

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 .

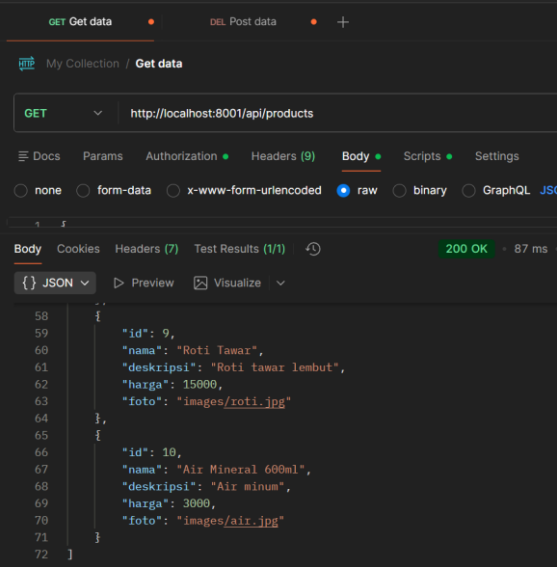
git commit -m "first commit"

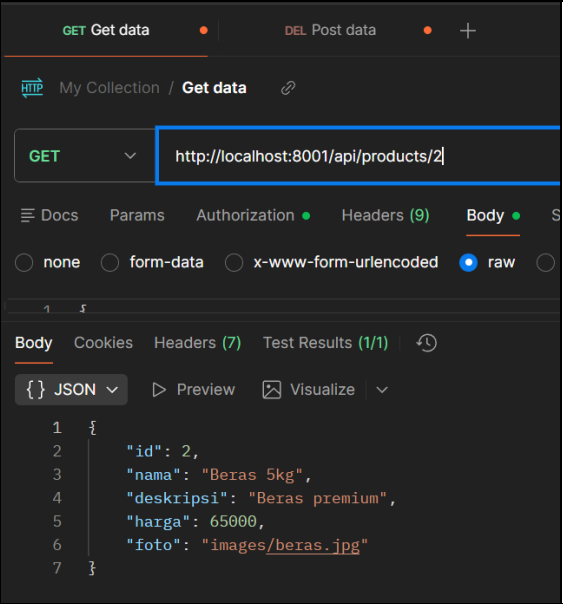
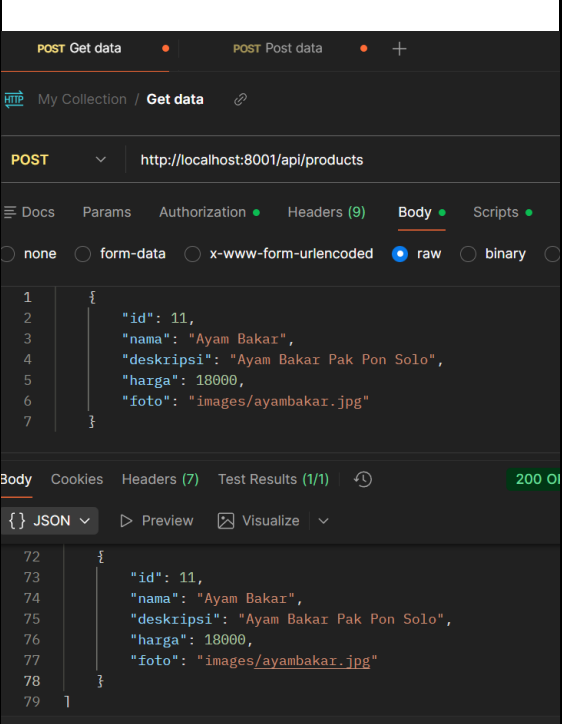
git branch -M main

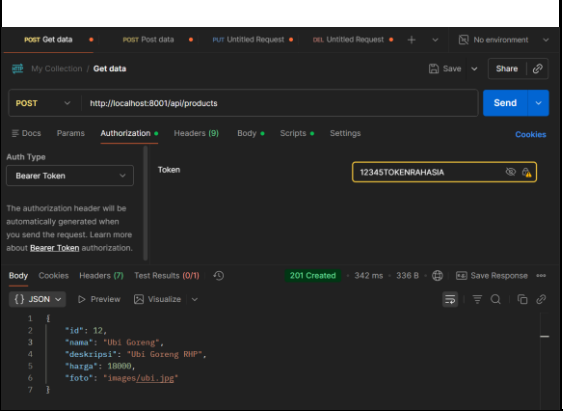
git remote add origin https://github.com/agunghakase/Latihan9.git

git push -u origin main

Hasil Pengerjaan

No.	Instruksi	Screenshot	Kendala/Saran
A.	Instalasi dan Konfigurasi		
1.	GET		
2.	GET ID 2		

		 <p>The screenshot shows a GET request in Postman to the URL <code>http://localhost:8001/api/products/2</code>. The response body is in JSON format, showing a product with the following details:</p> <pre>{ "id": 2, "nama": "Beras 5kg", "deskripsi": "Beras premium", "harga": 65000, "foto": "images/beras.jpg"}</pre>	
3	POST	 <p>The screenshot shows a POST request in Postman to the URL <code>http://localhost:8001/api/products</code>. The response body is in JSON format, showing a product with the following details:</p> <pre>{ "id": 11, "nama": "Ayam Bakar", "deskripsi": "Ayam Bakar Pak Pon Solo", "harga": 18000, "foto": "images/ayambakar.jpg"}</pre>	
4	PUT		

			
5	DELETE		
6	POST TOKENRAHASIA		
B.	Github dan Viscode		
1.			
2.			
...			