

Pertanyaan Praktikum 1

a. Fungsi GetAPI: Untuk menjalankan method promise yang dimana domainPath dapat di *resolve* (di *fetch*) atau di *reject*

Fungsi getNewsBlog: Untuk mendeklarasi bagaimana artikel akan di fetch

Pertanyaan Praktikum 2

1. Untuk menambah data yang akan di tambahkan pada file .json
2. Adalah value yang akan di jadikan acuan yaitu insertArtikel

Pertanyaan Praktikum 3

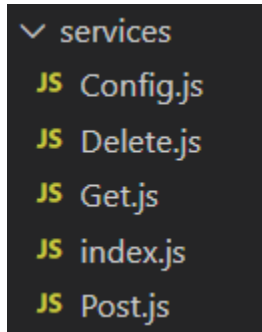
1. Untuk menghapus data yang ada pada file .json
2. Adalah value yang akan di jadikan acuan yaitu insertArtikel

Pertanyaan Praktikum 4

1. Membuat file baru kemudian mengimport-nya pada index.js

TUGAS GLOBAL API

1. Buat folder services di dalam folder src, kemudian buat file baru. Seperti berikut



2. Config

```
export const domainPath = "http://localhost:3001";
```

3. Delete

```
import { domainPath } from "./Config";

const DeleteAPI = (path, data) => {
  const promise = new Promise((resolve, reject) => {
    fetch(`${domainPath}/${path}/${data}`, {
      method: "DELETE",
      headers: {
        Accept: "application/json",
        "Content-Type": "application/json",
      },
    }).then(
      (result) => {
        resolve(result);
      },
      (err) => {
        reject(err);
      }
    );
  });
  return promise;
};
```

```
export default DeleteAPI;
```

4. Get

```
import { domainPath } from "./Config";

const GetAPI = (path) => {
  const promise = new Promise((resolve, reject) => {
    fetch(`${domainPath}/${path}`)
      .then((response) => response.json())
      .then(
        (result) => {
          resolve(result);
        },
        (err) => {
          reject(err);
        }
      );
  });
  return promise;
};

export default GetAPI;
```

5. index

```
import GetAPI from "./Get";
import PostAPI from "./Post";
import DeleteAPI from "./Delete";

const getNewsBlog = () => GetAPI("posts?_sort=id&_order=desc");
const postNewsBlog = (dataYgDiKirim) => PostAPI("posts", dataYgDiKirim);
const deleteNewsBlog = (dataYgDiHapus) => DeleteAPI("posts", dataYgDiHapus);

const API = {
  getNewsBlog,
  postNewsBlog,
  deleteNewsBlog,
```

```
};  
  
export default API;
```

6. Post

```
import { domainPath } from "./Config";  
  
const PostAPI = (path, data) => {  
  const promise = new Promise((resolve, reject) => {  
    fetch(`${domainPath}/${path}`, {  
      method: "post",  
      headers: {  
        Accept: "application/json",  
        "Content-Type": "application/json",  
      },  
      body: JSON.stringify(data),  
    }).then(  
      (result) => {  
        resolve(result);  
      },  
      (err) => {  
        reject(err);  
      }  
    );  
  });  
  return promise;  
};  
  
export default PostAPI;
```

7. Blogpost

```
import { domainPath } from "./Config";  
  
const PostAPI = (path, data) => {  
  const promise = new Promise((resolve, reject) => {  
    fetch(`${domainPath}/${path}`, {  
      method: "post",  
      headers: {
```

```

    Accept: "application/json",
    "Content-Type": "application/json",
  },
  body: JSON.stringify(data),
}).then(
  (result) => {
    resolve(result);
  },
  (err) => {
    reject(err);
  }
);
});
return promise;
};

export default PostAPI;

```

8. Post

```

import React from "react";

const Post = props => {
  return (
    <div className="artikel">
      <div className="gambar-artikel">
        </img>
      </div>
      <div className="konten-artikel">
        <p className="nim">NIM: {props.nim}</p>
        <p className="nama">Nama: {props.nama}</p>
        <p className="alamat">Alamat: {props.alamat}</p>
        <p className="hp">Nomor HP: {props.hp}</p>
        <p className="angkatan">Angkatan (Tahun): {props.angkatan}
      </p>
        <p className="status">Status: {props.status}</p>
        <button className="btn btn-sm btn-warning"
          onClick={() => props.hapusArtikel(props.nim)}>

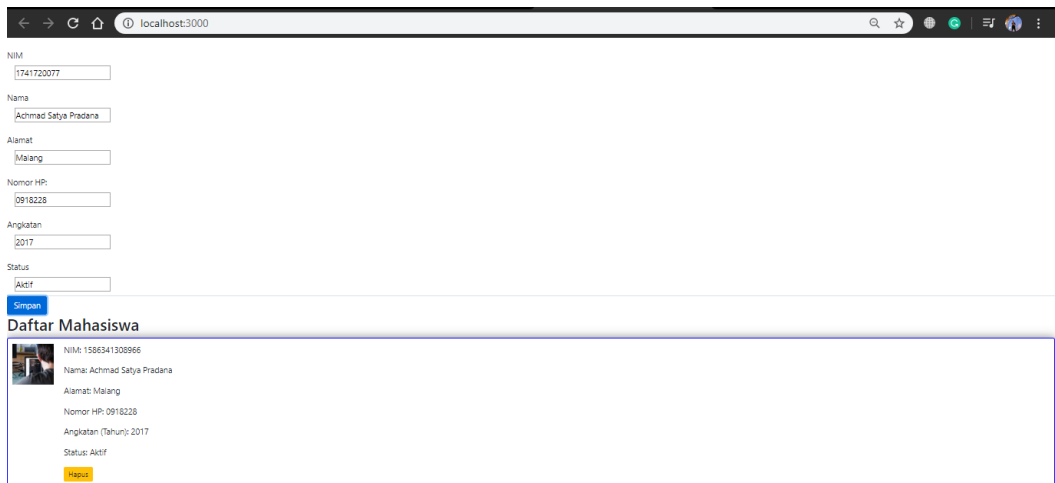
```

```
        Hapus
      </button>
    </div>
  </div>
);
};


export default Post;
```

9. Hasil

Penambahan Mahasiswa:



The screenshot shows a web browser at localhost:3000. On the left is a registration form with fields for NIM, Nama, Alamat, Nomor HP, Angkatan, Status, and a 'Simpan' button. On the right is a table titled 'Daftar Mahasiswa' containing one student record with a profile picture, name, NIM, address, phone number, graduation year, status, and a 'Hapus' button.

Daftar Mahasiswa							
	NIM: 1586341308966	Nama: Achmad Satya Pradana	Alamat: Malang	Nomor HP: 0918228	Angkatan (Tahun): 2017	Status: Aktif	<button>Hapus</button>

Hapus

localhost:3000

NIM

Nama

Alamat

Nomor HP:

Angkatan

Status

[Simpan](#)

Daftar Mahasiswa

Status pada json-server

```
POST /posts 201 44.912 ms - 166
GET /posts?_sort=id&_order=desc 200 4.633 ms - 188
DELETE /posts/1586341308966 200 5.911 ms - 2
GET /posts?_sort=id&_order=desc 200 5.575 ms - 2
```

Post = Penambahan Mahasiswa

Delete = Penghapusan Mahasiswa