

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();
};
```

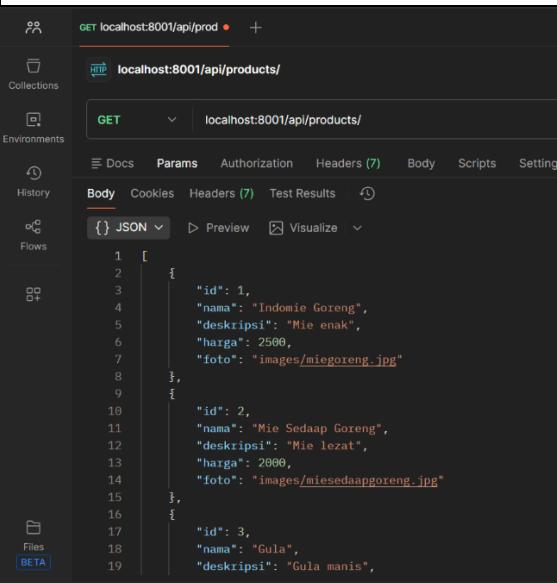
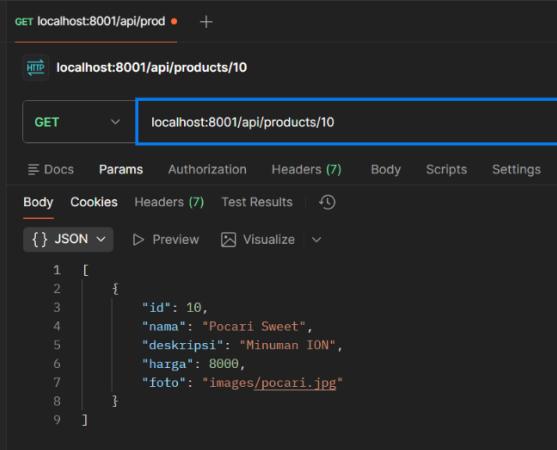
2. Pada file product.routes.js panggil auth.middleware.js
import { authBearer } from "../middleware/auth.middleware.js"
3. Lalu tambahkan authBearer ke endpoints yang perlu di proteksi
router.post("/", authBearer, createProduct);
router.put("/:id", authBearer, updateProduct);
router.delete("/:id", authBearer, deleteProduct);
4. Gunakan POSTMAN untuk akses 3 endpoints ini dengan menambahkan bearer **12345TOKENRAHASIA**

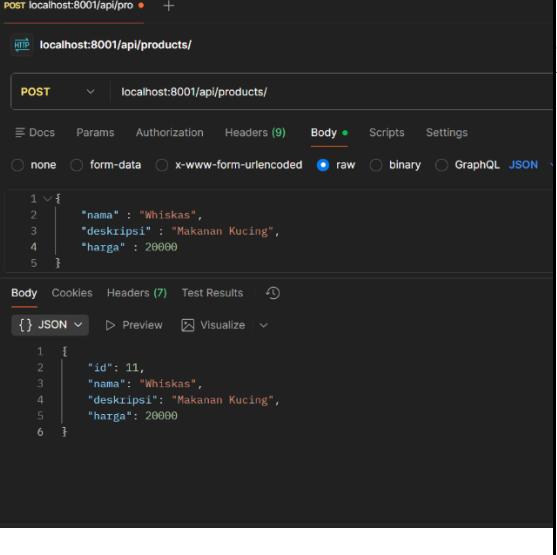
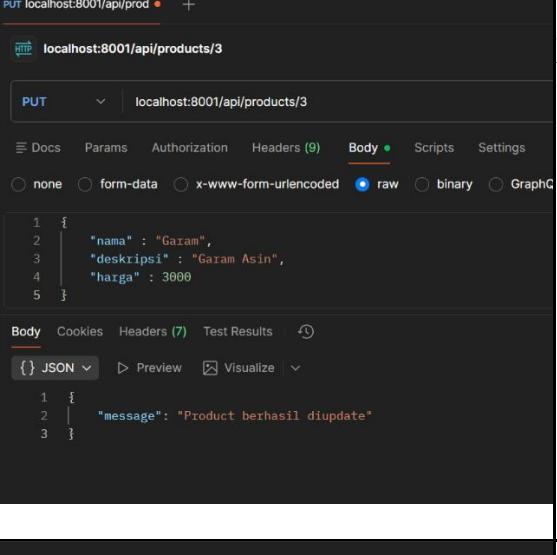
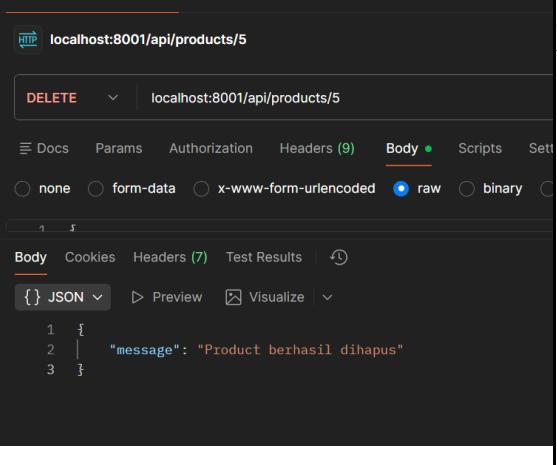
F. Github + Visual Code

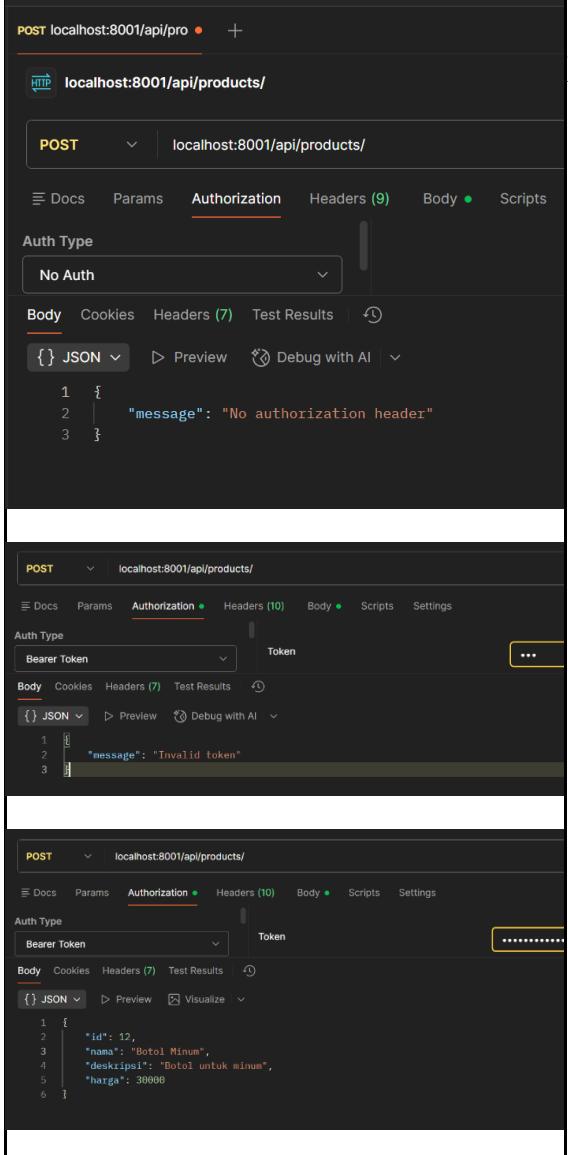
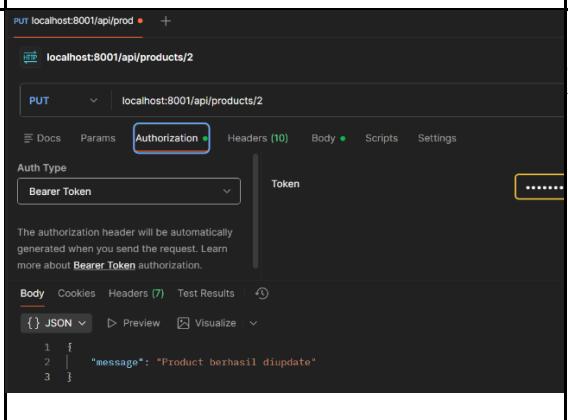
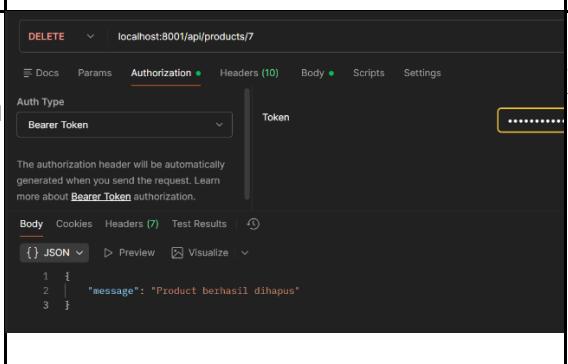
- Buat proyek di Github dengan nama **Latihan9**

```
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 Penggerjaan

No.	Instruksi	Screenshot	Kendala/Saran
A.	Testing API		
1.	localhost:8001/api/products/ (GET)		Tidak ada kendala ataupun saran pada tahapan ini.
2.	localhost:8001/api/products/:id (GET)		Tidak ada kendala ataupun saran pada tahapan ini.

3.	localhost:8001/api/products/ (POST)		Tidak ada kendala ataupun saran pada tahapan ini.
4.	localhost:8001/api/products/:id (PUT)		Tidak ada kendala ataupun saran pada tahapan ini.
5.	localhost:8001/api/products/:id (DELETE)		Tidak ada kendala ataupun saran pada tahapan ini.
B.	authBearer		

1.	<pre>router.post('/', authBearer, productController.createProduct);</pre>	 <p>The first screenshot shows a POST request to <code>localhost:8001/api/products/</code> with 'No Auth' selected in the Authorization dropdown. The response body is: <pre> 1 { 2 "message": "No authorization header" 3 }</pre></p> <p>The second screenshot shows a POST request with 'Bearer Token' selected in the Authorization dropdown, but the token field is empty. The response body is: <pre> 1 { 2 "message": "Invalid token" 3 }</pre></p> <p>The third screenshot shows a POST request with a valid Bearer token in the Authorization dropdown. The response body is: <pre> 1 [2 { 3 "id": 12, 4 "name": "Botol Minum", 5 "deskripsi": "Botol untuk minum", 6 "harga": 30000 7 } 8]</pre></p>	Tidak ada kendala ataupun saran pada tahapan ini.
2.	<pre>router.put('/:id', authBearer, productController.updateProduct);</pre>	 <p>A PUT request to <code>localhost:8001/api/products/2</code> with 'Bearer Token' selected in the Authorization dropdown. The response body is: <pre> 1 { 2 "message": "Product berhasil diupdate" 3 }</pre></p>	Tidak ada kendala ataupun saran pada tahapan ini.
3.	<pre>router.delete('/:id', authBearer, productController.deleteProduct);</pre>	 <p>A DELETE request to <code>localhost:8001/api/products/7</code> with 'Bearer Token' selected in the Authorization dropdown. The response body is: <pre> 1 { 2 "message": "Product berhasil dihapus" 3 }</pre></p>	Tidak ada kendala ataupun saran pada tahapan ini.

C.	Github		
1.	Commit	<pre>PS D:\expressjs\APIproject9> git push -u origin main Enumerating objects: 5319, done. Counting objects: 100% (5319/5319), done. Delta compression using up to 12 threads Compressing objects: 100% (5065/5065), done. Writing objects: 100% (5319/5319), 5.73 MiB 528.00 KiB/s, done. Total 5319 (delta 1072), reused 0 (delta 0), pack-reused 0 remote: Resolving deltas: 100% (1072/1072), done. To https://github.com/Vincentt-0/Latihan9.git * [new branch] main -> main branch 'main' set up to track 'origin/main'. PS D:\expressjs\APIproject9></pre>	Tidak ada kendala ataupun saran pada tahapan ini.
2.	Link Github	https://github.com/Vincentt-0/Latihan9	