

2311104023
Yoga Eka Pratama
SE0701

Tugas Pendahuluan 09

The screenshot displays a VS Code environment with an OpenAPI SwaggerUI preview. The left sidebar shows the Explorer and Outline panels. The center panel shows the OpenAPI definition for 'tugasPendahuluan.json'. The right panel shows the SwaggerUI preview for the 'default' server at 'http://localhost:5000 - Server Lokal Mahasiswa API'.

OpenAPI Definition (tugasPendahuluan.json):

```
1 {
2   "openapi": "3.0.0",
3   "info": {
4     "title": "Mahasiswa API",
5     "description": "API untuk Manajemen Data Mahasiswa kelompok",
6     "version": "1.0.0"
7   },
8   "servers": [
9     {
10      "url": "http://localhost:5000",
11      "description": "Server Lokal Mahasiswa API"
12    }
13  ],
14  "paths": {
15    "/api/mahasiswa": {
16      "get": {
17        "summary": "Ambil semua mahasiswa",
18        "description": "Mengembalikan daftar mahasiswa yang ter",
19        "responses": {
20          "200": {
21            "description": "Daftar mahasiswa berhasil diambil",
22            "content": {
23              "application/json": {
24                "schema": {
25                  "type": "array",
26                  "items": {
27                    "$ref": "#/components/schemas/Mahasiswa"
28                  }
29                }
30              }
31            }
32          }
33        }
34      },
35      "post": {
36        "summary": "Tambahkan mahasiswa baru",
37        "description": "Menambahkan objek mahasiswa baru ke daft",
38        "requestBody": {
39          "required": true,
40          "content": {
41            "application/json": {
42              "schema": {
43                "type": "object",
44                "properties": {
45                  "name": {
46                    "type": "string",
47                    "required": true
48                  },
49                  "nim": {
50                    "type": "string",
51                    "required": true
52                  }
53                }
54              }
55            }
56          }
57        }
58      }
59    }
60  }
61 }
```

SwaggerUI Preview (default server):

GET /api/mahasiswa Ambil semua mahasiswa

Mengembalikan daftar mahasiswa yang tersimpan

Parameters: No parameters

Responses:

Code	Description	Links
200	Daftar mahasiswa berhasil diambil	No links

Media type: application/json

Example Value:

```
{
  "name": "string",
  "nim": "string"
}
```

POST /api/mahasiswa Tambahkan mahasiswa baru

Menambahkan objek mahasiswa baru ke daftar

Parameters: No parameters

Request body required: application/json

Example Value:

```
{
  "name": "John Doe",
  "nim": "1302199999"
}
```

Responses:

Code	Description	Links
201	Mahasiswa berhasil ditambahkan	No links

GET /api/mahasiswa/{index} Ambil mahasiswa berdasarkan index

DELETE /api/mahasiswa/{index} Hapus mahasiswa berdasarkan index

Schemas

The image displays two screenshots of a VS Code editor with an OpenAPI SwaggerUI preview. The left pane shows the OpenAPI specification file, and the right pane shows the rendered SwaggerUI interface.

Top Screenshot: GET /api/mahasiswa/{index}

Method: GET
Path: /api/mahasiswa/{index}
Summary: Ambil mahasiswa berdasarkan index

Parameters:

Name	Description
index * required	index

Responses:

Code	Description	Links
200	Data mahasiswa ditemukan	No links
404	Mahasiswa tidak ditemukan	No links

Example Value:

```
{  "nama": "string",  "nim": "string"}
```

Bottom Screenshot: DELETE /api/mahasiswa/{index}

Method: DELETE
Path: /api/mahasiswa/{index}
Summary: Hapus mahasiswa berdasarkan index

Parameters:

Name	Description
index * required	index

Responses:

Code	Description	Links
204	Mahasiswa berhasil dihapus	No links
404	Mahasiswa tidak ditemukan	No links

Schemas:

Mahasiswa >

Penjelasan:

paths: Endpoint-endpoint yang tersedia dalam API:

- GET /api/mahasiswa: Mengembalikan seluruh daftar mahasiswa.
- POST /api/mahasiswa: Menambahkan data mahasiswa baru.
- GET /api/mahasiswa/{index}: Mengambil data mahasiswa berdasarkan urutan/index dalam list.
- DELETE /api/mahasiswa/{index}: Menghapus mahasiswa berdasarkan index.

components/schemas: Mendefinisikan struktur objek Mahasiswa yang terdiri dari:

- name: Nama mahasiswa
 - nim: Nomor induk mahasiswa
- Keduanya bersifat *required*.