

TUGAS INDIVIDU
PEMROGRAMAN BERBASIS FRAMEWORK

“Global API”



oleh:

Tika Yulianti / 1741720153

TI-3A

PROGRAM STUDI D-IV TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

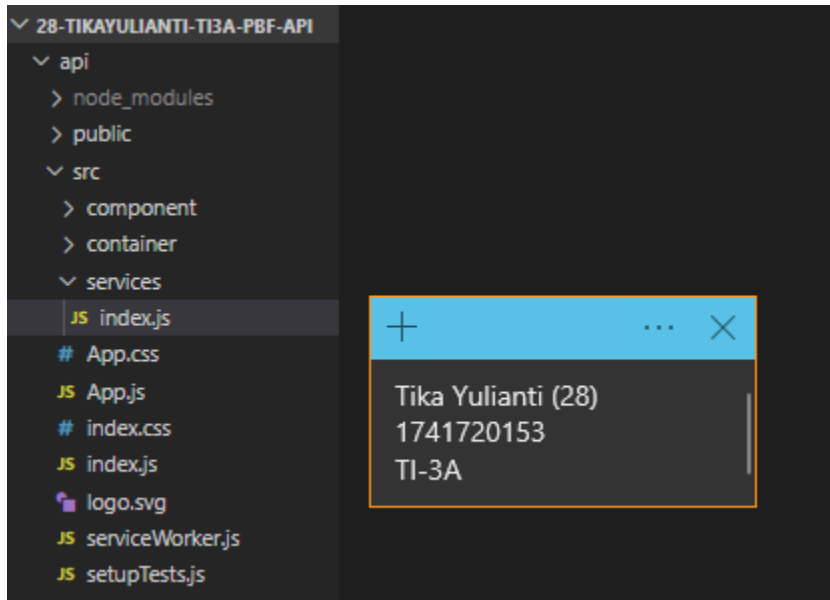
2020

Praktikum 1

Global API service GET

A. Langkah-langkah praktikum 1:

1. Membuat folder baru bernama “**services**” di dalam folder **src**, kemudian membuat file baru bernama **index.js** seperti gambar berikut:



2. Membuka file **index.js** pada folder **services**, lalu menambahkan kode program sebagai berikut:

```
api > src > services > JS index.js > ...
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 |
16 | const getNewsBlog = () => GetAPI('post?_sort=id&_order=src');
17 |
18 | const API = {
19 |   ...getNewsBlog
20 | }
21 |
22 | export default API;
```

3. Membuka file **BlogPost.jsx** pada folder **container**, lalu mengganti fungsi `ambilDataDariServerAPI` seperti kode dibawah ini:

```
...ambilDataDariServerAPI = () => {
...  API.getNewsBlog().then(result => {
...    this.setState({
...      listArtikel: result
...    })
...  })
...}
```

4. Menyimpan kedua file tersebut

B. Pertanyaan praktikum 1

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

Jawaban:

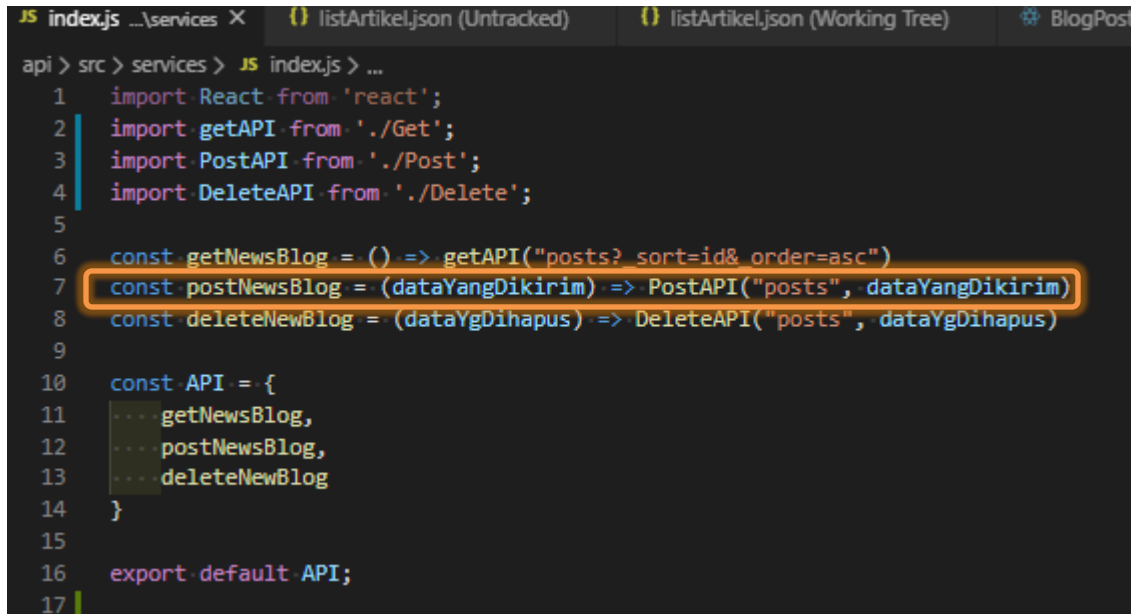
Pemanggilan **GetAPI** berfungsi untuk me-request data artikel. Proses ini memanage ke dalam suatu tempat yaitu **index.js**

Praktikum 2

Global API service POST


A. Langkah-langkah praktikum 2

1. Membuat fungsi untuk menampung action **POST** dari file **BlogPost.jsx** pada file **services/index.js**. Kemudian menginisialisasi fungsi tersebut sebagai berikut:



```
api > src > services > JS index.js > ...
1  import React from 'react';
2  import getAPI from './Get';
3  import PostAPI from './Post';
4  import DeleteAPI from './Delete';
5
6  const getNewsBlog = () => getAPI("posts? sort=id& order=asc")
7  const postNewsBlog = (dataYangDikirim) => PostAPI("posts", dataYangDikirim)
8  const deleteNewBlog = (dataYgDihapus) => DeleteAPI("posts", dataYgDihapus)
9
10 const API = {
11   ... getNewsBlog,
12   ... postNewsBlog,
13   ... deleteNewBlog
14 }
15
16 export default API;
17
```

2. Selanjutnya pindah ke file **BlogPost.jsx** dan mengganti isi dari fungsi **handleTombolSimpan** menjadi seperti berikut:



```
... handleTombolSimpan = () => { ..... // fungsi untuk meng
... // fetch('http://localhost:3001/posts', {
... // ..... method: 'post', .....
... // ..... headers: {
... // ..... 'Accept': 'application/json',
... // ..... 'Content-Type': 'application/json'
... // ..... },
... // ..... body: JSON.stringify(this.state.insertArtikel) ..
... // } })
... API.postNewsBlog(this.state.insertArtikel)
... .then((response) => {
...   this.ambilDataDariServerAPI(); | .....
... });
... }
```

3. Selesai. Kode program pada **BlogPost.jsx** sedikit lebih simple dari seelumnya.

B. Pertanyaan Praktikum 2

1. Perhatikan file **index.js**, apa tujuan dibuatnya fungsi **PostAPI** dan fungsi **postNewsBlog**?

Jawaban:

Fungsi **PostAPI** adalah untuk mengirim data kepada server API.

2. Pada fungsi **postNewsBlog**, terdapat variable **dataYangDiKirim**. Apa tujuan dari dibuatnya variable tersebut?

Jawaban:

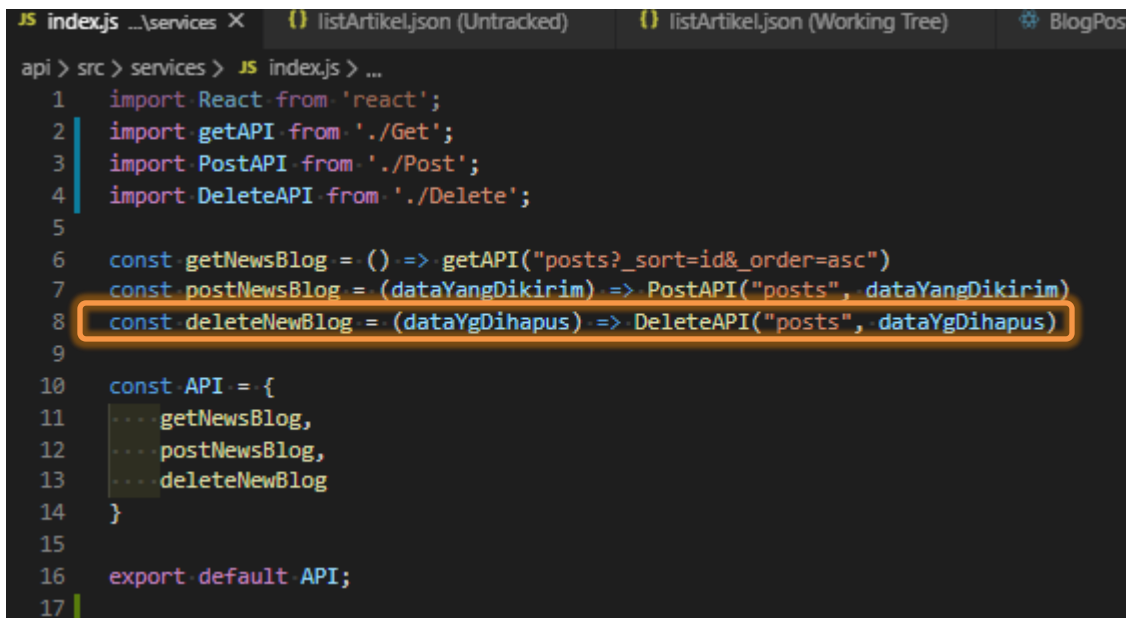
variable **dataYangDiKirim** berfungsi menerima data sementara yang kemudian akan di kirim ke **PostAPI** untuk dikirim lagi data kepada server API.

Praktikum 3

Global API Service DELETE

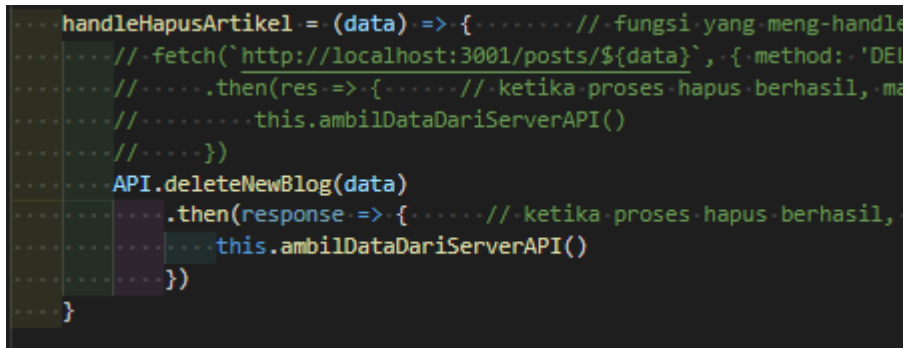
A. Langkah-langkah praktikum 3

1. Membuat fungsi untuk menampung action **DELETE** dari file **BlogPost.jsx** pada file **services/index.js** dan menginisialisasi fungsi tersebut



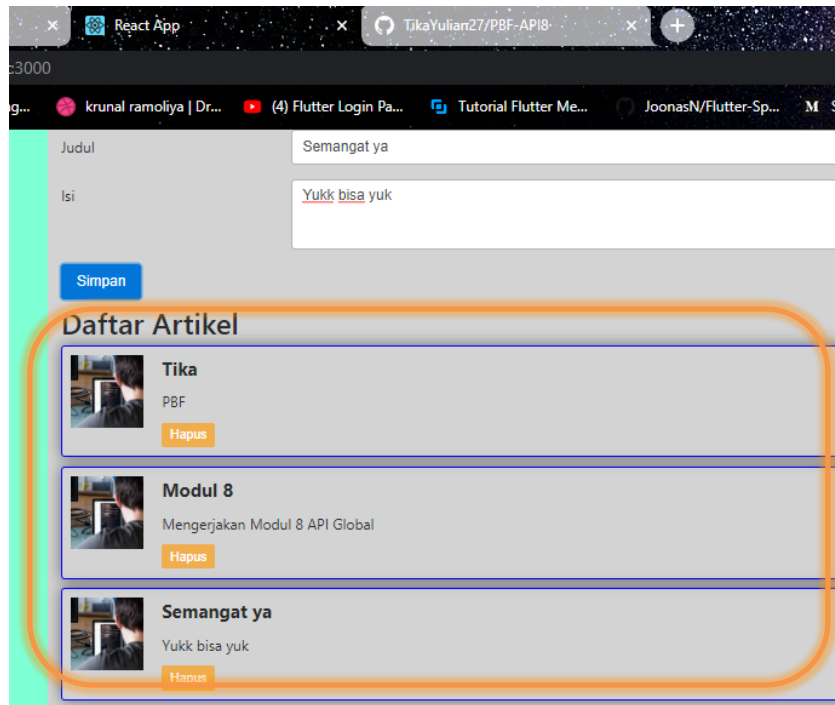
```
api > src > services > JS index.js > ...
1  import React from 'react';
2  import getAPI from './Get';
3  import PostAPI from './Post';
4  import DeleteAPI from './Delete';
5
6  const getNewsBlog = () => getAPI("posts?_sort=id&_order=asc")
7  const postNewsBlog = (dataYangDikirim) => PostAPI("posts", dataYangDikirim)
8  const deleteNewBlog = (dataYgDihapus) => DeleteAPI("posts", dataYgDihapus)
9
10 const API = {
11   ...getNewsBlog,
12   ...postNewsBlog,
13   ...deleteNewBlog
14 }
15
16 export default API;
17
```

2. Mengganti isi dari fungsi **handleHapusArtikel** pada file **BlogPost.jsx**

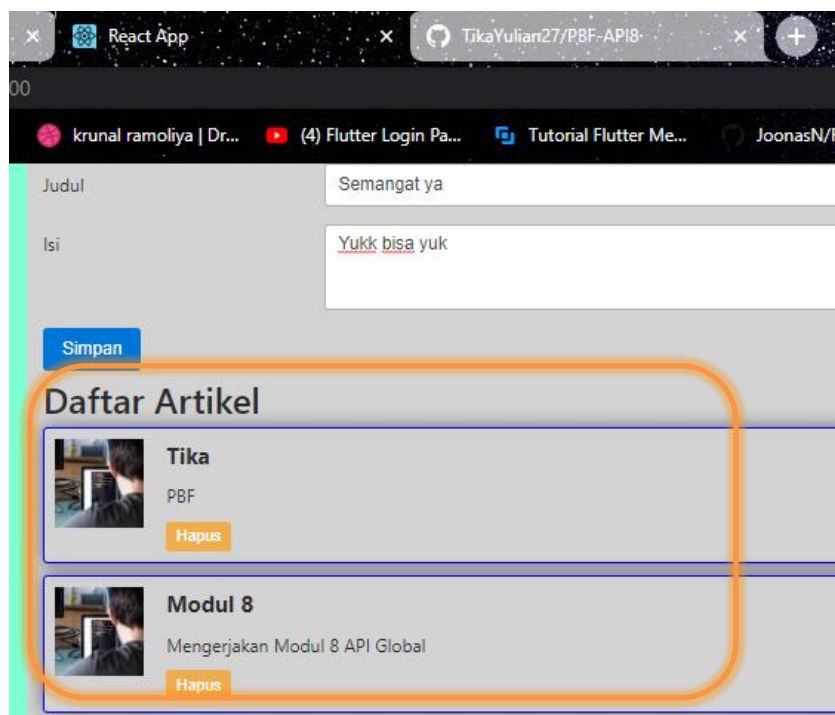


```
... handleHapusArtikel = (data) => { .....//fungsi yang meng-handle
... //fetch(`http://localhost:3001/posts/${data}`, {method: 'DEL
... //.....then(res => { .....//ketika proses hapus berhasil, ma
... //.....this.ambilDataDariServerAPI()
... //.....})
... API.deleteNewBlog(data)
... //.....then(response => { .....//ketika proses hapus berhasil,
... //.....this.ambilDataDariServerAPI()
... //.....})
... }
```

3. Selesai. Kode program pada **BlogPost.jsx** sedikit lebih simple dari seelumnya.
4. Menjalankan proses hapus artikel melalui browser.



Tampilan sebelum proses hapus artikel



Tampilan sesudah proses hapus artikel

5. Daftar artikel berkurang satu, setelah mengeklik button **“Hapus”**

B. Pertanyaan Praktikum 3

1. Perhatikan file **index.js**, apa tujuan dibuatnya fungsi **DeleteAPI** dan fungsi **deleteNewsBlog**?

Jawaban:

fungsi **DeleteAPI** adalah untuk merequest hapus data pada server API.

2. Pada fungsi **deleteNewsBlog**, terdapat variable **dataYangDiHapus**. Apa tujuan dari dibuatnya variable tersebut?

Jawaban:

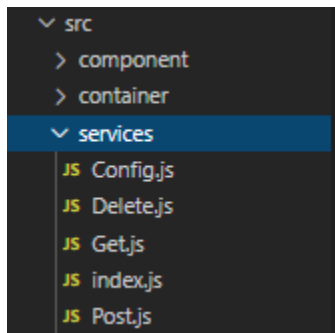
variable **dataYangDiHapus** berfungsi menghapus data sementara yang kemudian akan di kirim ke **DeleteAPI** untuk dihapus lagi data kepada server API.

Praktikum 4

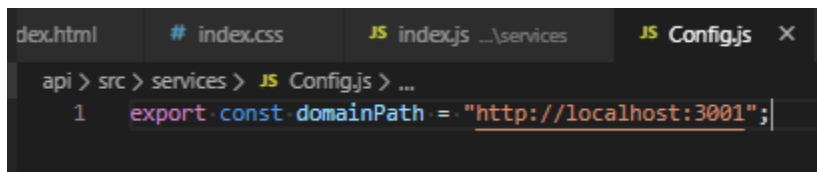
Manage Global API service

A. Langkah-langkah praktikum 4

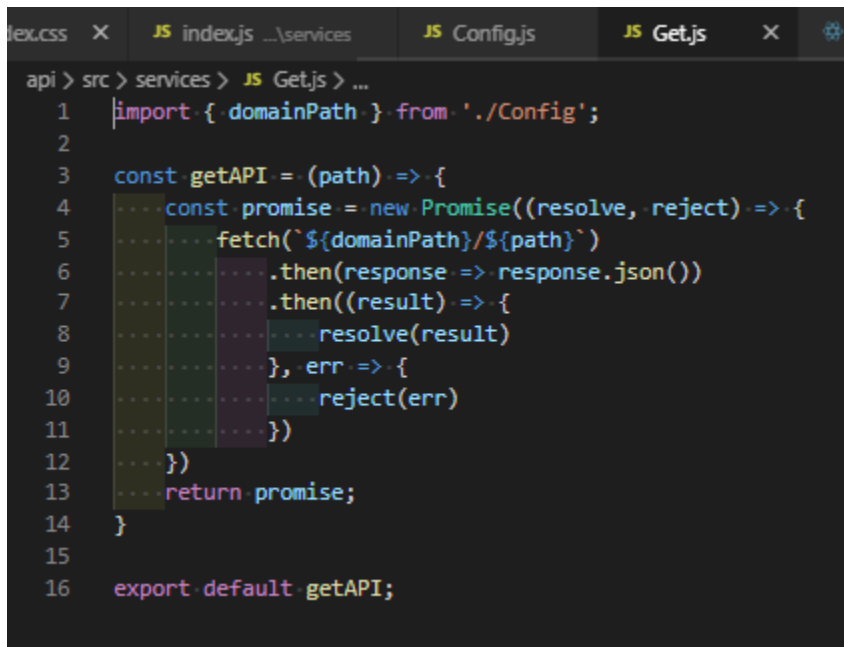
1. Membuat file **Config.js**, **Get.js**, **Post.js**, dan **Delete.js** dalam folder **services** seperti berikut:



2. Membuka file **Config.js** dan isikan kode seperti berikut:



3. Membuka file **Get.js** dan isikan kode seperti berikut:



4. Membuka file **Post.js** dan isikan kode seperti berikut:

```
JS index.js ...services X JS Config.js JS Get.js JS Post.js X
api > src > services > JS Post.js > default
1 import { domainPath } from './Config';
2
3 const PostAPI = (path, data) => {
4   const promise = 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    })
13     .then((result) => {
14       resolve(result)
15     }, err => {
16       reject(err)
17     })
18   })
19   return promise;
20 }
21
22 export default PostAPI;
```

5. Membuka file **Delete.js** dan isikan kode seperti berikut:

```
api > src > services > JS Delete.js > ...
1 import { domainPath } from './Config';
2
3 const DeleteAPI = (path, data) => {
4   const promise = new Promise((resolve, reject) => {
5     fetch(`${domainPath}/${path}/${data}`, {
6       method: "DELETE"
7     })
8     .then((result) => {
9       resolve(result)
10    }, err => {
11      reject(err)
12    })
13  })
14  return promise;
15 }
16
17 export default DeleteAPI;
```

6. Merubah **index.js** menjadi seperti berikut:

```

JS index.js ...services X {} listArtikel.json (Untracked) {} listArtikel.json (Working Tree) BlogPo
api > src > services > JS index.js > ...
1 import React from 'react';
2 import getAPI from './Get';
3 import PostAPI from './Post';
4 import DeleteAPI from './Delete';
5
6 const getNewsBlog = () => getAPI("posts?_sort=id&_order=asc")
7 const postNewsBlog = (dataYangDikirim) => PostAPI("posts", dataYangDikirim)
8 const deleteNewBlog = (dataYgDihapus) => DeleteAPI("posts", dataYgDihapus)
9
10 const API = {
11   ...getNewsBlog,
12   ...postNewsBlog,
13   ...deleteNewBlog
14 }
15
16 export default API;
17

```

7. Final Result

React App

TikaYulian27/PBF-API8

00

krunal ramoliya | Dr...

(4) Flutter Login Pa...

Tutorial Flutter Me...

JoonasN/Flutter-Sp

Judul


Global API

Isi

Praktikum 4

Simpan


Daftar Artikel



Tika

PBF


Hapus



Modul 8

Mengerjakan Modul 8 API Global

Hapus



Global API

Praktikum 4

Hapus

B. Pertanyaan praktikum 4

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

Jawaban:

Membuat ketiga file tersebut yang saling berintegrasi dengan js