

MODUL PRAKTIKUM

PEMROGRAMAN BERBASIS FRAMEWORK

Modul 9

* Global API service (GET)
* Global API Service (POST)
* Global API Service (DELETE)

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

Global API

Pada Modul 4 yang dulu kita pelajari mengenai bagaimana kita berinteraksi dengan API. Kita dihadapkan akan bagaimana cara kita mengelola banyak daftar alamat API yang disediakan dan mengumpulkannya menjadi satu kelompok. Sehingga kita dapat menambah, mengurangi, ataupun memperbaiki daftar API pada satu tempat.

Salah satu caranya adalah dengan mengelola/mengatur pemanggilan API secara GLOBAL, sehingga kita tidak perlu melakukan “***coding ulang***” API yang memiliki karakter yang sama. Contoh seperti saat kita memanggil API (GET) untuk mendapatkan *list* Artikel, *list* komentar, dan *list* pengguna. API untuk ketiga proses tersebut sebenarnya sama, yang membedakan hanya alamat url API yang dituju, sehingga kita bisa menaruh list API pada satu tempat untuk mempermudah dalam mengelolanya.

Pada praktikum pembahasan Global API ini, kita menggunakan bahan dari materi yang sebelumnya yaitu Modul 4 – Interaksi dengan API. Sehingga kita dapat memahami perbedaan antara Modul 4 yang hanya melakukan interaksi dengan API dengan modul ini (Modul 8) yang mengelola daftar list API kedalam satu tempat sehingga proses manajemen API lebih mudah dan cepat.

Keuntungan menggunakan Global API Bagi Para Developer Antara Lain:

1. Terpusat.

API yang dibuat atau yang akan dipanggil terpusat pada satu tempat.

1. Kustomisasi

Mudah dan effisien saat melakukan kustomisasi maupun perbaikan alamat API.

1. Mudah di manage

Berhubung list API berada pada satu tempat, sehingga proses manage daftar API baik penambahan API, pengurangan, maupun memperbaiki API (perubahan alamat url API) bisa dilakukan dengan mudah.

1. Terhindar dari proses yang sama (redundansi)

Berhubung daftar API dalam satu tempat sehingga kita bisa memantau meminimalisir proses yang sama atau pembuatan API yang sama persis.

# Praktikum 1 Global API service GET

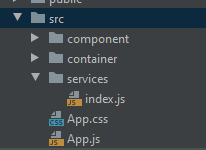
## Run Project dan Server Fake API

Sebelum memulai praktikum, kita mulai terlebih dahulu menyiapkan source hasil dari praktikum Modul 4 dan menjalankan server project dan server Fake API tersebut, yaitu buka 2 jendela *command prompt* (CMD) pada project yang sudah kita buat (pada Modul 4). Jendela pertama kita ketikkan perintah **npm start** untuk menjalankan local server project, dan jendela CMD kedua kita isikan perintah **json-server --watch listArtikel.json --port 3001** untuk menjalankan server fake API.

## Langkah Praktikum

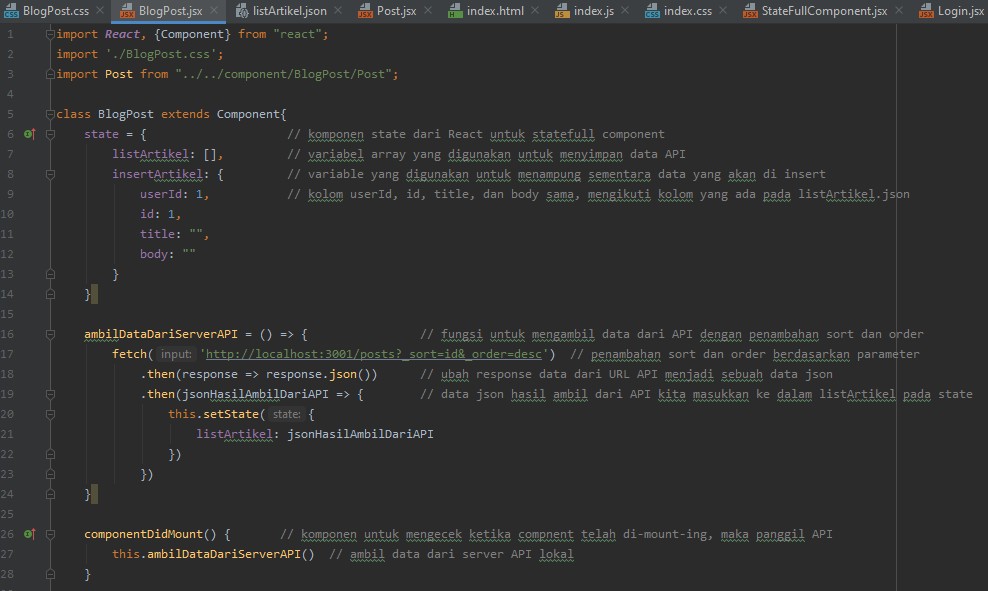
Dalam pembuatan Global Service API ini kita akan memerlukan tempat untuk mengumpulkan resource API yang ada, maka kita akan membuat tempat untuk menampung resource tersebut

* + 1. Buat folder baru bernama **“Services”** dalam folder **src**, kemudian buat file **index.js** seperti pada Gambar 1.1

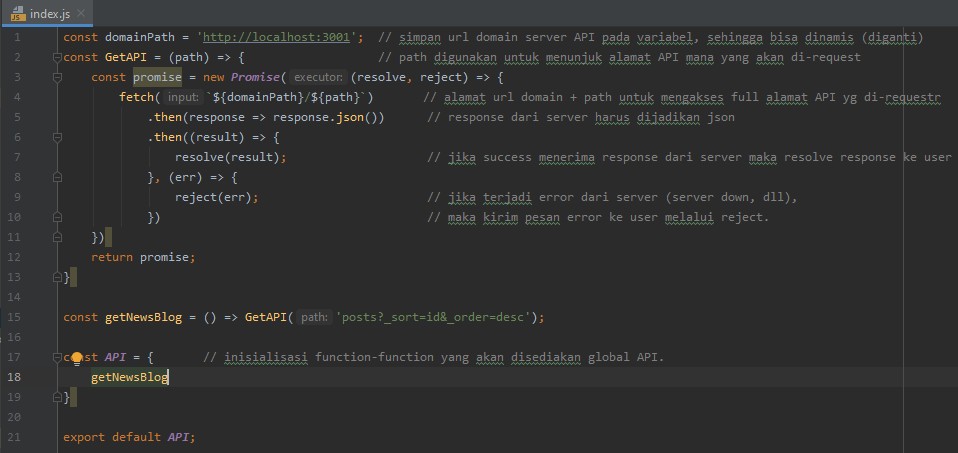


Gambar 1.1. Pembuatan folder dan file untuk Global Service API

* + 1. Buka file **BlogPost.jsx** pada folder container (*statefull component*) dan akan tampil kode program seperti Gambar 1.2.

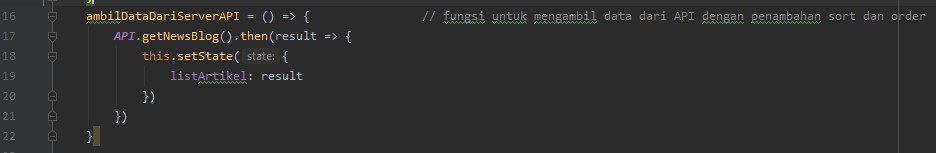


Gambar 1.2. Isi kode program BlogPost

* + 1. Pada fungsi **ambilDataDariServerAPI** (baris16) terdapat pemanggilan API GET untuk me- request data artikel. Proses dari fungsi inilah yang akan kita manage kedalam satu tempat yaitu **index.js**.
    2. Buka file **index.js** pada folder **service**, yang telah kita buat tadi dan tuliskan baris kode seperti Gambar 1.3 berikut

Gambar 1.3. kode program index.js

* + 1. Kembali ke file **BlogPost.js**. Ganti baris 17 sampai baris 23 menjadi seperti Gambar 1.4



Gambar 1.4. edit kode program BlogPost

* + 1. Simpan file **BlogPost.js** dan **index.js** tersebut, dan amati apa yang terjadi pada browser kalian.

## Pertanyaan Praktikum 1

a. Perhatikan file **index.js**, apa tujuan dibuatnya fungsi **GetAPI** pada baris 2 dan fungsi

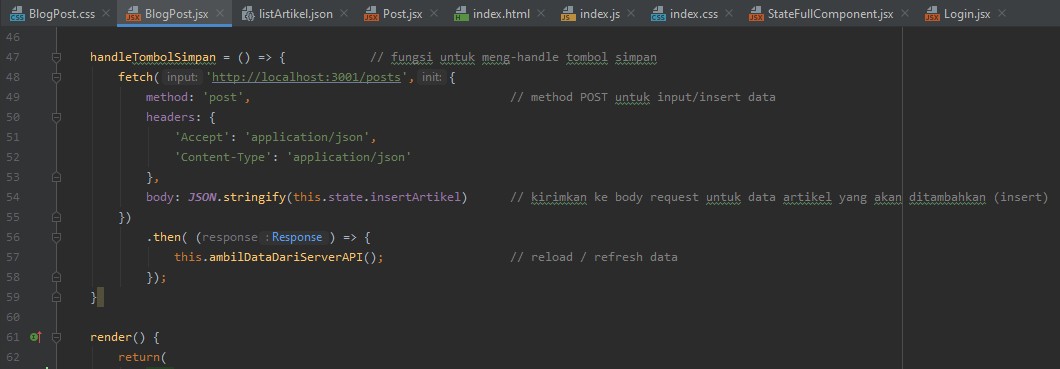
**getNewsBlog** pada baris 16?

# Praktikum 2 Global API service POST

## Langkah Praktikum 2

Seperti pada langkah praktikum 1 dimana kita me-*manage* API GET untuk mendapatkan data dari server. Sekarang kita akan me-*manage* API POST untuk mengirimkan data kepada server API. Langkah-langkahnya adalah

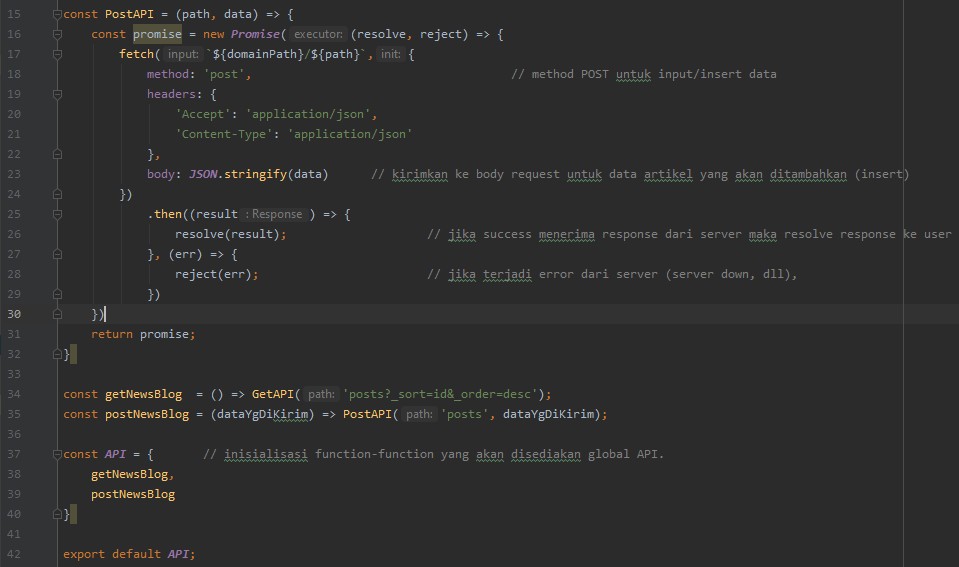
* + 1. Kita perhatikan pada fungsi **handleTombolSimpan** pada file **BlogPost.js**. Bagian inilah yang selanjutnya kita kelola agar bisa di-manage.



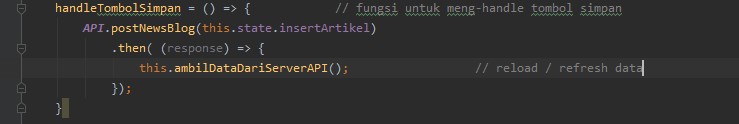
Gambar 2.1. Kode program BlogPost fungsi handleTombolSimpan

* + 1. Buatlah fungsi untuk menampung action POST dari file **BlogPost.js** pada file

**services/index.js** dan **inisialisasi** fungsi tersebut seperti pada Gambar 2.2



Gambar 2.2. Kode program index.js untuk handle POST

* + 1. Selanjutnya pindah ke file **BlogPost.js** dan ganti isi dari fungsi **handleTombolSimpan** menjadi seperti Gambar 2.3

Gambar 2.3. Kode program pengganti handleTombolSimpan

* + 1. Selesai. Kode program pada BlogPost.js sedikit lebih simple dari seelumnya.
    2. Silahkan kalian jalankan proses input artikel melalui browser dan amati apa yang terjadi.

## Pertanyaan Praktikum 2

* + 1. Perhatikan file **index.js**, apa tujuan dibuatnya fungsi **PostAPI** dan fungsi **postNewsBlog**?
    2. Pada fungsi **postNewsBlog**, terdapat variable **dataYangDiKirim**. Apa tujuan dari dibuatnya variable tersebut?

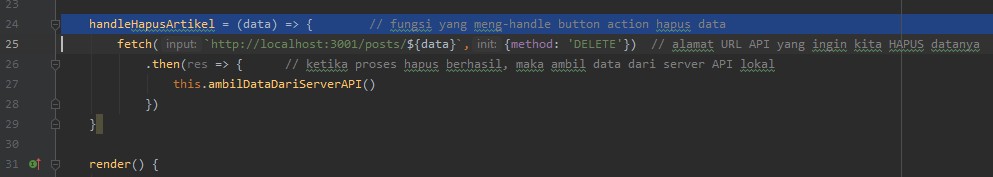
# Praktikum 3

**Global API service DELETE**

## Langkah Praktikum 3

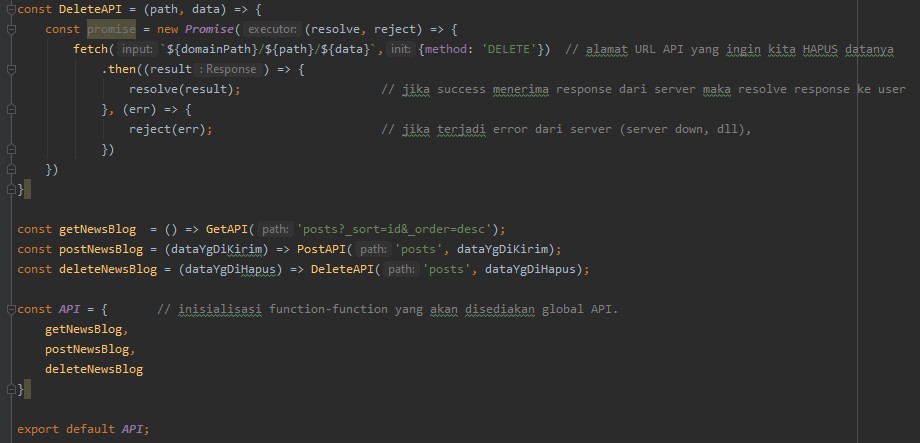
Seperti pada langkah praktikum 2 dimana kita me-*manage* API GET dan POST untuk mendapatkan data dari server. Sekarang kita akan me-*manage* API DELETE untuk request hapus data pada server API. Langkah-langkahnya adalah

* + 1. Kita perhatikan pada fungsi **handleHapusArtikel** pada file **BlogPost.jsx**. Bagian inilah yang selanjutnya kita kelola agar bisa di-manage.

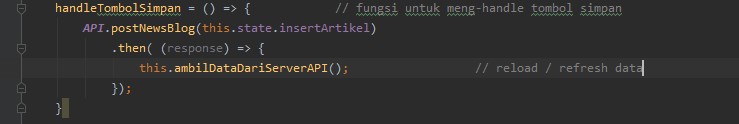


Gambar 3.1. Kode program BlogPost fungsi handleHapusArtikel

* + 1. Buatlah fungsi untuk menampung action DELETE dari file **BlogPost.jsx** pada file

**services/index.js** dan **inisialisasi** fungsi tersebut seperti pada Gambar 3.2

Gambar 3.2. Kode program index.js untuk handle DELETE

* + 1. Selanjutnya pindah ke file **BlogPost.jsx** dan ganti isi dari fungsi **handleHapusArtikel** menjadi seperti Gambar 3.3

Gambar 3.3. Kode program pengganti handleTombolSimpan

* + 1. Selesai. Kode program pada **BlogPost.jsx** sedikit lebih simple dari seelumnya.
    2. Silahkan kalian jalankan proses hapus artikel melalui browser dan amati apa yang terjadi.

## Pertanyaan Praktikum 3

* + 1. Perhatikan file **index.js**, apa tujuan dibuatnya fungsi **DeleteAPI** dan fungsi **deleteNewsBlog**?
    2. Pada fungsi **deleteNewsBlog**, terdapat variable **dataYangDiHapus**. Apa tujuan dari dibuatnya variable tersebut?

# Praktikum 4 Manage Global API service

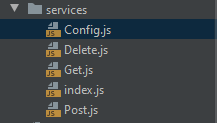
## Manage Global API

Pada global API yang telah kita lakukan pada praktikum 1, 2 dan 3. Kita mengetahui bahwa saat kita benar-benar melakukan coding untuk membuat API, kita akan dihadapkan pada banyak url API yang akan kita sediakan. Sehingga kita butuh manage lagi Global API untuk lebih teratur. Salah satu cara untuk me-*manage* Global API adalah dengan memisahkan API berdasarkan action-nya (GET, POST, DELETE). Sehingga, missal saat terjadi error pada DELETE API, kita

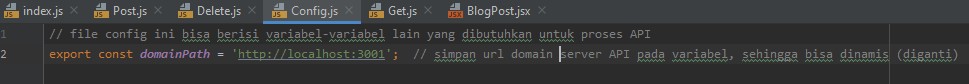
cukup akan membuka Global API DELETE yang akan kita perbaiki.

## Langkah Praktikum 4

Langkah-langkah yang dilakukan pada praktikum 4 ini adalah

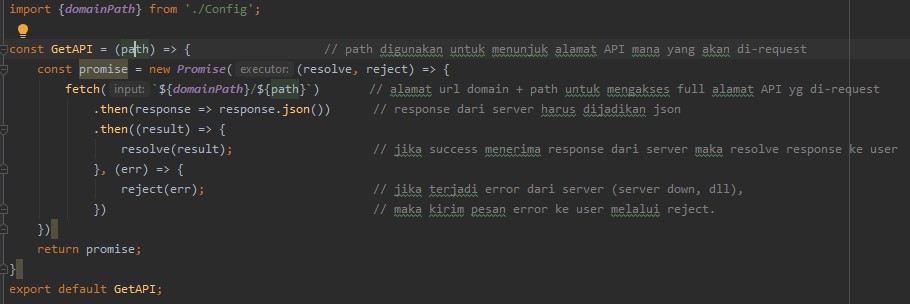
* + 1. Buatlah file **Config.js**, **Get.js**, **Post.js**, dan **Delete.js** dalam folder **services** seperti pada Gambar 4.1

Gambar 4.1. Buat file

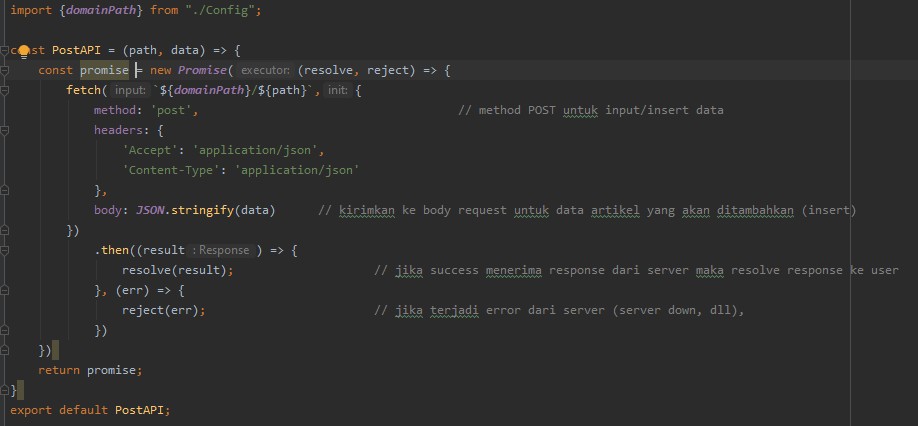
* + 1. Buka **Config.js** dan isikan kode seperti Gambar 4.2

Gambar 4.2. Kode program config.js

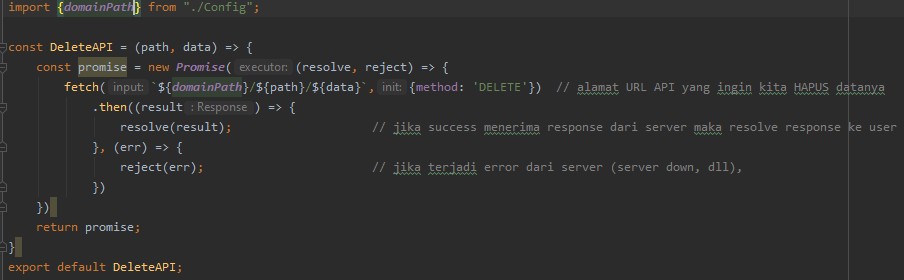
* + 1. Buka **Get.js** dan isikan kode seperti Gambar 4.3



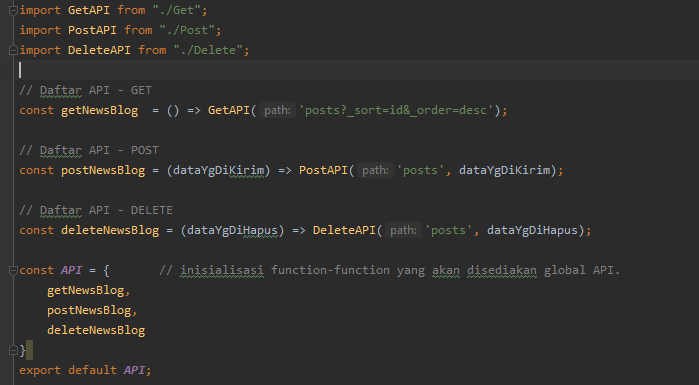
Gambar 4.3. Kode program Get.js

* + 1. Buka **Post.js** dan isikan kode seperti Gambar 4.4

Gambar 4.4. Kode program Post.js

* + 1. Buka **Delete.js** dan isikan kode seperti Gambar 4.5

Gambar 4.5. Kode program Delete.js

* + 1. Pada index.js kita edit menjadi seperti Gambar 4.6
    2. Amati apa yang terjadi

Gambar 4.6. Kode program index.js

## Pertanyaan Praktikum 4

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

# TUGAS PRAKTIKUM

Ubahlah **Tugas pada Modul 4** yang sudah kalian buat dengan memanfaatkan Global API seperti praktikum yang kita lakukan pada Modul 8 ini.

**~ ~ ~ Selamat Mengerjakan ~ ~ ~**