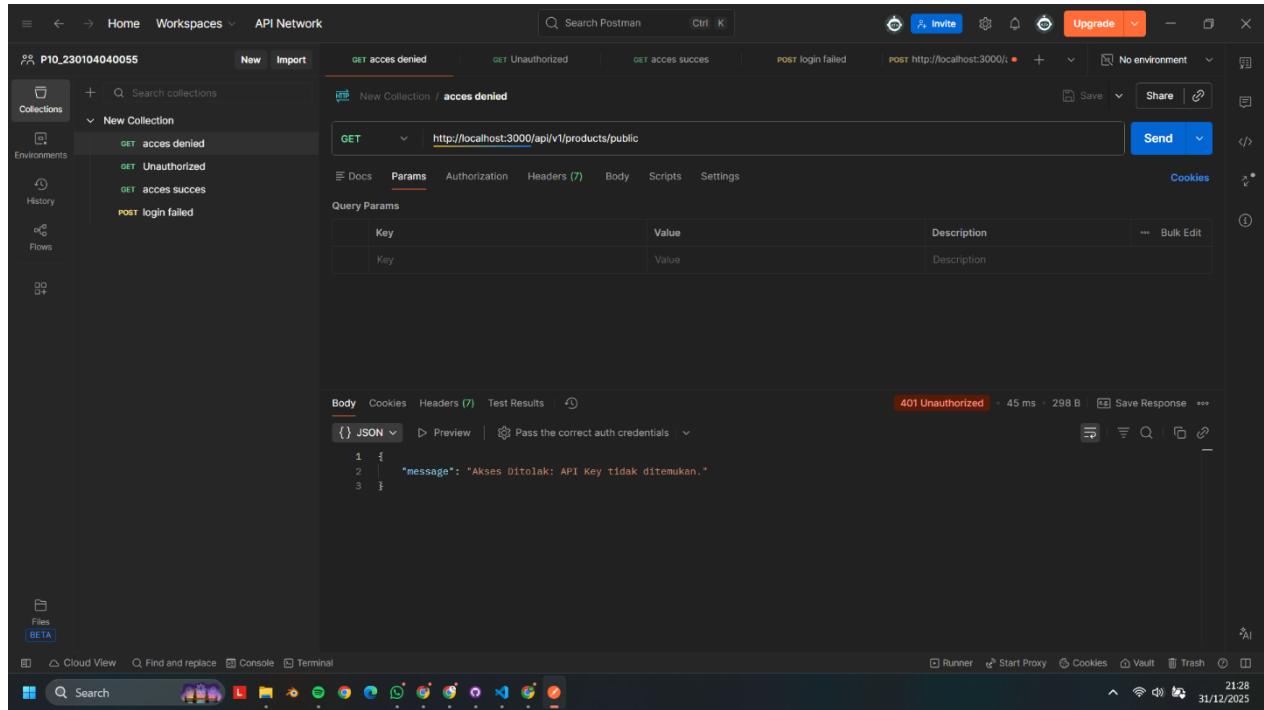


PRAKTIKUM 10

Simulasi API Key & OAuth 2.0

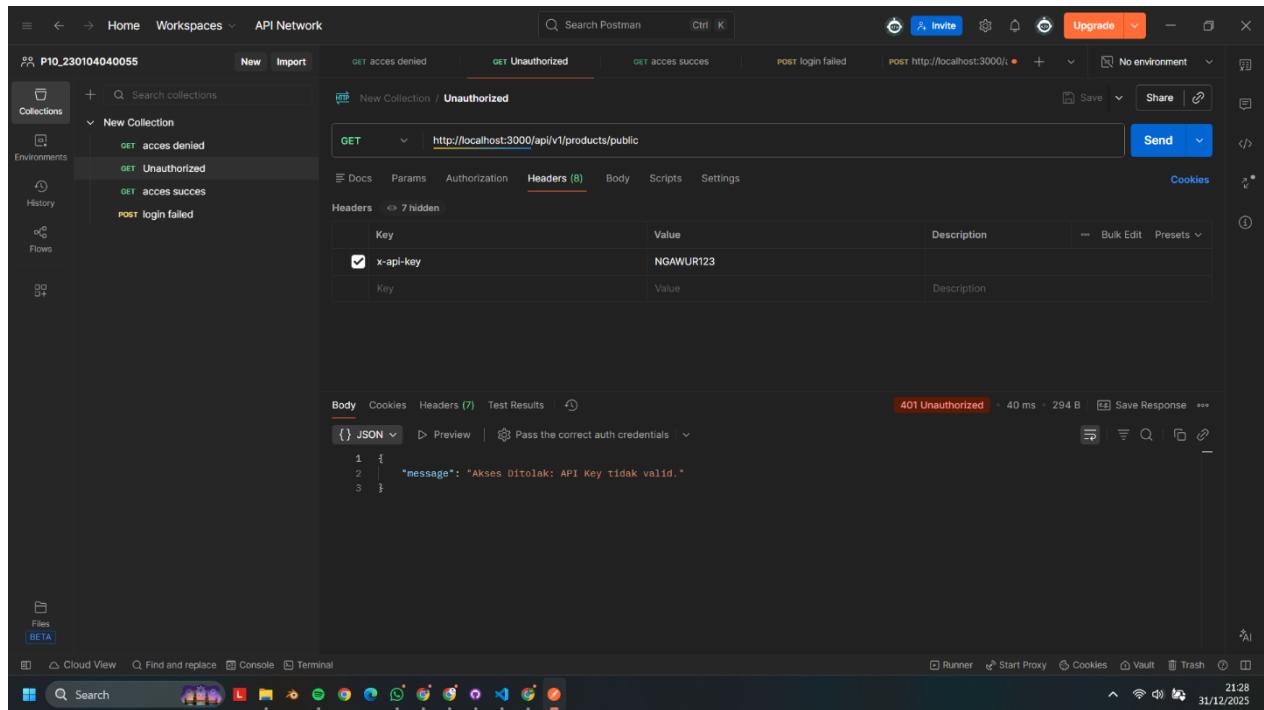
Nama : Febi Novia Putri
Kelas : TI23B
NIM : 230104040055

1. Check no API



The screenshot shows the Postman application interface. On the left, there's a sidebar with collections, environments, and flows. The main area displays a collection named "P10_230104040055". Under this collection, there are several requests: "acces denied", "Unauthorized", "acces succes", and "login failed". The "acces denied" request is currently selected. It shows a GET method pointing to "http://localhost:3000/api/v1/products/public". The "Headers" tab is active, showing an empty header. The "Body" tab also contains an empty JSON object. In the "Test Results" tab, it shows a 401 Unauthorized response with a duration of 45 ms and a size of 298 B. The response body is a JSON object with a single key "message": "Akses Ditolak: API Key tidak ditemukan."

2. Check Fake API



This screenshot shows the same Postman interface as the previous one, but with a different request. The "Unauthorized" request is now selected. It has a "Headers" tab with one entry: "x-api-key" set to "NGAWUR123". The "Body" tab is empty. In the "Test Results" tab, it shows a 401 Unauthorized response with a duration of 40 ms and a size of 294 B. The response body is a JSON object with a single key "message": "Akses Ditolak: API Key tidak valid."

3. Check API Key Valid

The screenshot shows the Postman interface with a collection named 'P10_230104040055'. A new collection named 'acces success' is being created. A GET request is made to `http://localhost:3000/api/v1/products/public`. The response status is 200 OK, and the response body contains JSON data for two products:

```
1 {
  "message": "Daftar Produk (Akses: Public App Client A)",
  "data": [
    {
      "_id": "6955235c9131f17523662a89",
      "name": "Laptop Gaming Pro",
      "price": 15000000,
      "stock": 10,
      "description": "Laptop kencang.",
      "createdAt": "2025-12-31T13:21:32.093Z",
      "updatedAt": "2025-12-31T13:21:32.093Z"
    },
    {
      "_id": "6955235c9131f17523662a8b",
      "name": "Keyboard Mekanik",
      "price": 1500000,
      "stock": 50,
      "description": "Clicky sound."
    }
  ]
}
```

4. Login Admin

The screenshot shows the Postman interface with a collection named 'P10_230104040055'. A POST request is made to `http://localhost:3000/api/v1/auth/token` with the following raw JSON body:

```
1 {
  "username": "admin",
  "password": "password123"
}
```

The response status is 200 OK, and the response body is a JWT token:

```
1 {
  "token_type": "Bearer",
  "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpcC1ydjIyJyJib2YmQ3M2I9yZAiNwF1NzAxNiiSInJvD6Ui01jhZGipbIisImIhdCI6MTc2NzQyNjg2Myw1ZXhwIjoxNzY4MDMxNjYzTQ.m_H-0V4XjnMmePcsSPUY5xra59CCFp0QN0e-U3E",
  "user": {
    "id": "6958cb0ebd73b4c955ae7616",
    "username": "admin",
    "role": "admin"
  }
}
```

5. Login User

The screenshot shows the Postman application interface. A collection named "P10_2301040055" is selected. A new collection named "login user" is being created. A POST request is made to `http://localhost:3000/api/v1/auth/token` with the following body:

```
[{"username": "userbiasa", "password": "userpass"}]
```

The response status is `200 OK` with a response time of 136 ms and a size of 545 B. The response body is a JSON object containing a token and user information.

6. Create Product (Admin)

The screenshot shows the Postman application interface. A collection named "P10_2301040055" is selected. A new collection named "POST ADMIN" is being created. A POST request is made to `http://localhost:3000/api/v1/products/private` with the following headers:

Header	Value
User-Agent	PostmanRuntime/7.51.0
Accept	*
Accept-Encoding	gzip, deflate, br
Connection	keep-alive
Authorization	Bearer eyJhbGciOiUzIi1nIiR5cCl6IkpxVCJ9.eyJpZ... Key

The response status is `201 Created` with a response time of 26 ms and a size of 444 B. The response body is a JSON object containing a message and product data.

7. Create Product (User-Fail)

The screenshot shows the Postman application interface. A collection named 'P10_230104040055' is selected. A new collection named 'post user' is being created under it. A POST request is configured to send to `http://localhost:3000/api/v1/products/private`. The Headers tab shows the following configuration:

Key	Description
User-Agent	PostmanRuntime/7.51.0
Accept	/*
Accept-Encoding	gzip, deflate, br
Connection	keep-alive
Authorization	Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9eyJpZC...

The Body tab shows a JSON payload:

```
1 {  
2   "message": "Akses Ditolak: Hanya Admin."  
3 }
```

The response status is 403 Forbidden, with a response time of 9 ms and a body size of 283 B. The response content is:

```
1 {  
2   "message": "Akses Ditolak: Hanya Admin."  
3 }
```

8. Create Product (No Token)

The screenshot shows the Postman application interface. A collection named 'P10_230104040055' is selected. A POST request is configured to send to `http://localhost:3000/api/v1/products/private`. The Headers tab shows the following configuration:

Key	Description
Content-Type	application/json

The Body tab shows a JSON payload:

```
1 {  
2   "name": "Mouse IllegaL", "price": 1000  
3 }
```

The response status is 401 Unauthorized, with a response time of 8 ms and a body size of 298 B. The response content is:

```
1 {  
2   "message": "Akses Ditolak: API Key tidak ditemukan."  
3 }
```