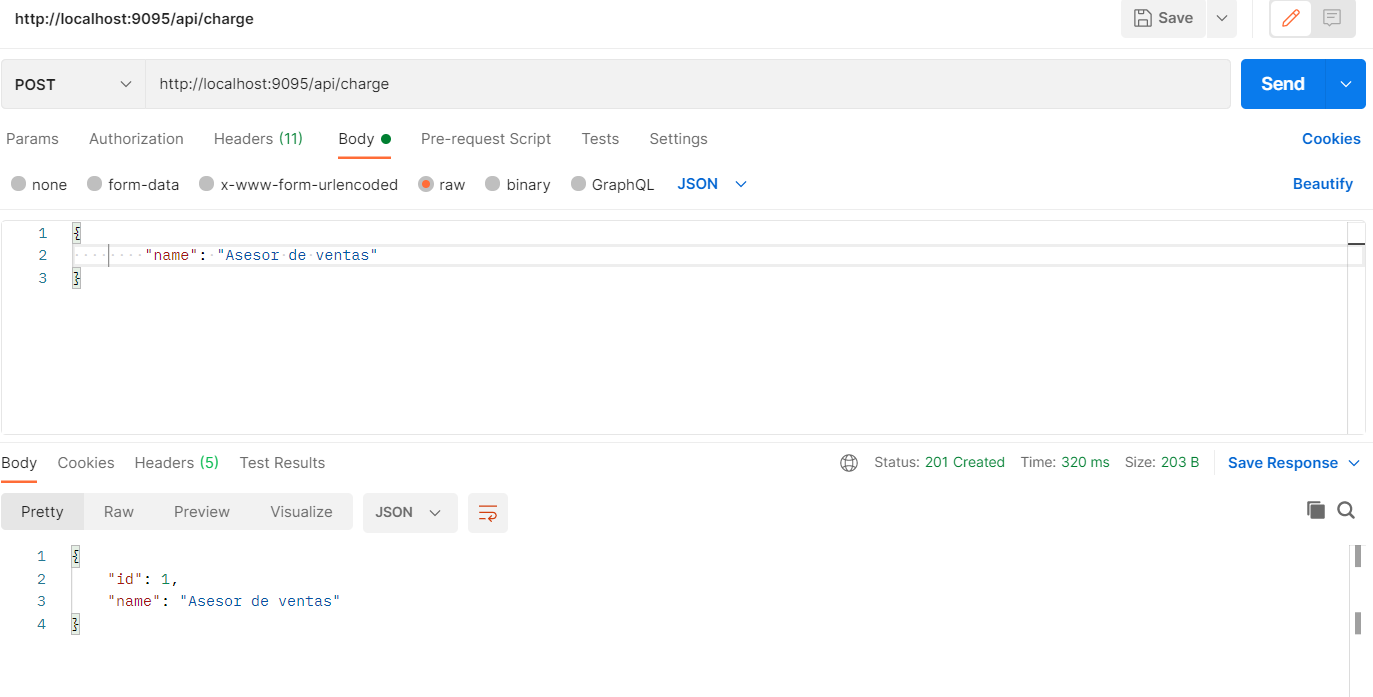
**Evidencia con ejecución de pruebas funcionales.**

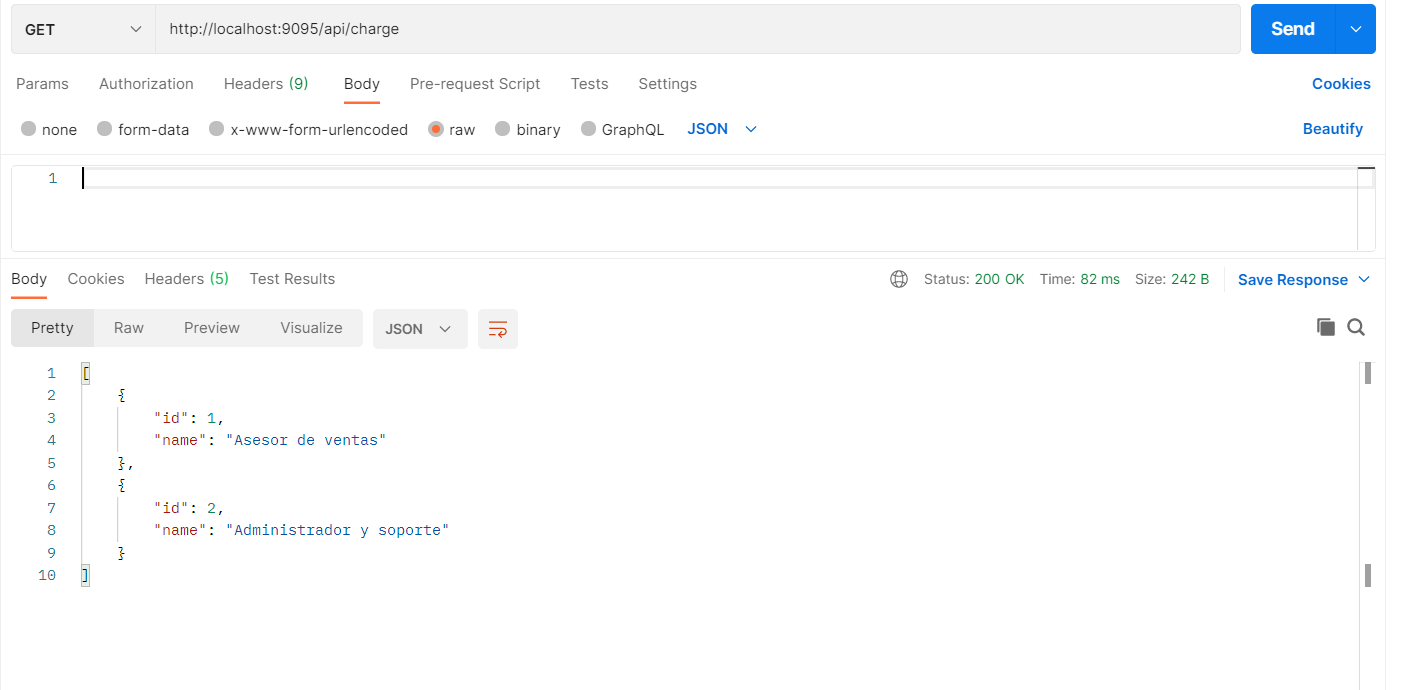
* **Registrar cargo**

**POST**: <http://localhost:9095/api/charge>



* **Listar cargos**

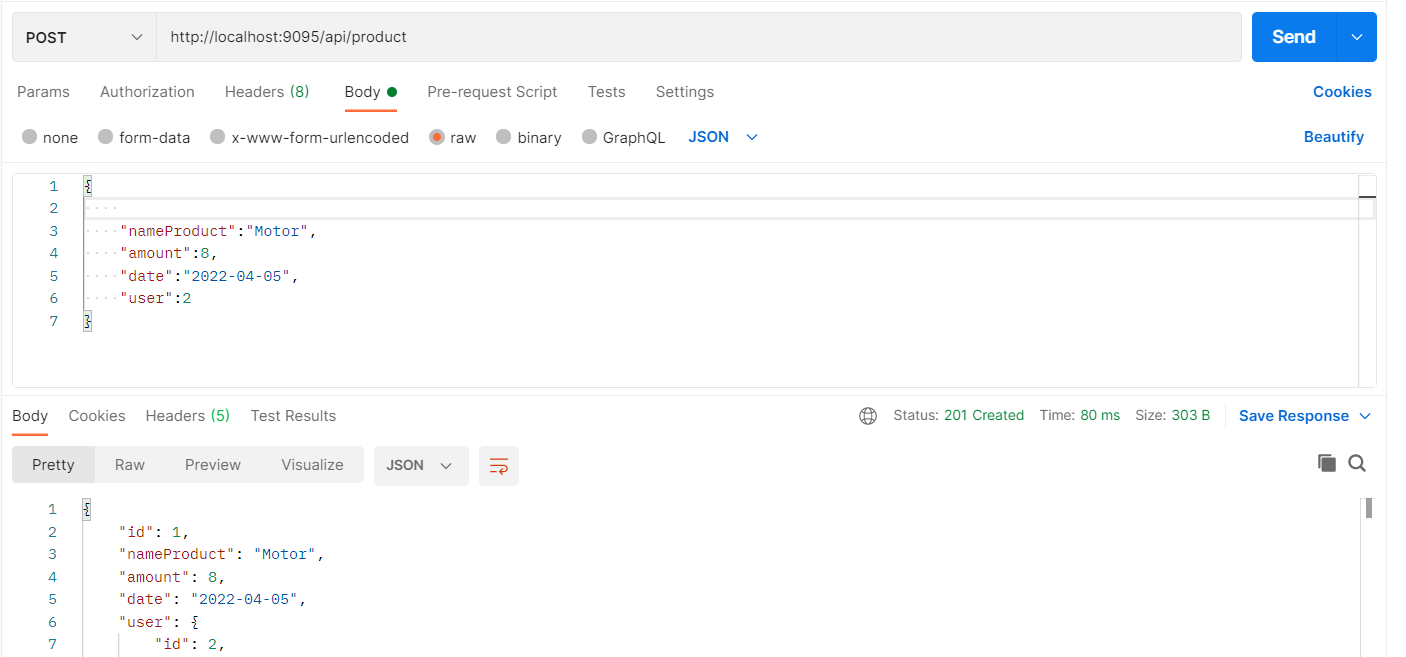
**GET:** <http://localhost:9095/api/charge>



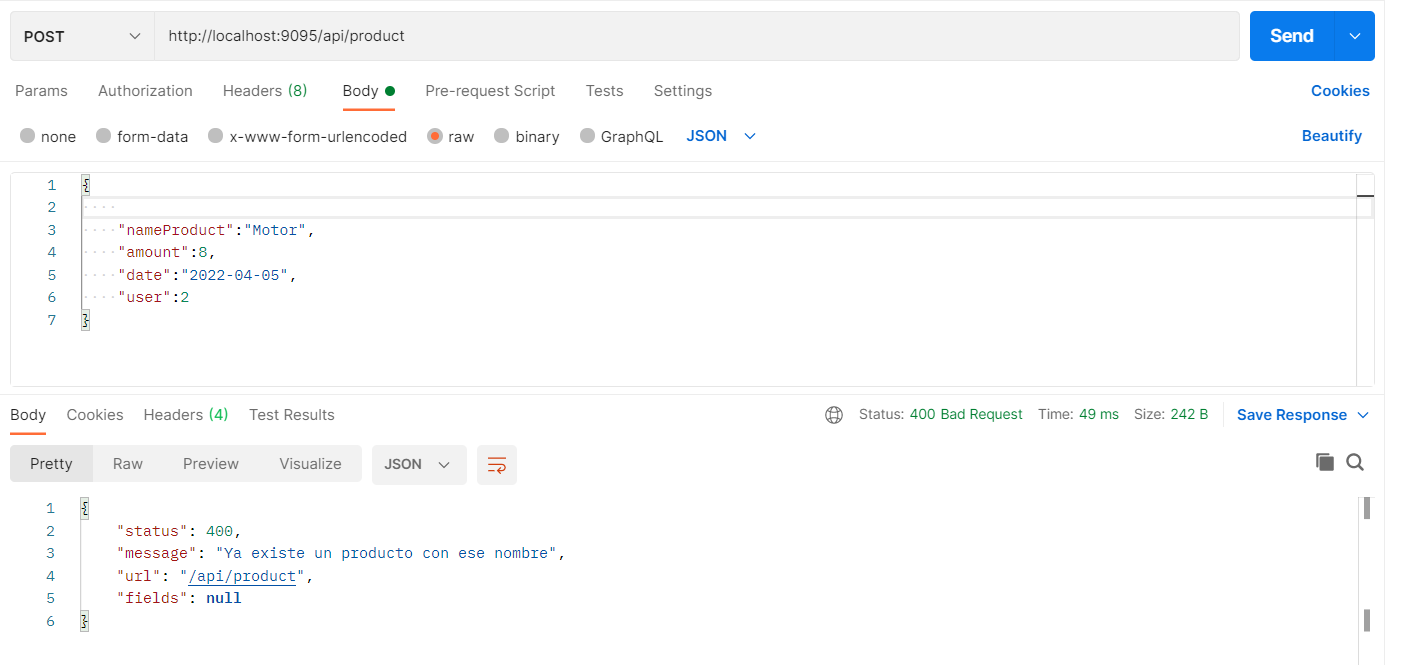
* **Registro de producto**

En este método se aplican diferentes validaciones.

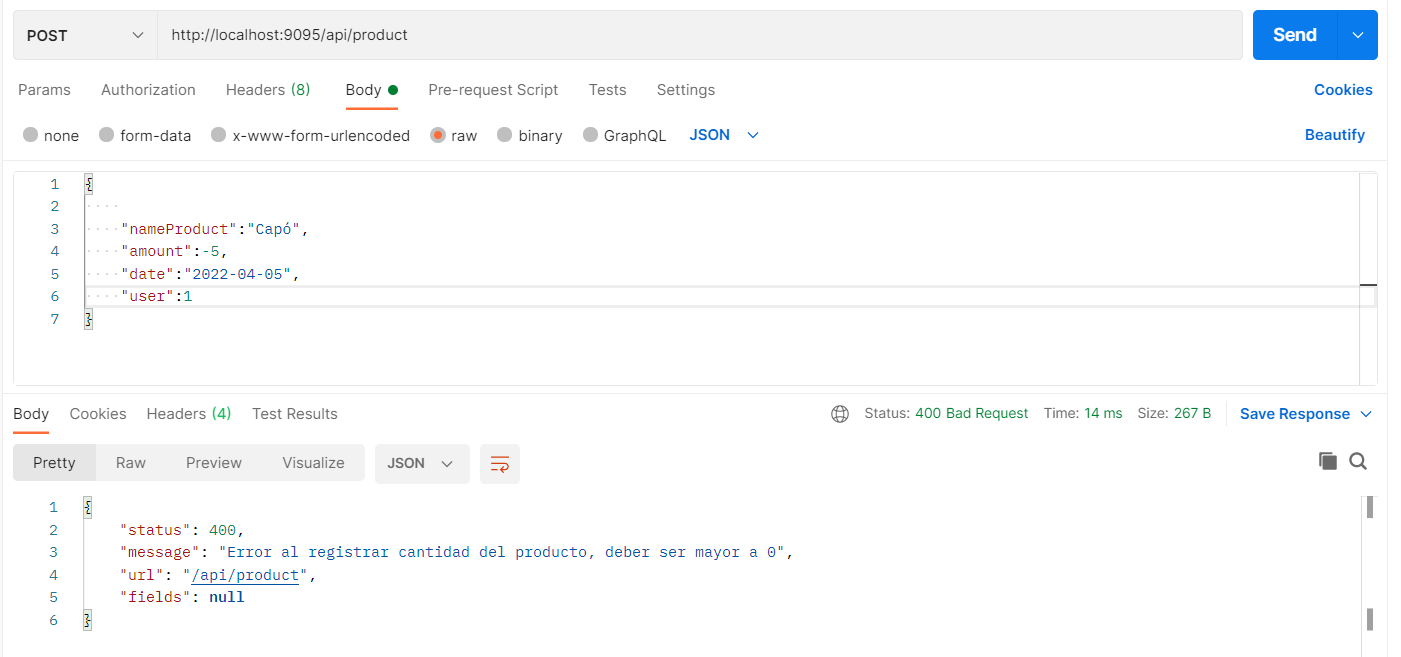
**POST:** <http://localhost:9095/api/product>



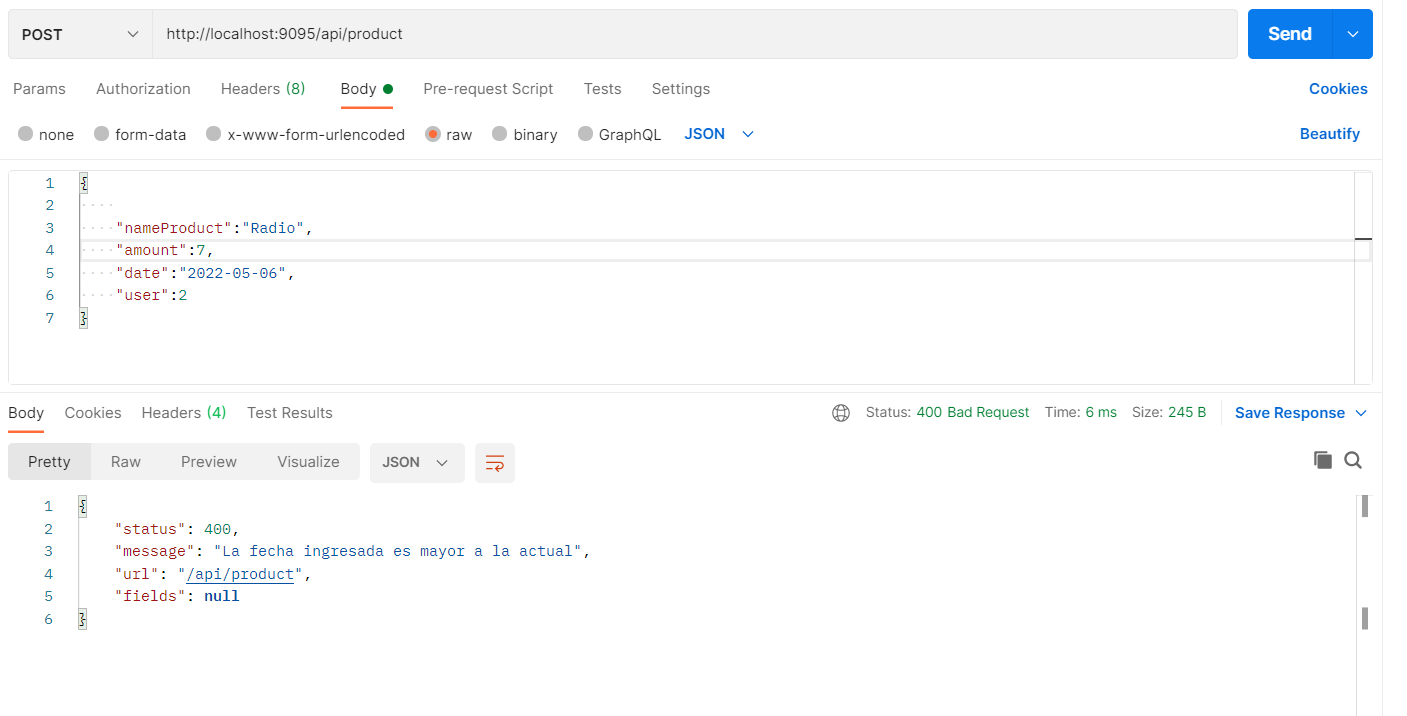
* Validación para que el nombre del producto no se repita.



* Validación para que la cantidad sea un numero entero.

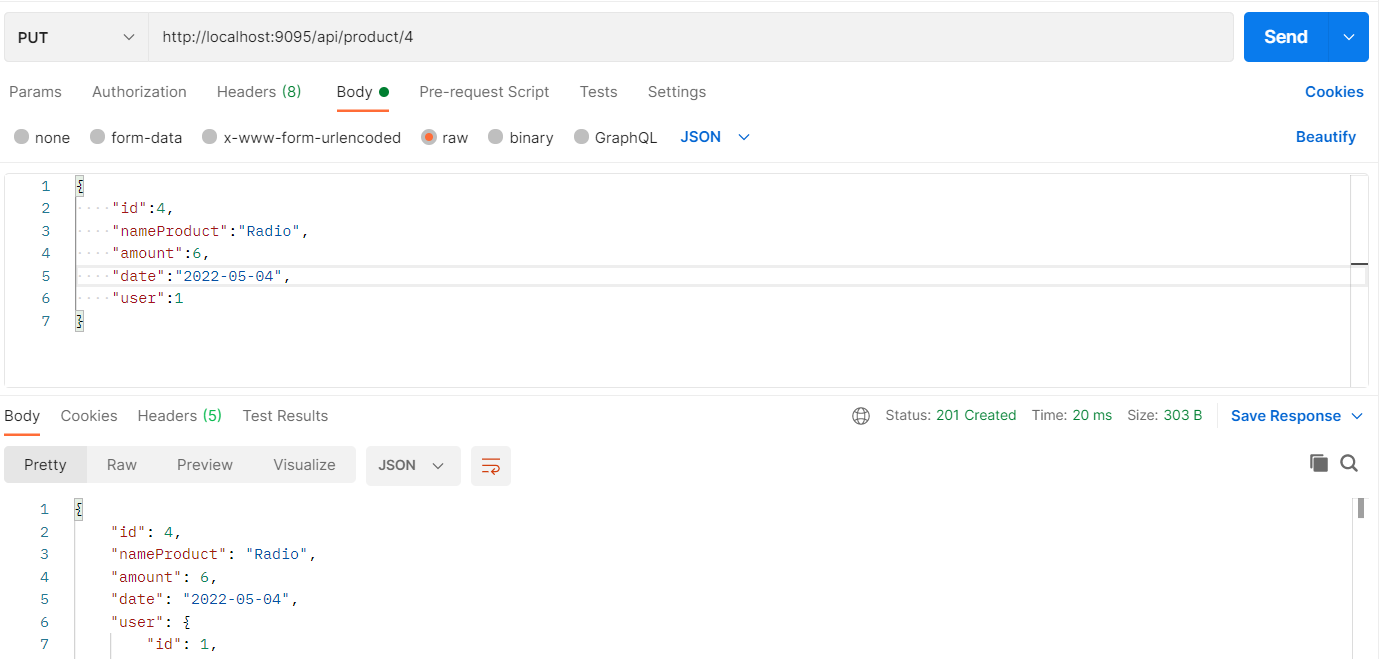


* Validación de la fecha de ingreso debe ser menor o igual a la fecha actual



* **Actualizar un producto.**

**PUT:** [http://localhost:9095/api/product/{id}](http://localhost:9095/api/product/%7bid%7d)



En este método actúa un disparador que en el momento de actualizar hace el registro en la tabla logs.

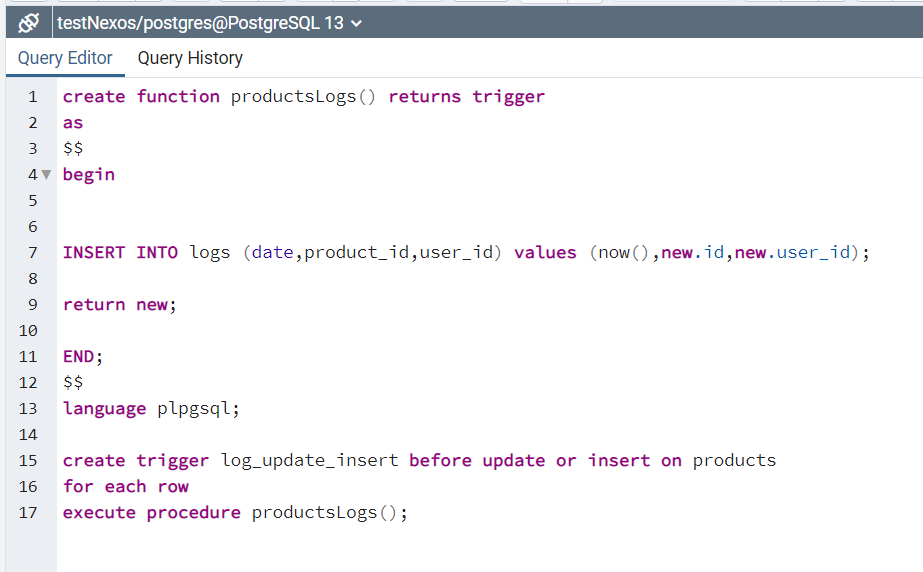
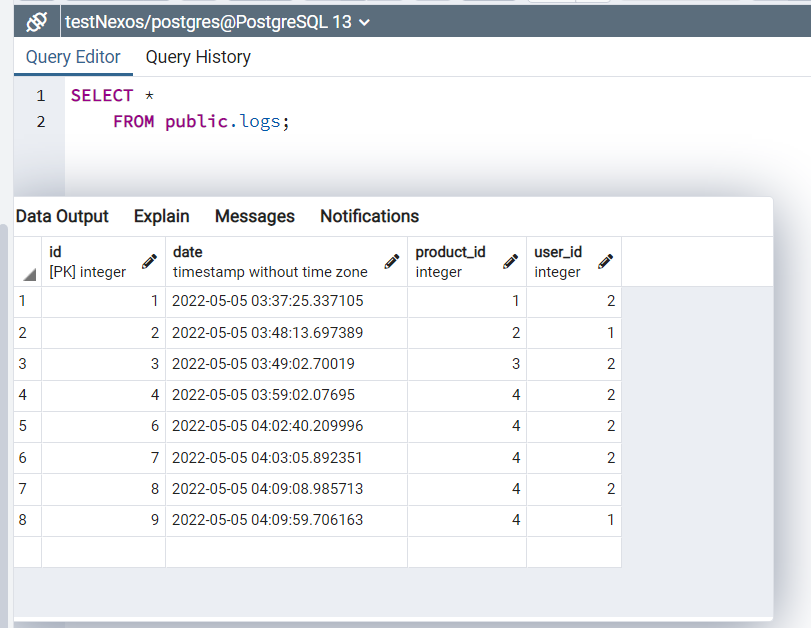


Tabla logs

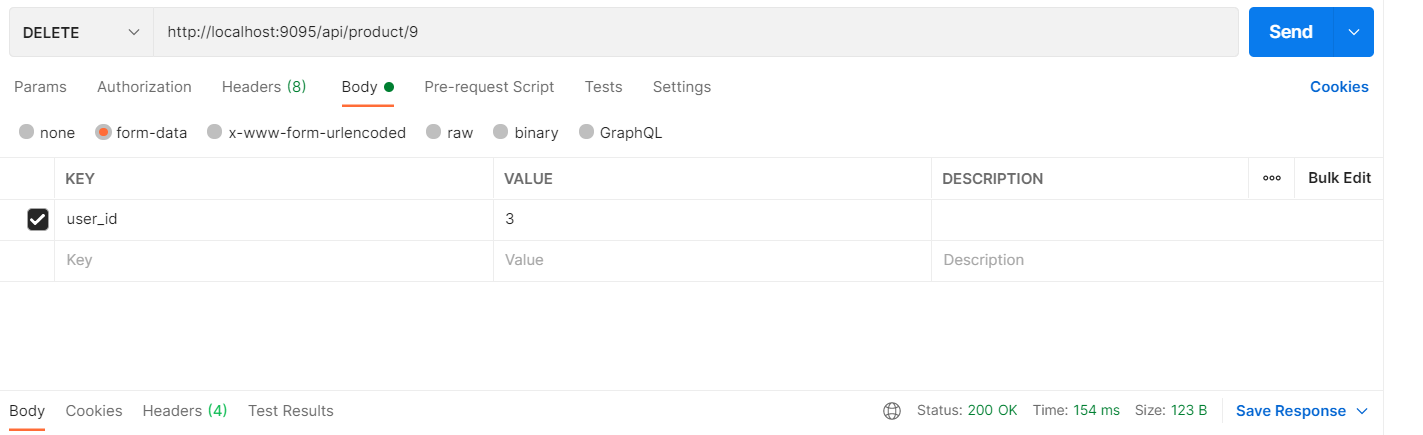


* **Eliminar un producto.**

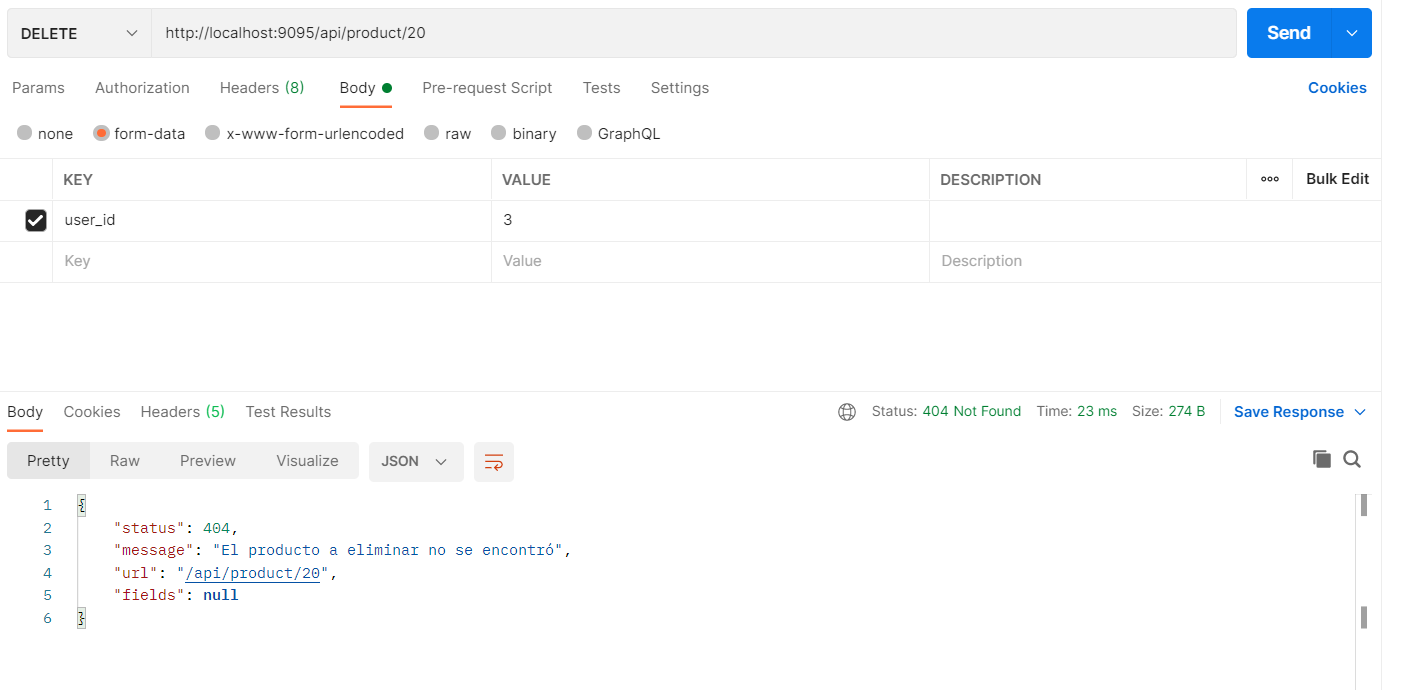
El producto solo puede ser eliminado por el usuario que lo registro.

Se manda el id de producto y el del usuario.

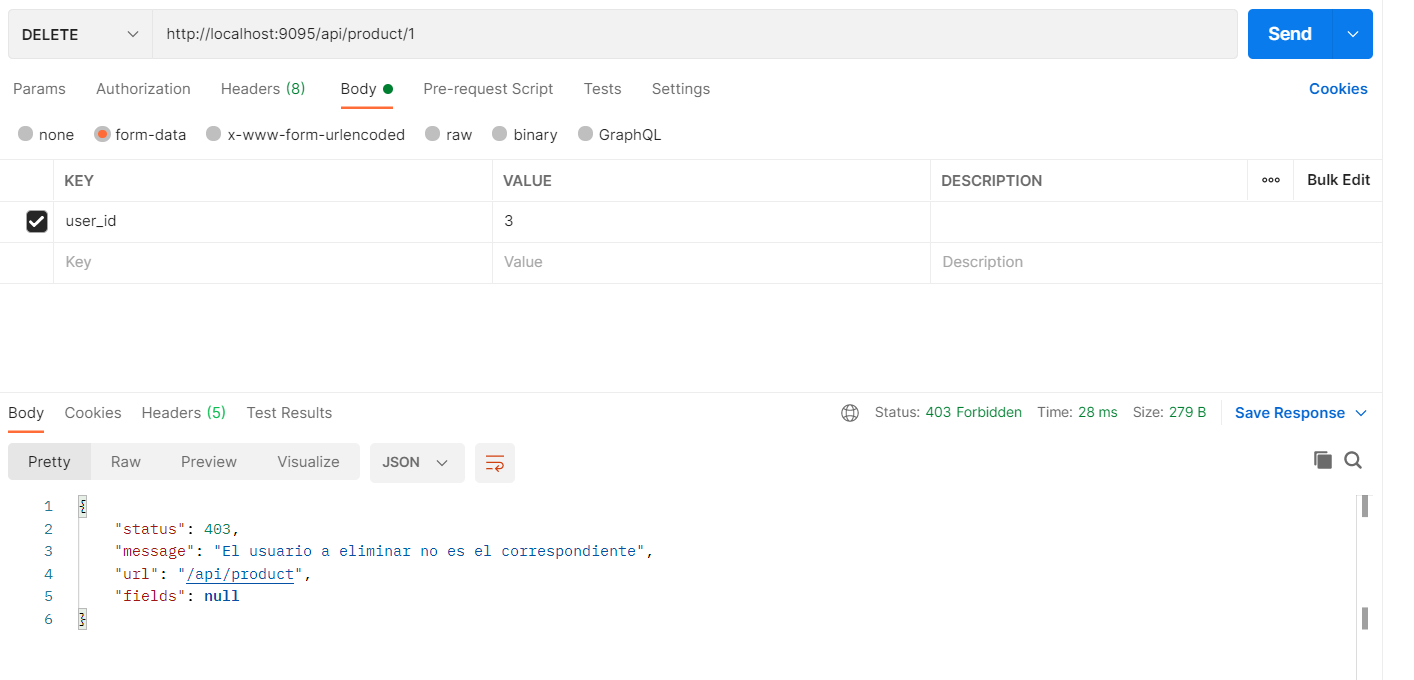
**DELETE:** [http://localhost:9095/api/product/{id}](http://localhost:9095/api/product/%7bid%7d)



También se hace la validación si el producto a eliminar existe.

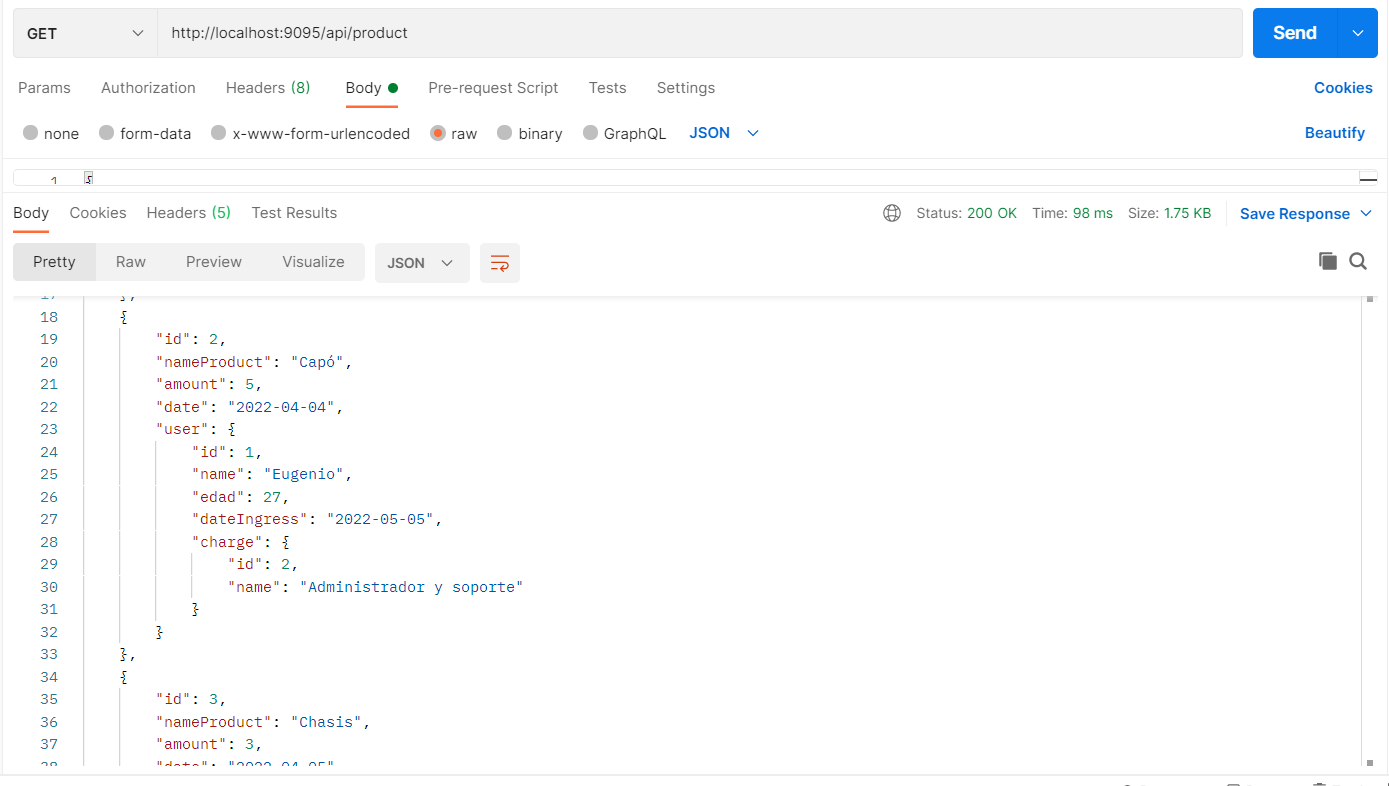


De la misma manera validamos si ese producto a eliminar le corresponde al usuario.



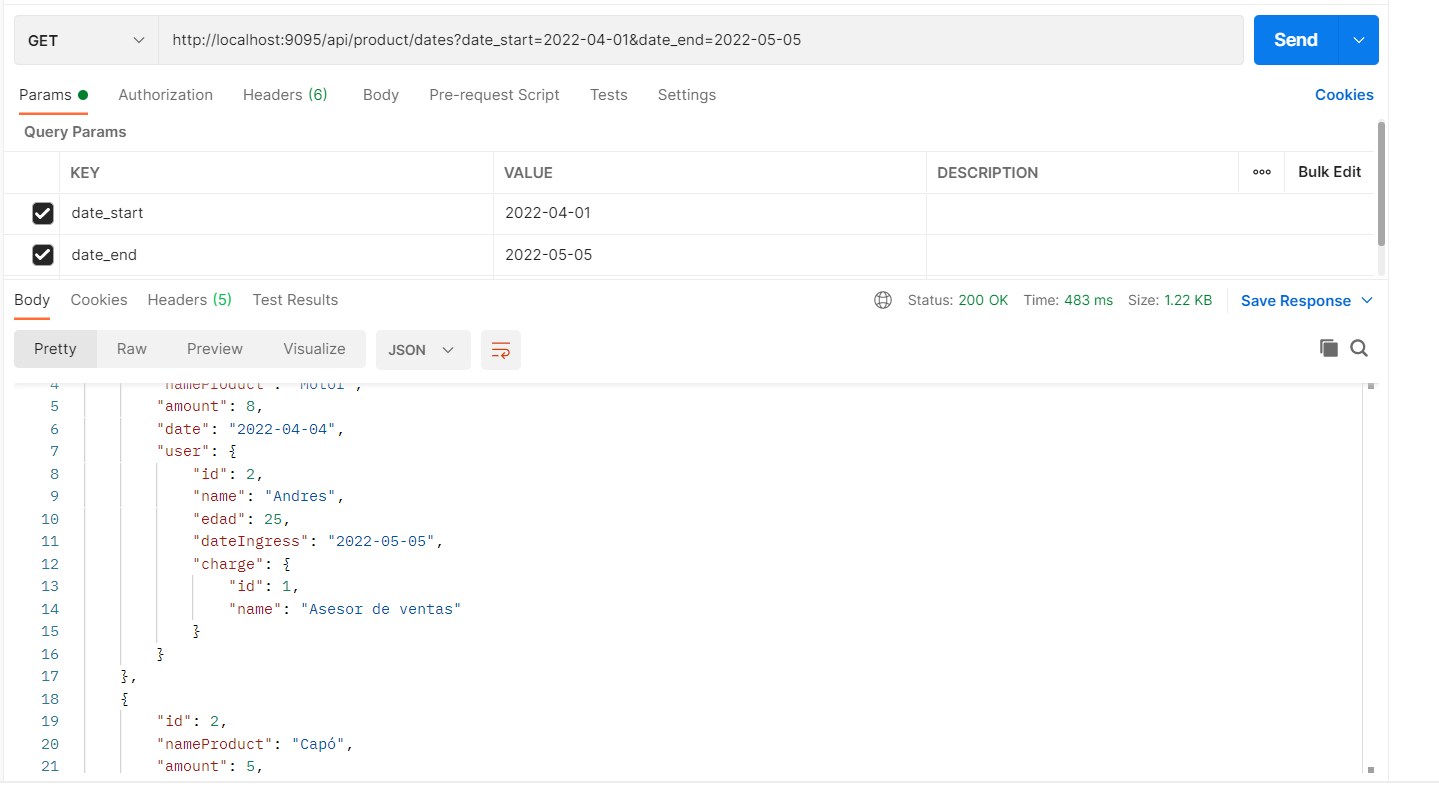
* **Listar productos**

**GET:** <http://localhost:9095/api/product>

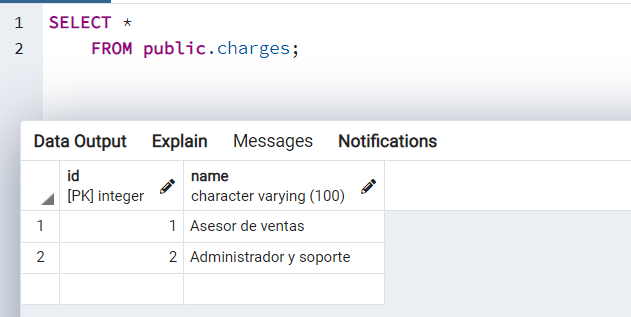


* **Filtro por rango de fecha.**

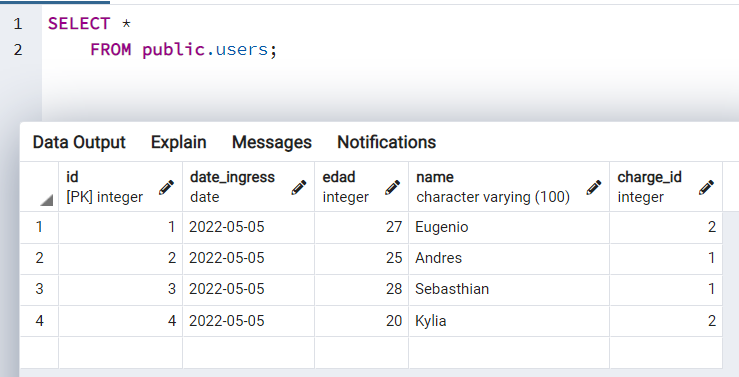
**GET:** <http://localhost:9095/api/product/dates>



**Tabla charges.**



**Tabla users.**



**Tabla products.**

