

LAPORAN MATA KULIAH  
PEMROGAMAN BERBASIS FRAMEWORK  
“ MODUL 9 ”



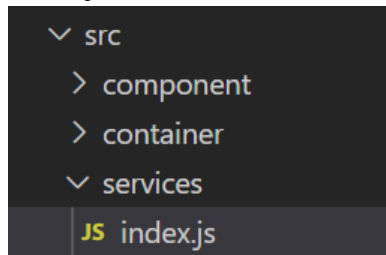
Nama : Ivan Crisnanda  
NIM : 1741720024  
Kelas : TI-3C

PROGRAM STUDI D-IV TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI MALANG

2020

- **Langkah Praktikum 1 – Global API Service GET**

index.js



BlogPost.jsx

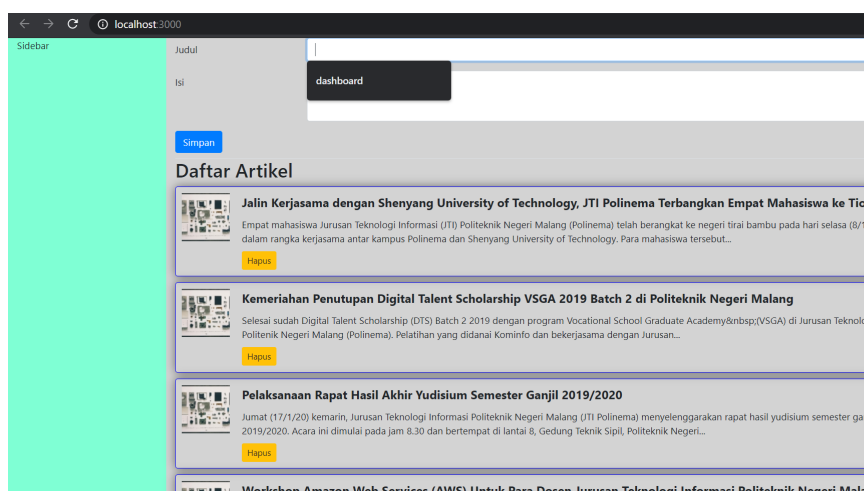
```
JS index.js  BlogPost.jsx X
src > container > BlogPost > BlogPost.jsx > ...
1  import React, {Component} from "react";
2  import './BlogPost.css';
3  import Post from "../../component/BlogPost/Post";
4
5  class BlogPost extends Component {
6    state = {
7      listArtikel: [],
8      insertArtikel: {
9        userId: 1,
10       id: 1,
11       title: "",
12       body: ""
13     }
14   }
15
16   ambilDataDariServerAPI = () => {
17     fetch('http://localhost:3001/posts')
18     .then(response => response.json())
19     .then(jsonHasilAmbilDariAPI => {
20       this.setState({
21         listArtikel: jsonHasilAmbilDariAPI
22       })
23     })
24   }
25 }
```

## index.js

```
JS index.js x BlogPost.jsx
src > services > JS index.js > [e] GetAPI
1  const domainPath = 'http://localhost:3001';
2  const GetAPI = (path) => {
3    const promise = new Promise((resolve, reject) => {
4      fetch(`${domainPath}/${path}`)
5        .then(response => response.json())
6        .then((result) => {
7          resolve(result);
8        }, (err) => {
9          reject(err);
10         })
11    })
12    return promise;
13  }
14
15  const getNewsBlog = () => GetAPI('posts?_sort=id&_order=desc');
16
17  const API = {
18    getNewsBlog
19  }
20
21  export default API;
```

## BlogPost.js

```
17  ambilDataDariServerAPI = () => {
18    API.getNewsBlog().then(result => {
19      this.setState({
20        listArtikel: result
21      })
22    })
23  }
```



- **Pertanyaan Praktikum 1**

- a. Perhatikan file index.js, apa tujuan dibuatnya fungsi GetAPI pada baris 2 dan fungsi getNewsBlog pada baris 16?

Jawaban :

- Fungsi GetAPI() untuk menyimpan dan menunjukkan alamat url mana yang akan di request.
- Fungsi getNewsBlog() fungsi untuk menjalankan GetAPI() dalam variable const.

- **Praktikum 2**

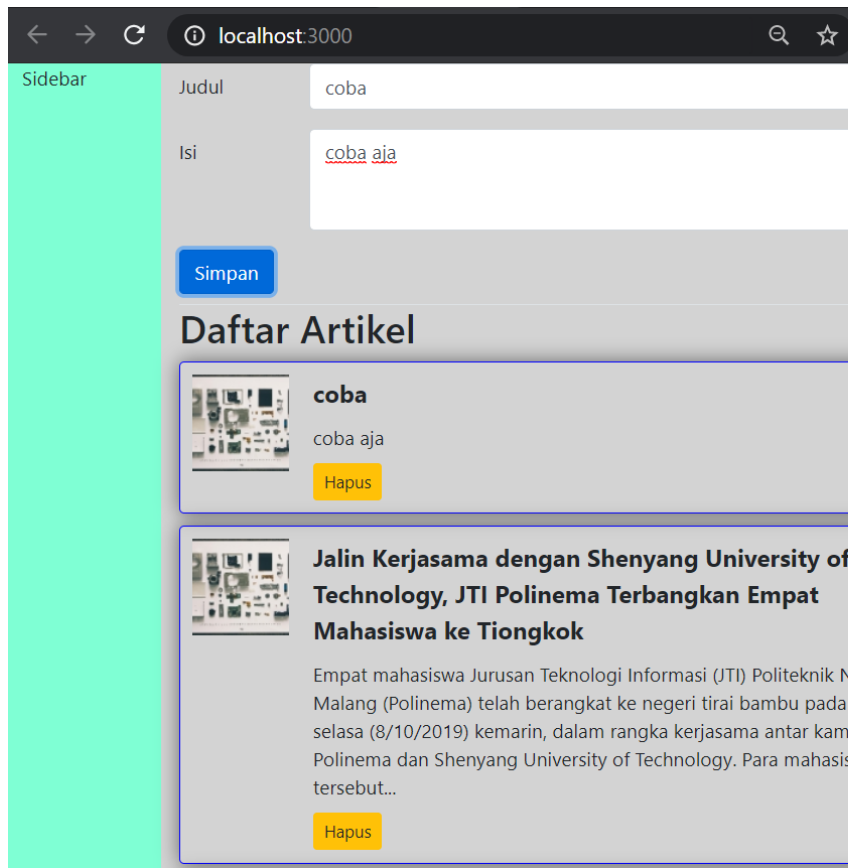
BlogPost.js

```
47 handleTombolSimpan = () => { // fungsi untuk meng-handle tombol simpan
48   fetch('http://localhost:3001/posts',{
49     method: 'post', // method POST untuk input/insert data
50     headers: {
51       'Accept': 'application/json',
52       'Content-Type': 'application/json'
53     },
54     body: JSON.stringify(this.state.insertArtikel) // kirimkan ke body request untuk data artikel yang akan
55   })
56   .then((response) => {
57     this.ambilDataDariServerAPI(); // reload / refresh data
58   });
59 }

15 const PostAPI = (path, data) => {
16   const promise = new Promise((resolve, reject) => {
17     fetch(`${domainPath}/${path}`, {
18       method: 'post',
19       headers: {
20         'Accept': 'application/json',
21         'Content-Type': 'application/json'
22       },
23       body: JSON.stringify(data)
24     })
25     .then((result) => {
26       resolve(result);
27     }, (err) => {
28       reject(err);
29     })
30   })
31   return promise;
32 }

33
34 const getNewsBlog = () => GetAPI('posts?_sort=id&_order=desc');
35 const postNewsBlog = (dataYgDiKirim) => PostAPI('posts', dataYgDiKirim)
36
37 const API = [
38   getNewsBlog,
39   postNewsBlog
40 ]

48 handleTombolSimpan = () => { // fungsi untuk meng-handle tombol simpan
49   API.postNewsBlog(this.state.insertArtikel)
50   .then((response) => {
51     this.ambilDataDariServerAPI();
52   });
53 }
```



- **Pertanyaan Praktikum 2**

- a. Perhatikan file index.js, apa tujuan dibuatnya fungsi PostAPI dan fungsi postNewsBlog?

Jawaban :

- Fungsi dari PostAPI() itu sendiri adalah untuk menambah data dimana jika penambahan data sukses menerima response dari server maka akan resolve ke user.
- Fungsi dari postNewsBlog() itu sendiri adalah untuk menjalankan fungsi PostAPI() dalam variable const.

- b. Pada fungsi postNewsBlog, terdapat variable dataYangDiKirim. Apa tujuan dari dibuatnya variable tersebut?

Jawaban :

- Tujuan dari dibuatnya variable tersebut adalah untuk menyimpan data dari form input.

- **Praktikum 3**

BlogPost.jsx.

```

31 handleHapusArtikel = (data) => { // fungsi yang meng-handle button action hapus data
32   fetch('http://localhost:3001/posts/${data}', {method: 'DELETE'}) // alamat URL API yang ingin kita HAPUS datanya
33   .then(res => { // ketika proses hapus berhasil, maka ambil data dari server API lokal
34     this.ambilDataDariServerAPI()
35   })
36 }

```



- **Pertanyaan Praktikum 3**

- a. Perhatikan file index.js, apa tujuan dibuatnya fungsi DeleteAPI dan fungsi deleteNewsBlog?

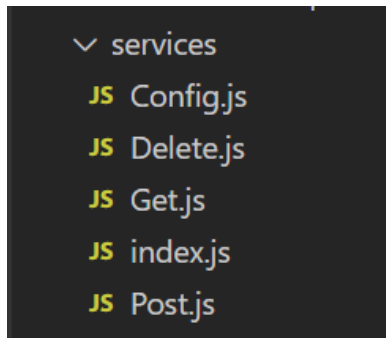
Jawaban :

- Fungsi DeleteAPI() itu sendiri adlaah untuk melakukan penghapusan data dimana jika sukses menerima response dari server maka akan resolve ke user.
  - Fungsi deleteNewsBlog() itu sendiri adalah untuk menjalankan fungsi DeleteAPI() dalam variable const.
- b. Pada fungsi deleteNewsBlog, terdapat variable dataYangDiHapus. Apa tujuan dari dibuatnya variable tersebut?

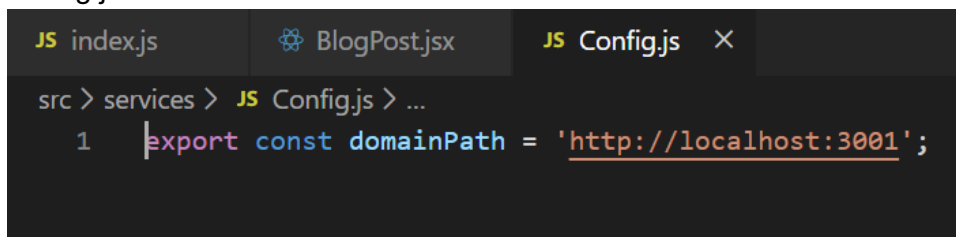
Jawaban :

- Tujuan dibuatnya variable tersebut adalah untuk menyimpan data yang akan kita hapus pada list data .json

- **Praktikum 4**



Config.js



## Get.js

```
JS index.js  BlogPost.jsx  JS Config.js  JS Get.js  X

src > services > JS Get.js > ...
 1  import {domainPath} from './Config';
 2
 3  const GetAPI = (path) => {
 4      const promies = new Promise((resolve,reject) => {
 5          fetch(`${domainPath}/${path}`)
 6          .then(response => response.json())
 7          .then((result) => {
 8              resolve(result);
 9          },(err) => {
10              reject(err);
11          })
12      })
13      return promies;
14  }
15
16  export default GetAPI;
```

## Post.js

```
JS index.js  BlogPost.jsx  JS Post.js  X

src > services > JS Post.js > ...
 1  import {domainPath} from './Config';
 2
 3  const PostAPI = (path,data) => {
 4      const promies = new Promise((resolve,reject)=>{
 5          fetch(`${domainPath}/${path}`, {
 6              method: 'post',
 7              headers: {
 8                  'Accept': 'application/json',
 9                  'Content-Type': 'application/json'
10              },
11              body: JSON.stringify(data)
12          }).then((result)=>{
13              resolve(result);
14          },(err)=>{
15              reject(err);
16          })
17      })
18      return promies;
19  }
20
21  export default PostAPI;
```



## Delete.js

```
JS index.js  BlogPost.jsx  JS Delete.js  X
src > services > JS Delete.js > ...
1  import {domainPath} from './Config';
2
3  const DeleteAPI = (path,data) => {
4      const promies = new Promise((resolve,reject) => {
5          fetch(`${domainPath}/${path}/${data}`, {method: 'DELETE'})
6              .then((result) => {
7                  resolve(result);
8              }, (err) => {
9                  reject(err);
10             })
11      })
12      return promies;
13  }
14
15  export default DeleteAPI;
```

## index.js

```
JS index.js  X  BlogPost.jsx  JS Delete.js
src > services > JS index.js > [0] default
1  import GetAPI from './Get';
2  import PostAPI from './Post';
3  import DeleteAPI from './Delete';
4
5  const getNewsBlog = () => GetAPI('posts?_sort=id&_order=desc');
6  const postNewsBlog = (dataYgDikirim) => PostAPI('posts',dataYgDikirim);
7  const deleteNewsBlog = (dataYangDiHapus) => DeleteAPI('posts',dataYangDiHapus);
8
9  const API = {
10      getNewsBlog,
11      postNewsBlog,
12      deleteNewsBlog,
13  }
14
15  export default API;
```

- **Pertanyaan Praktikum 4**

- a. Bagaimana caranya untuk menambahkan daftar API baru baik untuk method GET, POST, DELETE pada Global API?

Jawaban :

```
src > services > JS indexjs > API > deleteProfile
1  import GetAPI from "../Get";
2  import PostAPI from "../Post";
3  import DeleteAPI from "../Delete";
4
5  // // Daftar API Lama
6  // const getNewsBlog = () => GetAPI('posts?_sort=id&_order=desc');
7  // const postNewsBlog = (dataVgDikirim) => PostAPI('posts',dataVgDikirim);
8  // const deleteNewsBlog = (dataYangDiHapus) => DeleteAPI('posts',dataYangDiHapus);
9
10
11 // Daftar API Baru
12 const getProfile = () => GetAPI('profile?_sort=id&_order=des');
13 const profileBaru = (dataYangDikirim) => PostAPI('profile',dataYangDikirim);
14 const deleteProfile = (dataYangDiHapus) => DeleteAPI('profile',dataYangDiHapus);
15
16 const API = {
17   // getNewsBlog,
18   // postNewsBlog,
19   // deleteNewsBlog,
20   getProfile,
21   profileBaru,
22   deleteProfile
23 };
24
25 export default API;
```

```
src > services > JS indexjs > API > deleteProfile
1  import GetAPI from "../Get";
2  import PostAPI from "../Post";
3  import DeleteAPI from "../Delete";
4
5  // // Daftar API Lama
6  // const getNewsBlog = () => GetAPI('posts?_sort=id&_order=desc');
7  // const postNewsBlog = (dataVgDikirim) => PostAPI('posts',dataVgDikirim);
8  // const deleteNewsBlog = (dataYangDiHapus) => DeleteAPI('posts',dataYangDiHapus);
9
10
11 // Daftar API Baru
12 const getProfile = () => GetAPI('profile?_sort=id&_order=des');
13 const profileBaru = (dataYangDikirim) => PostAPI('profile',dataYangDikirim);
14 const deleteProfile = (dataYangDiHapus) => DeleteAPI('profile',dataYangDiHapus);
15
16 const API = {
17   // getNewsBlog,
18   // postNewsBlog,
19   // deleteNewsBlog,
20   getProfile,
21   profileBaru,
22   deleteProfile
23 };
24
25 export default API;
```

```
src > services > JS indexjs > API > deleteProfile
1  import GetAPI from "../Get";
2  import PostAPI from "../Post";
3  import DeleteAPI from "../Delete";
4
5  // // Daftar API Lama
6  // const getNewsBlog = () => GetAPI('posts?_sort=id&_order=desc');
7  // const postNewsBlog = (dataVgDikirim) => PostAPI('posts',dataVgDikirim);
8  // const deleteNewsBlog = (dataYangDiHapus) => DeleteAPI('posts',dataYangDiHapus);
9
10
11 // Daftar API Baru
12 const getProfile = () => GetAPI('profile?_sort=id&_order=des');
13 const profileBaru = (dataYangDikirim) => PostAPI('profile',dataYangDikirim);
14 const deleteProfile = (dataYangDiHapus) => DeleteAPI('profile',dataYangDiHapus);
15
16 const API = {
17   // getNewsBlog,
18   // postNewsBlog,
19   // deleteNewsBlog,
20   getProfile,
21   profileBaru,
22   deleteProfile
23 };
24
25 export default API;
```