Praktikum 9 - Matakuliah Pilihan 1 (Web)
Program Studi: Teknik Informatika

Lakukan praktikum dibawah ini, dan buat screenshot untuk pembuktian mengerjakan setiap poin dengan mengisi tabel dibawah, kemudian tunjukan hasil akhir dari men-share repository github yang telah dibuat.

A. Menyediakan API Enpoints

1. Lanjutkan Project Praktikum 8, dengan menggunakan file yang sama (copy)
2. Tambahkan pada database, sebuah tabel produk Isi tabel produk seperti pada tabel berikut: (Buat 10 item)

id	nama	deskripsi	harga	foto
1.	Indomie Goreng		2500	images/miegoreng.jpg
2.	…	…	…	…
10.	…	…	..	…

3. Seperti pada perintah praktikum 8 buatkan beberapa endpoints
   GET : localhost:8001/api/products/
   GET : localhost:8001/api/products/:id
   POST : localhost:8001/api/products/
   PUT : localhost:8001/api/products/:id
   DELETE: localhost:8001/api/products/:id

4. Pastikan memiliki file dengan struktur sebagai berikut.
   controllers/products.controller.js
   routes/products.routes.js
   models/products.model.js
5. Test API dengan menggunakan **POSTMAN**

B. Menambahkan Proteksi API (Simple - Bearer Method)

1. Buat sebuah folder bernama middlewares, kemudian buat didalamnya sebuah file dengan nama auth.middleware.js dengan kode program seperti dibawah ini

```
export const authBearer = (req, res, next) => {
    const authHeader = req.headers.authorization;

    // Tidak ada Authorization
    if (!authHeader) {
        return res.status(401).json({ message: "No authorization header" });
    }

    // Harus Bearer
    if (!authHeader.startsWith("Bearer ")) {
        return res.status(401).json({ message: "Bearer token required" });
    }

    // Ambil token
    const token = authHeader.split(" ")[1];

    // Token yang benar (misal hardcode)
    const VALID_TOKEN = "12345TOKENRAHASIA";

    if (token !== VALID_TOKEN) {
        return res.status(403).json({ message: "Invalid token" });
    }

    next();
};
```

2. Pada file product.routes.js panggil auth.middleware.js *import { authBearer } from "../middleware/auth.middleware.js"*

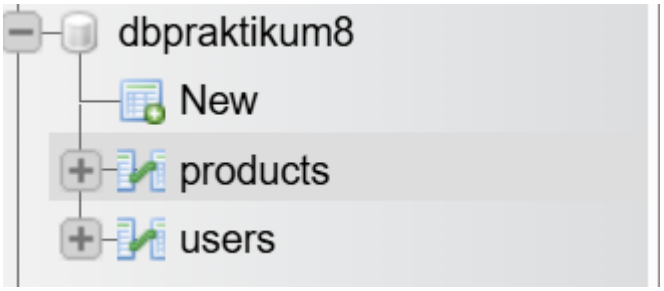
3. Lalu tambahkan authBearer ke endpoints yang perlu di proteksi router.post("/", authBearer, createProduct); router.put("/:id", authBearer, updateProduct); router.delete("/:id", authBearer, deleteProduct);

4. Gunakan POSTMAN untuk akses 3 endpoints ini dengan menambahkan bearer
   12345TOKENRAHASIA

F. Github + Visual Code

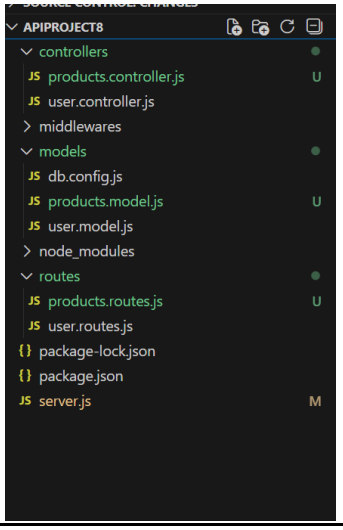
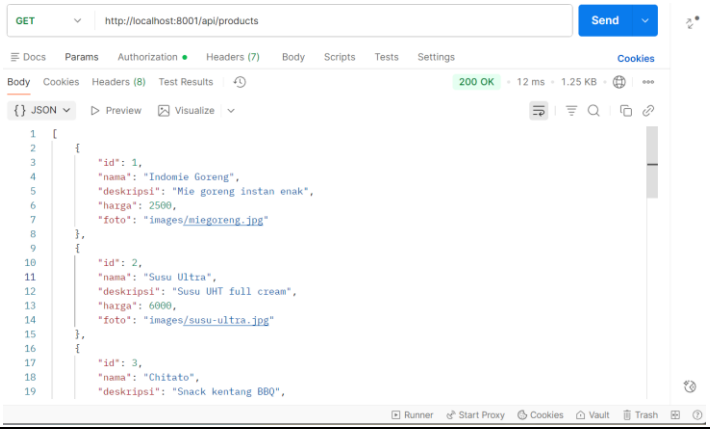
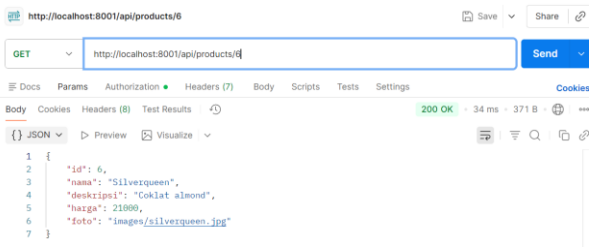
1. Buat proyek di Github dengan nama **Latihan9**
   git init
   git add
   .
   git commit -m "first commit" git branch -M main git remote add origin https://github.com/**agunghakase**/**Latihan9.git** git push -u origin main

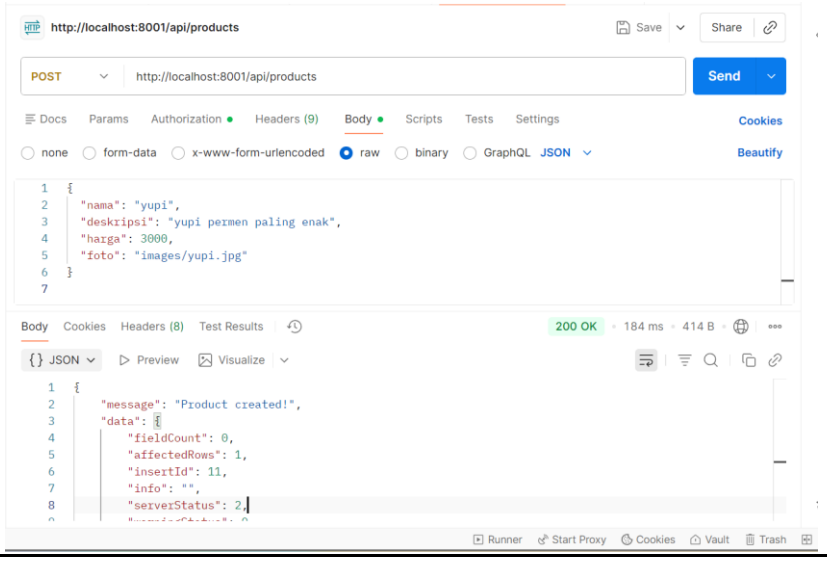
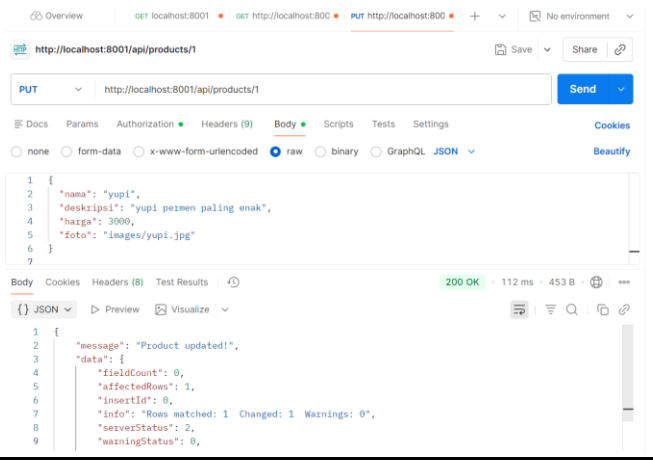
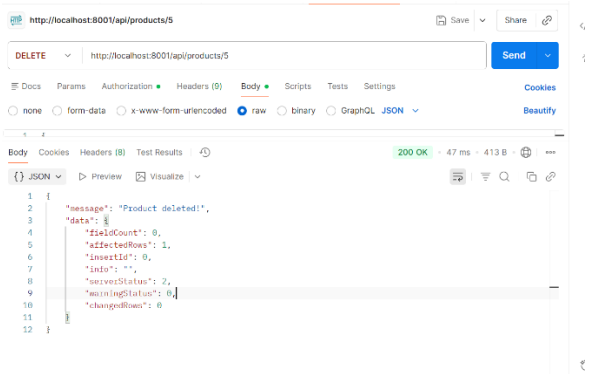
Hasil Pengerjaan

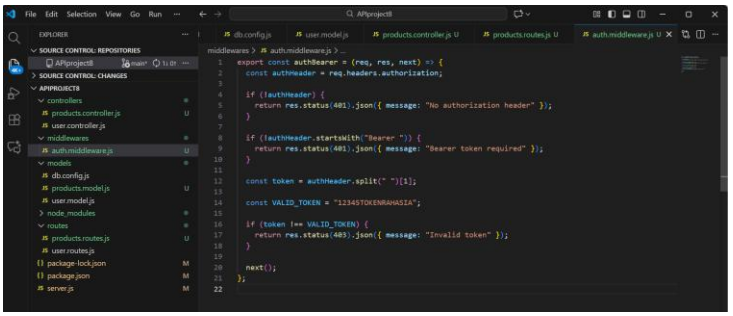
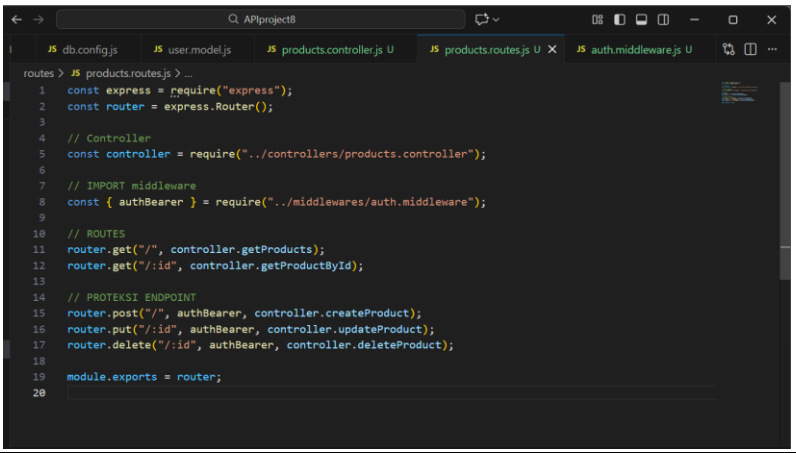
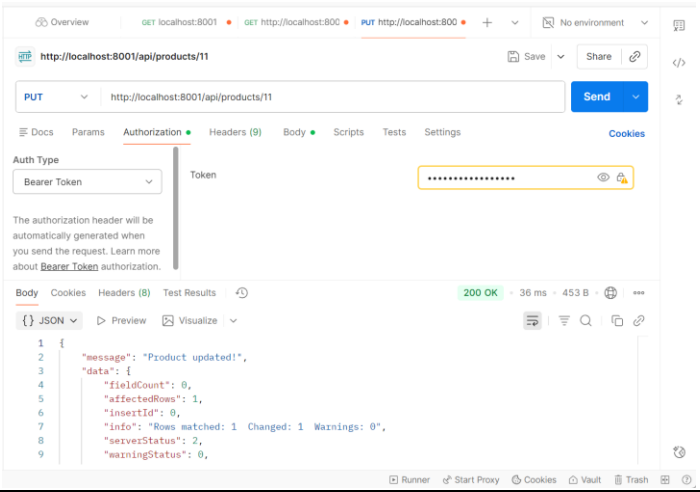
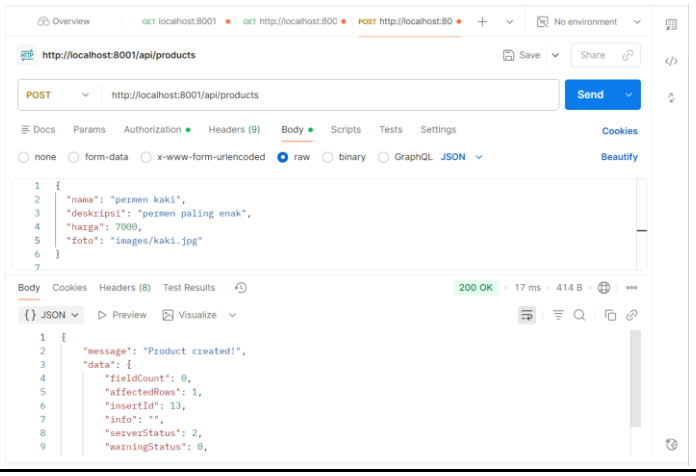
No.	Instruksi	Screenshot	Ke
A.	Installasi dan Konfigurasi		
1.	Membuat table baru products		
2.	Mengisi table producst		

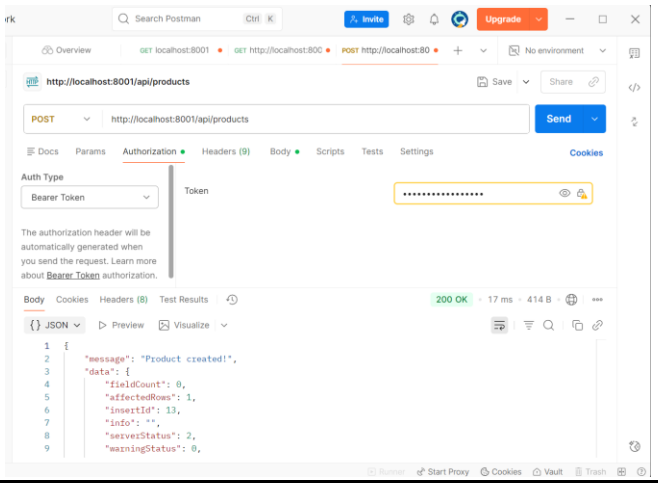
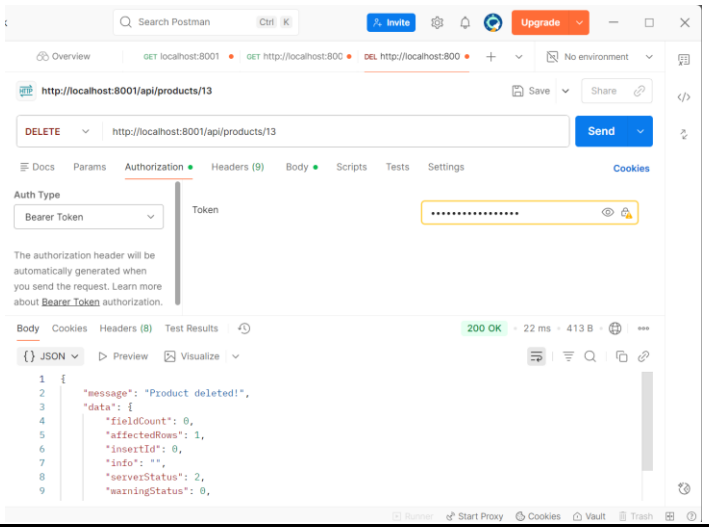
	id	nama	deskripsi	harga	foto
Edit Copy Delete	1	Indomie Goreng	Mie goreng instan enak	2500	images/miegoreng.jpg
Edit Copy Delete	2	Susu Ultra	Susu UHT full cream	6000	images/susu-ultra.jpg
Edit Copy Delete	3	Chitato	Snack kentang BBQ	12000	images/chitato.jpg
Edit Copy Delete	4	Aqua 600ml	Air mineral	3500	images/aqua600.jpg
Edit Copy Delete	5	Yakult	Minuman probiotik	9000	images/yakult.jpg
Edit Copy Delete	6	Silverqueen	Coklat almond	21000	images/silverqueen.jpg
Edit Copy Delete	7	Kopi ABC	Kopi instan	1500	images/kopiabc.jpg
Edit Copy Delete	8	Tango Wafer	Wafer vanilla	8000	images/tango.jpg
Edit Copy Delete	9	Taro Net	Snack rumput laut	7000	images/taro.jpg
Edit Copy Delete	10	Pocky Coklat	Stik snack coklat	11000	images/pocky.jpg

Check all With selected: Edit Copy Delete Export

3.	Membuat endpoint	
4.	Struktur file	
5.	Test API localhost:8001/api/products/	
6.	Test API localhost:8001/api/products/	

7.	POST : localhost:8001/api/products/	
	PUT : localhost:8001/api/products/:id	
	DELETE: localhost:8001/api/products/:id	
B.	Github dan Viscode	

1.	Membuat folder minddelware dan file auth.middleware.js	
2.	Memambahkan autbearer di products.routes.js	
3.	Mengakses put setelah di bearer	
4.	Mengakses post setelah di bearer	

5.	Mengakses post setelah di bearer	
6.	Mengakses delete setelah di bearer	
E.	GITHUB	
1.	Link github	https://github.com/ftrnaa/latihan-8.git