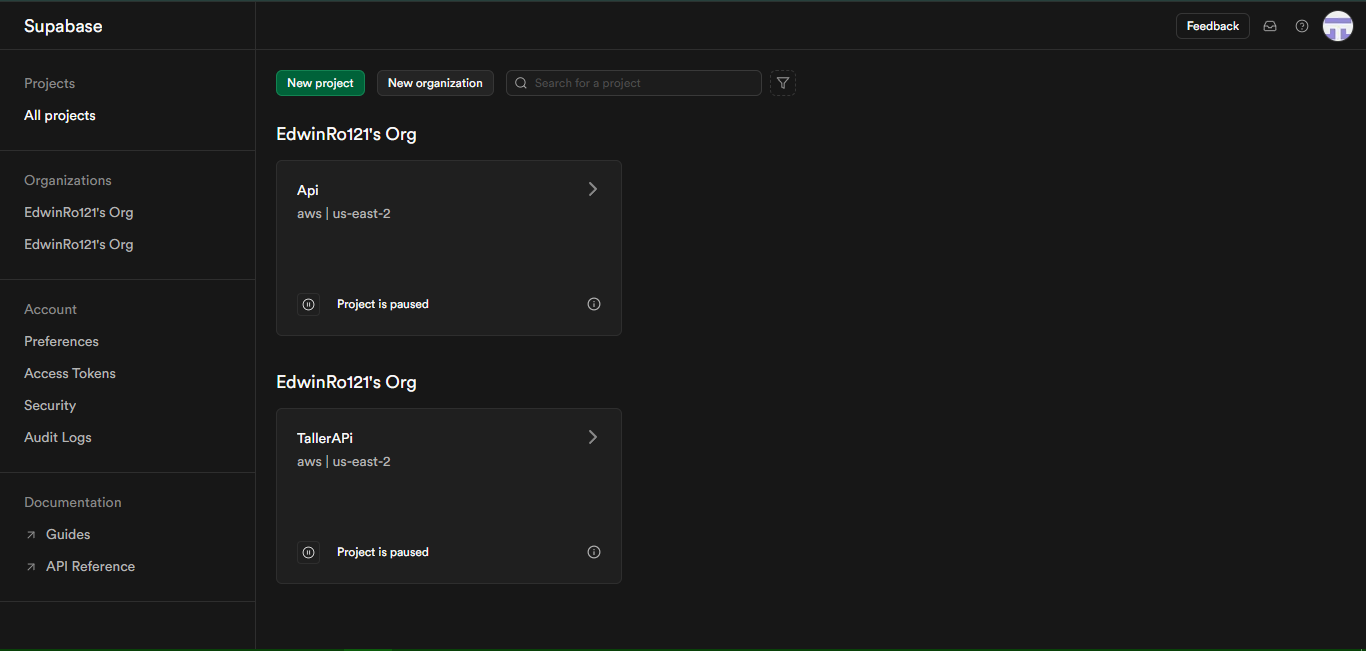
**Universidad UNIMINUTO**

**23 de abr. de 25**

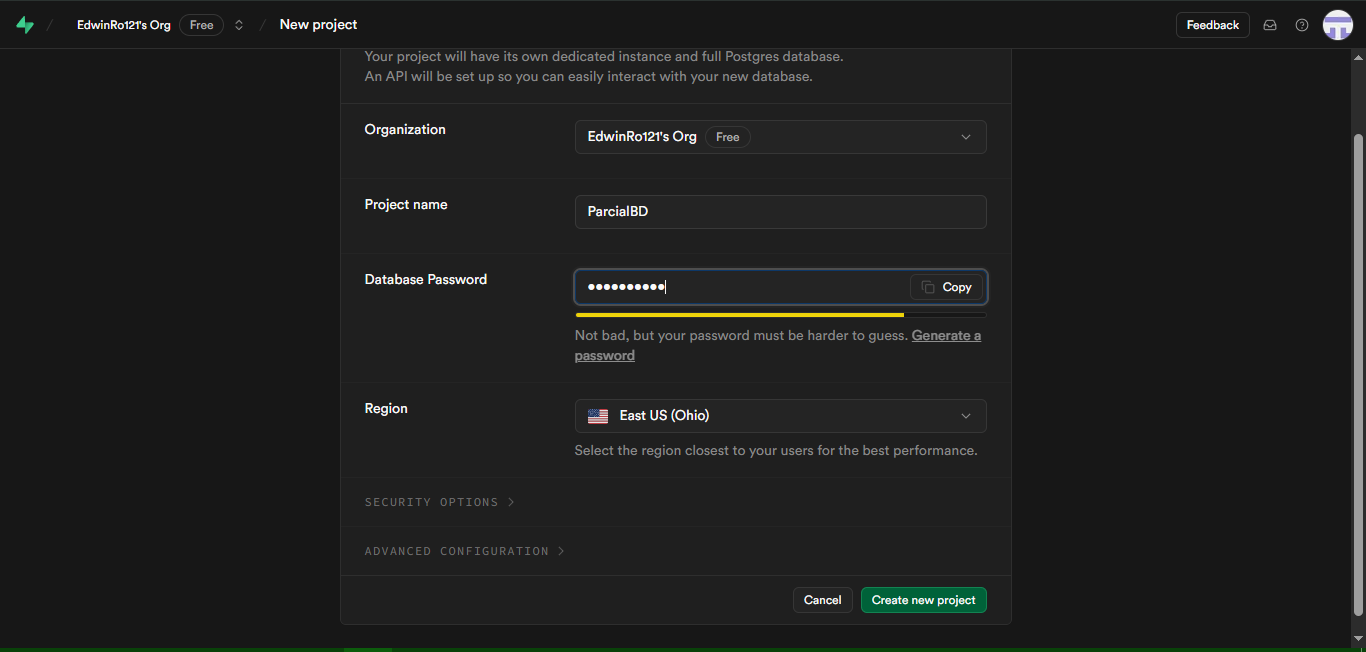
**Zipaquirá Cundinamarca**

* Se ingresa a supabase y se crar un nuevo proyecto

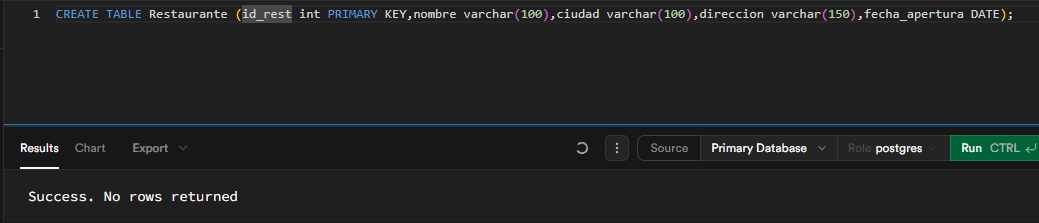


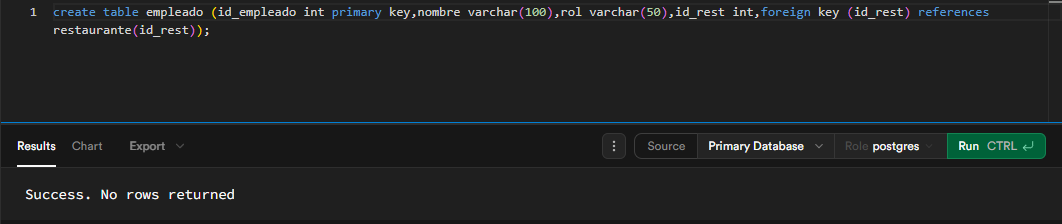


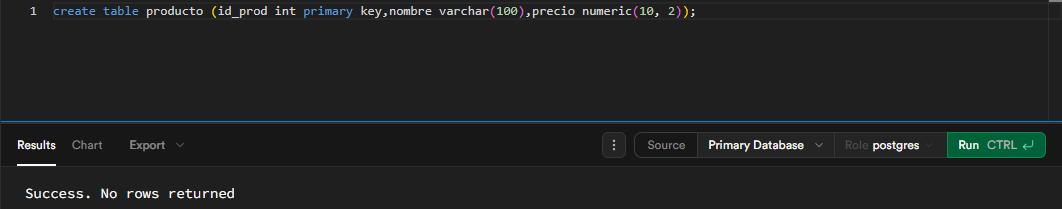
* Se asigna el nombre y contraseña

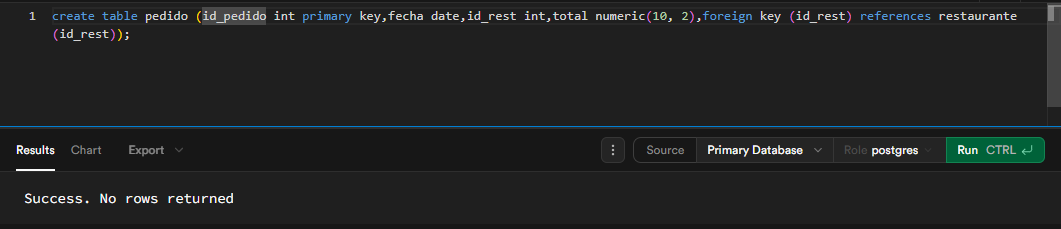


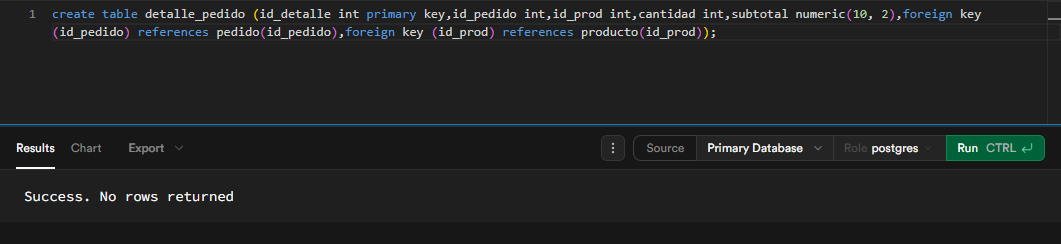
* Se crean las tablas con sus respectivos parámetros y sus respectivas relaciones



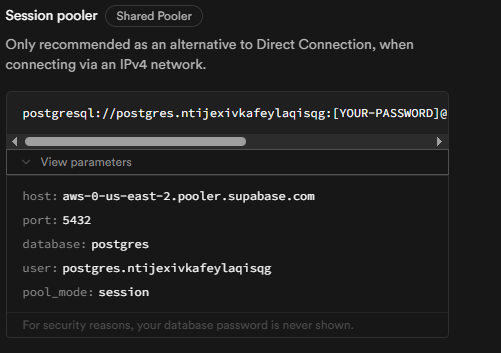




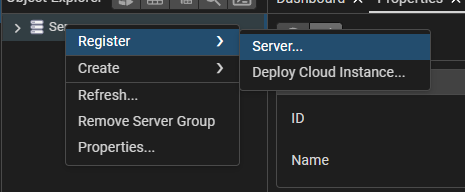




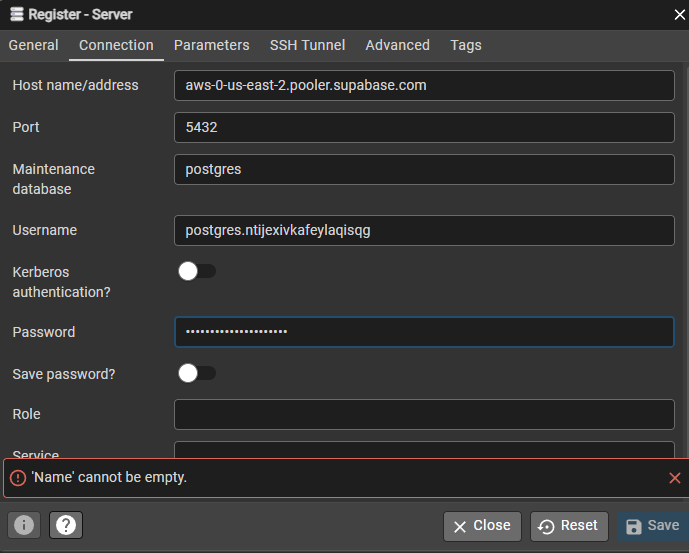
* Se miran los parámetros de la conexión

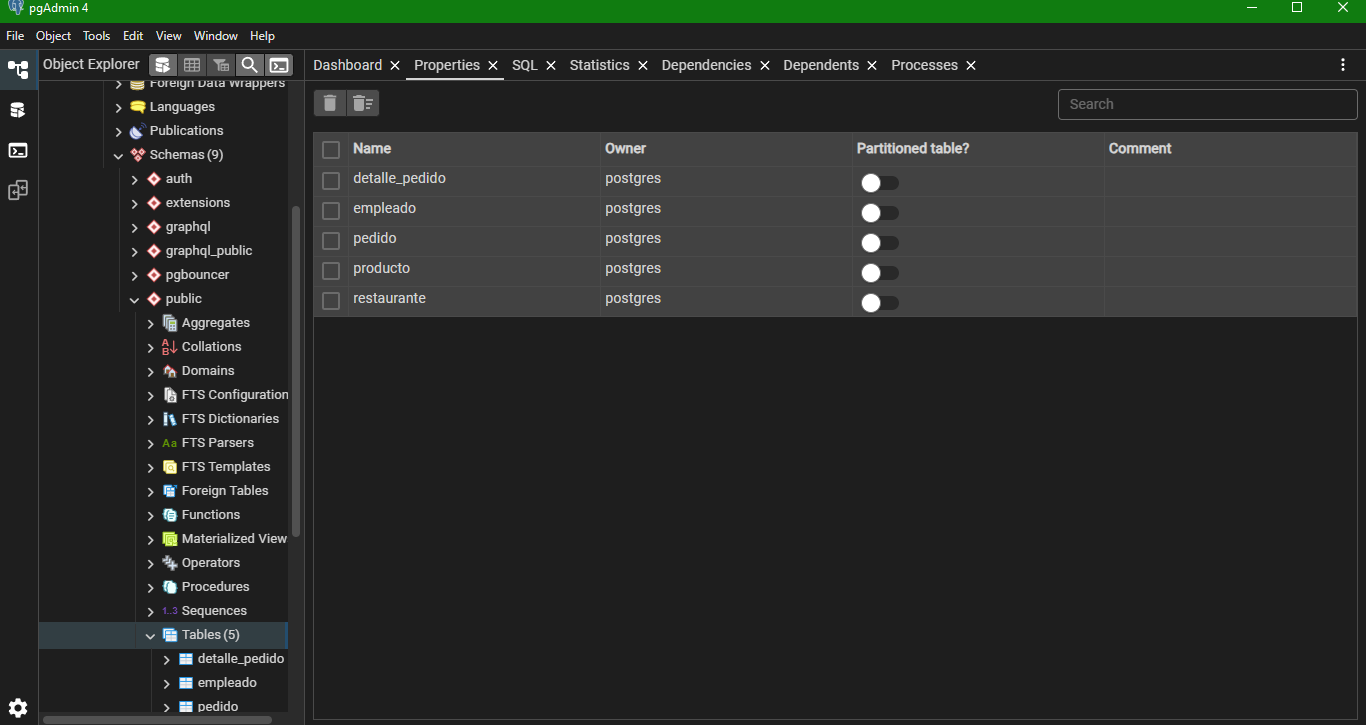


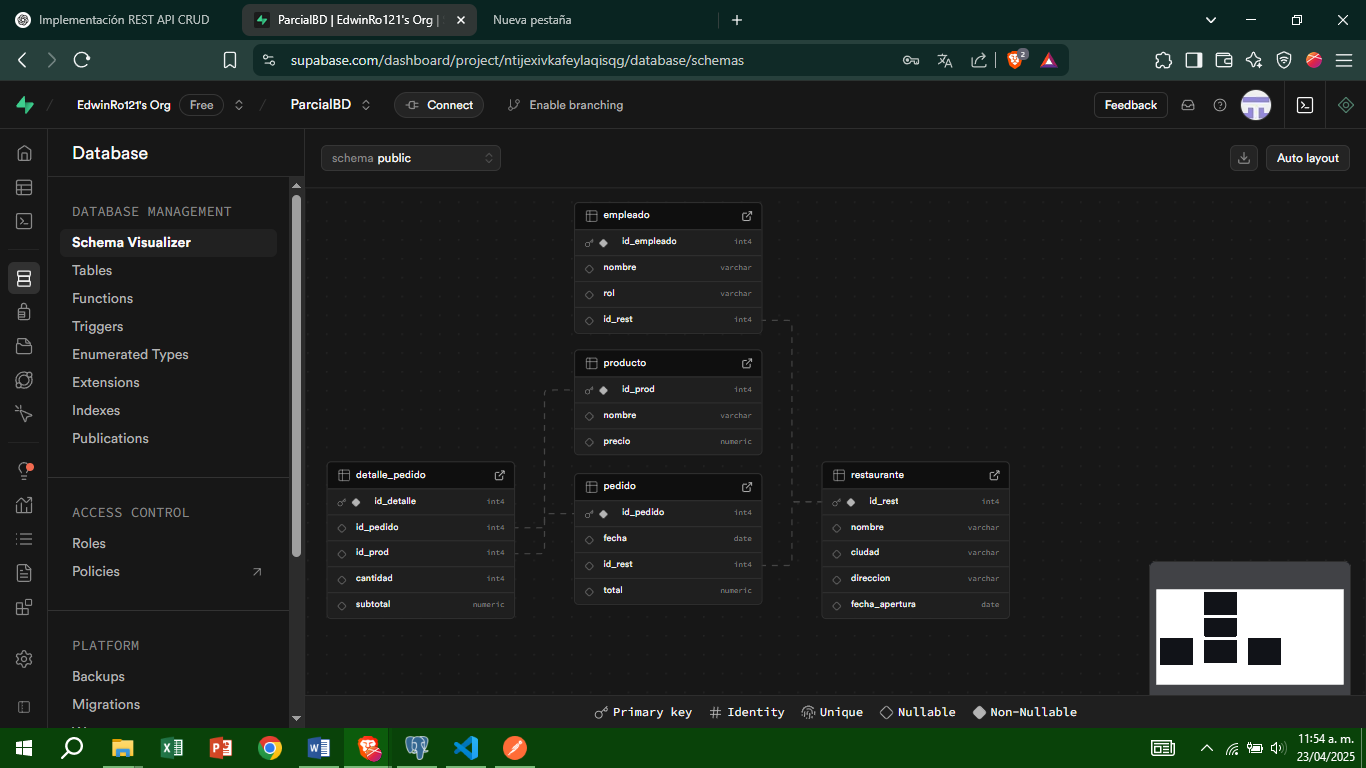
* Se crea un nuevo servidor en pgadmin4



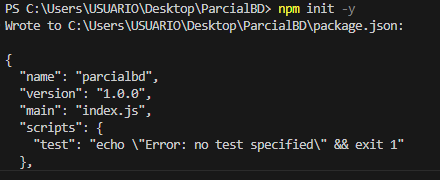
* Se insertan los parámetros de conexión de supabase



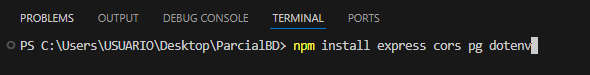




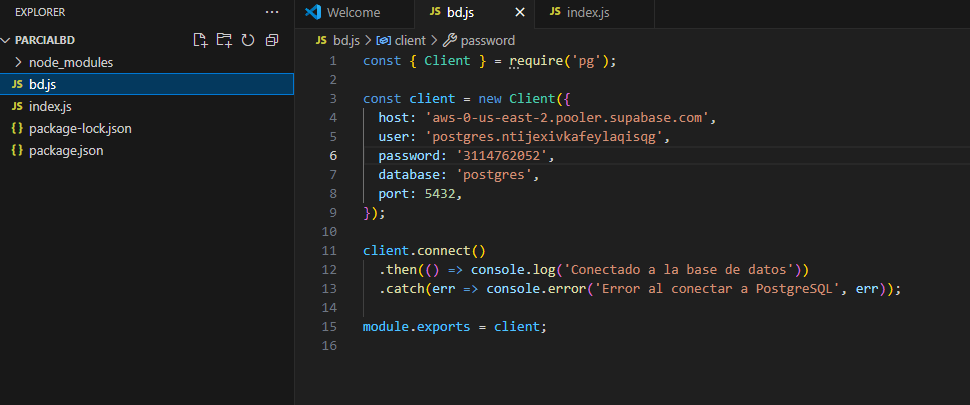
* Se abre el visual studio y se asigna una carpeta después se inicia un repositorio



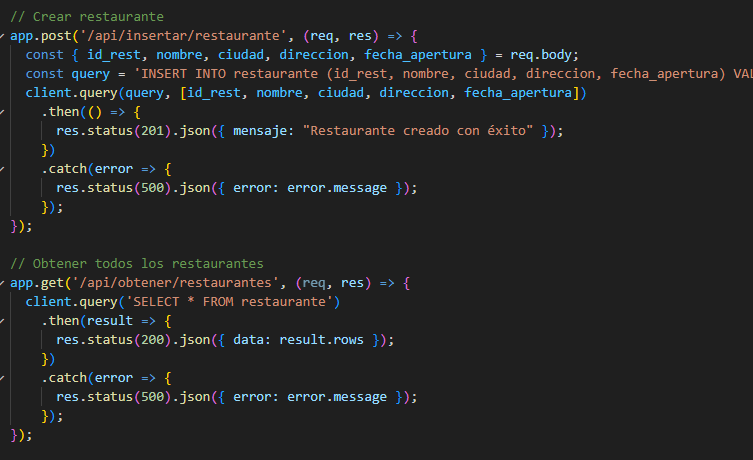
* Se instalan las respectivas las respectivas herramientas que usaremos



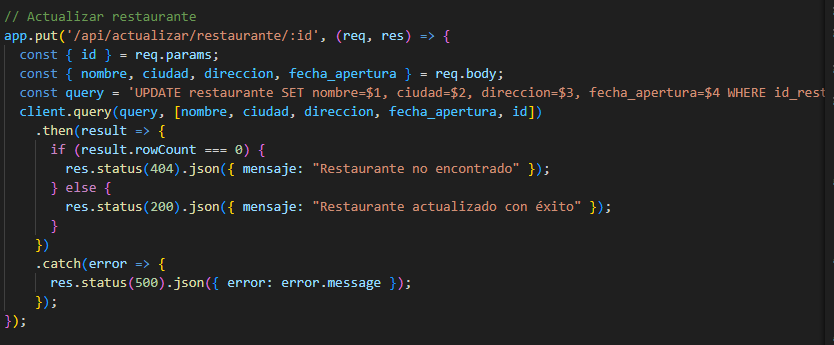
* Se crea un nuevo archivo y se crea la conexión con supabase



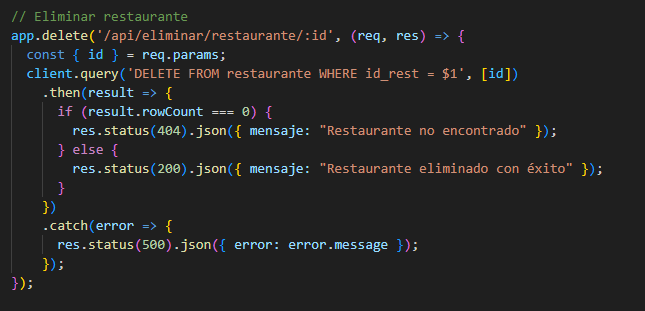
* Se crea otro archivo llamado index.js y se crea las api de cada tabla



* app.post() crea una ruta POST.
* req.body obtiene los datos enviados por el cliente (en JSON).
* Se arma una consulta SQL con INSERT INTO para guardar el restaurante.
* Se usa client.query() (desde tu conexión a PostgreSQL) para ejecutar la consulta.
* Si todo va bien, responde con 201 (Created).
* Si falla, responde con 500 (Internal Server Error).
* GET para consultar todos los restaurantes.
* Ejecuta SELECT \* FROM restaurante para obtener todos los registros.
* result.rows es un array con todos los resultados.
* Responde con un objeto JSON que contiene la información



* Ruta PUT para actualizar un restaurante.
* Usa :id como parámetro en la URL req.params.id
* Actualiza los campos especificados en el body.
* rowCount verifica si se actualizó al menos un registro. Si no, muestra 404.



* Ruta DELETE para eliminar un restaurante por ID.
* Ejecuta DELETE FROM restaurante WHERE id\_rest = $1 ($1 significa el dato q se le ingrese por ejemplo 4 , Ejecuta DELETE FROM restaurante WHERE id\_rest = 4)
* Si no encuentra el restaurante, rowCount es 0, así que responde con 404