



Semana 3

Carlos López Lorenzo



Índice

Semana 3	1
1.Implementar endpoint para obtener productos por su id	2
2.Añadir validaciones en caso de errores	3
3.Prueba en postman	3



1.Implementar endpoint para obtener productos por su id

Crearemos un endpoint para obtener todos los datos de un personaje buscando por su id como el siguiente:

```
app.get("/personajes/:id", async (req, res) => {
  try {
    const personaje = await Personaje.findById(req.params.id);

    if (!personaje) {
      return res.status(404).json({ error: "Personaje no encontrado" });
    }

    res.json(personaje);
  } catch (error) {
    res.status(500).json({ error: "Error al obtener el personaje" });
  }
});
```

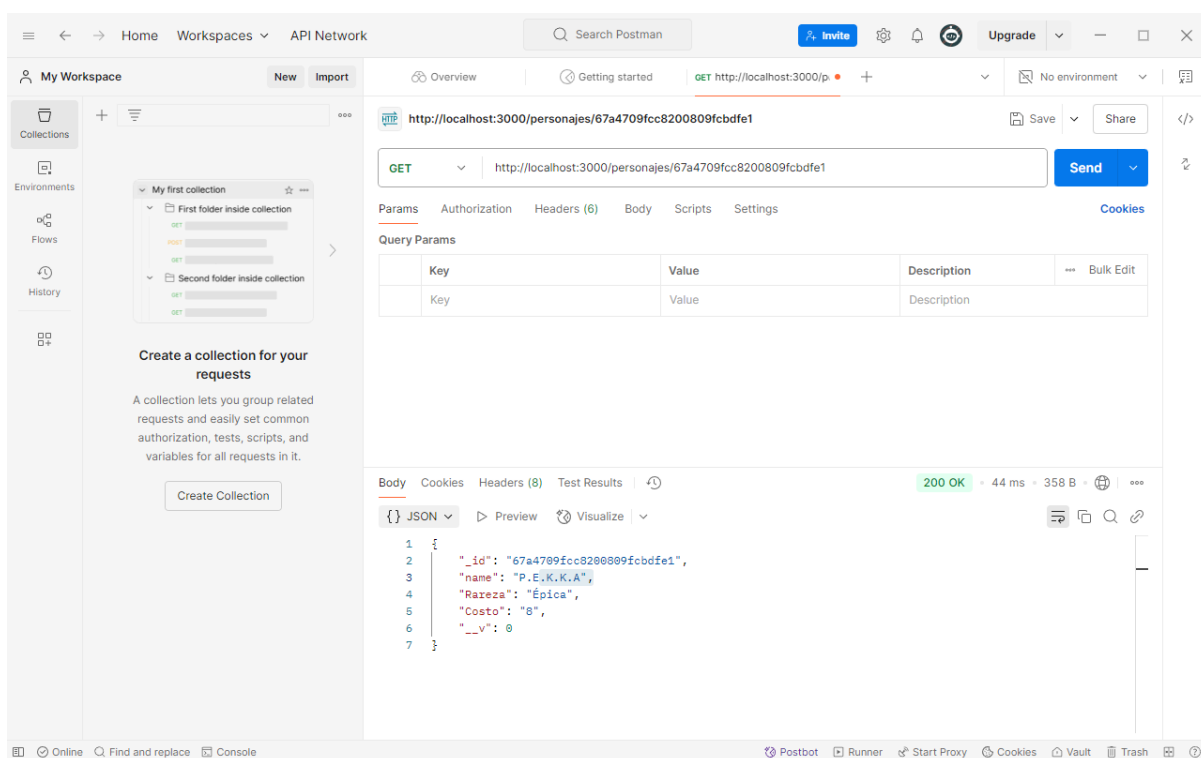
La función buscará el personaje en la base de datos en base al parámetro (id) que introducimos.

2.Añadir validaciones en caso de errores

Como podemos ver en la foto anterior he incluido un pedazo de código en caso de que al buscar esa id no se encuentre personaje muestre un mensaje de error , también hay un catch en caso de que la consulta no se haga correctamente o haya otro tipo de error lo cual mostrará otro mensaje diferente.

3.Prueba en postman

Ahora para probar el funcionamiento iremos a postman y añadiremos una id a nuestra sentencia que hemos usado anteriormente para obtener todos los personajes, como podemos ver aquí:



En la imagen podemos ver que se listan todos los datos guardados de este personaje con esta id.