

Menciptakan dan menggunakan Rest API

Praktikum ini membantu untuk membuat operasi CRUD REST api tanpa menggunakan Framework REST. Anda dapat mendefinisikan struktur apa saja dan disesuaikan sesuai kebutuhan Anda, Anda dapat menetapkan aturan untuk framework REST API dan mudah mengaksesnya. Tidak wajib untuk memiliki framework kerja untuk membangun API REST Anda menggunakan php. Anda dapat membuat api RESTnya menggunakan kode PHP inti.

Ada beberapa framework REST menggunakan untuk PHP adalah sebagai berikut.

- Laravel
- Phalcon
- Lumen
- Slim
- Silex

Mari kita mulai membuat contoh REST API, saya akan membuat folder dengan nama '**restful-api**'. Folder ini akan menyimpan semua file yang diperlukan untuk praktikum ini. mahasiswa akan membuat Rest API untuk modul karyawan yang akan memiliki HTTP Get untuk permintaan untuk mengambil catatan dari mysql, Post untuk tambahkan catatan ke MySQL, Put untuk perbarui catatan ke mysql dan Delete untuk hapus record data yang ditentukan dari mysql. Detail dari REST API adalah sebagai berikut:

Route	Method	Type	Posted JSON	Description
/employees	GET	JSON	-	Dapatkan semua data karyawan
/employees/{id}	GET	JSON	-	Get a single employee data
/employees	POST	JSON	<pre>{ "employee_name": "Moses", "employee_age": "19", "employee_salary" : "9000000" }</pre>	Masukkan catatan karyawan baru ke dalam basis data
/employees/13319020	PUT	JSON	<pre>{ "employee_name": "Moses", "employee_age": "27", "employee_salary" : "12000000" }</pre>	Perbarui catatan karyawan ke dalam basis data
/employees	DELETE	JSON	<pre>{ "id" : 13319020 }</pre>	Hapus catatan karyawan tertentu dari database

Mahasiswa akan menggunakan file-file berikut untuk praktikum REST API ini:

index.php: File ini adalah file entri, File ini mencegah navigasi file folder.

connection.php: File ini akan digunakan untuk memiliki string koneksi mysql.

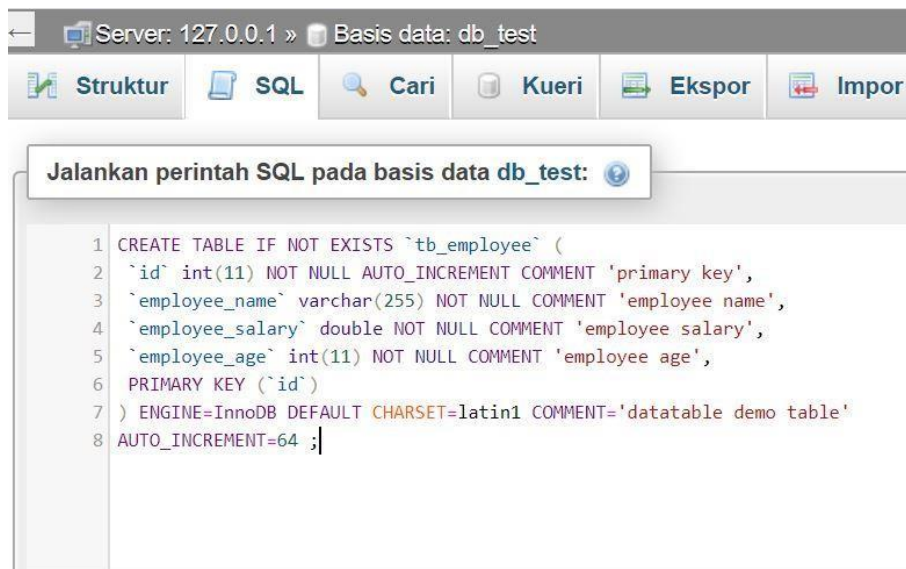
v1 / employee.php: File ini akan berisi semua titik akhir istirahat dan metode tindakan.

v1 / .htaccess: File ini digunakan untuk tujuan perutean/routing ke endpoint.

Skema Tabel Pegawai

Buat '**db_test**' database MySQL, jika Anda sudah memiliki database maka jalankan sql di bawah ke dalam window-query SQL database.

```
CREATE TABLE IF NOT EXISTS `tb_employee` (  
  `id` int(11) NOT NULL AUTO_INCREMENT COMMENT 'primary key',  
  `employee_name` varchar(255) NOT NULL COMMENT 'employee name',  
  `employee_salary` double NOT NULL COMMENT 'employee salary',  
  `employee_age` int(11) NOT NULL COMMENT 'employee age',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='datatable demo table'  
AUTO_INCREMENT=64 ;
```



Selanjutnya isi setiap kolom pada tabel anda dengan dummy data/data percobaan dengan klik menu **tambahkan** dan isilah sesuai keinginan anda. Ini terlihat pada gambar berikut:

Server: 127.0.0.1 » Basis data: db_test » Tabel: tb_employee "datatable demo table"

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor

✓ Menampilkan baris 0 - 3 (total 4, Pencarian dilakukan dalam 0,0007 detik.)

`SELECT * FROM `tb_employee``

☐ Profil [Edit dikota]

☐ Tampilkan semua | Jumlah baris: 25 Saring baris: Cari di tabel ini Sort by ke

+ Opsi

	id	employee_name	employee_salary	employee_age
	primary key	employee name	employee salary	employee age
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	1	rivaldo jelek	5000000	49
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	13319016	jansutris	3000000	23
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	13319017	elisabeth	4000000	34
<input type="checkbox"/> Ubah <input type="checkbox"/> Salin <input type="checkbox"/> Hapus	13319018	Arta Hutapea	9000000	34

☐ Pilih Semua Dengan pilihan: ☐ Ubah ☐ Salin ☐ Hapus ☐ Ekspor

Jadi, saya sudah punya 4 data dummy yang terisi pada masing-masing kolom.

Koneksi Database MySQL Dengan PHP Menggunakan Mysqli

Buatlah file **connection.php** untuk menghubungkan MySQL dengan php menggunakan metode **mysqli_connect()**. Tambahkan kode di bawah ini ke file tersebut.

```
<?php
    $server = "localhost";
    $user = "root";
    $password = "";
    $nama_database = "db_test";

    $db = mysqli_connect($server, $user, $password, $nama_database);

    if(!$db)
    {
        die("Gagal terhubung dengan database: " . mysqli_connect_error());
    }
?>
```

Catatan:

Anda perlu mengganti **nama basis data**, **nama pengguna db**, **kata sandi db** dan **nama host** sesuai konfigurasi database anda.

.Htaccess Rewriterule For Clean URLs

kita akan membuat file **.htaccess** v1 / folder dan menulis beberapa aturan untuk REST API dengan URL yang rapi. kita akan menambahkan aturan berikut.

```
RewriteEngine On # Turn on the rewriting engine
RewriteRule ^employees/?$ employees.php [NC,L]
RewriteRule ^employees/([0-9]+)/?$ employees.php?id=$1 [NC,L]
```

REST API Untuk Mengambil Semua Record Dari MySQL

kita akan membuat Permintaan HTTP Request dengan jenis GET untuk mengakses semua record/data karyawan dari MySQL. kita akan menggunakan query MySQL untuk mengambil data dari tabel karyawan dan mengirim array data JSON ke klien sebagai response object.

Sekarang kita akan membuat file **employee.php** didalam folder **v1** dan menambahkan file koneksi MySQL untuk mengakses data tabel database.

```
<?php
// Connect to database
include("../connection.php");
$connection = $db;
$request_method=$_SERVER["REQUEST_METHOD"];
```

Catatan:

Kita akan menulis kode program pada file **employee.php** secara bertahap untuk tiap fungsi. Kode program yang lengkap akan saya cantumkan pada akhir modul praktikum ini.

Saya juga menggunakan metode **\$_SERVER** untuk mengakses Informasi action, seperti rest request untuk menambah, mengedit, atau menghapus.

kita akan membuat metode **SWITCH-CASE** pada php untuk mengakses *method* yang tepat terhadap *rest request*.

```
switch($request_method)
{
    case 'GET':
        // Retrive Products
        if(!empty($_GET["id"]))
        {
            $id=intval($_GET["id"]);
            get_employees($id);
        }
        else
        {
            get_employees();
        }
        break;
    default:
```

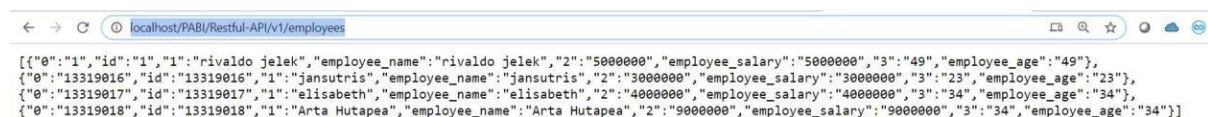
```
// Invalid Request Method
header("HTTP/1.0 405 Method Not Allowed");
break;
}
```

kita telah membuat permintaan GET untuk mengambil semua data karyawan dari database mysql, untuk data karyawan tunggal kita melakukan *passing* id karyawan pada argument fungsi. kita akan membuat metode/function **get_employees()** seperti di bawah ini.

```
function get_employees()
{
    global $connection;
    $query="SELECT * FROM tb_employee";
    $response=array();
    $result=mysqli_query($connection, $query);
    while($row=mysqli_fetch_array($result))
    {
        $response[]=$row;
    }
    header('Content-Type: application/json');
    echo json_encode($response);
}
```

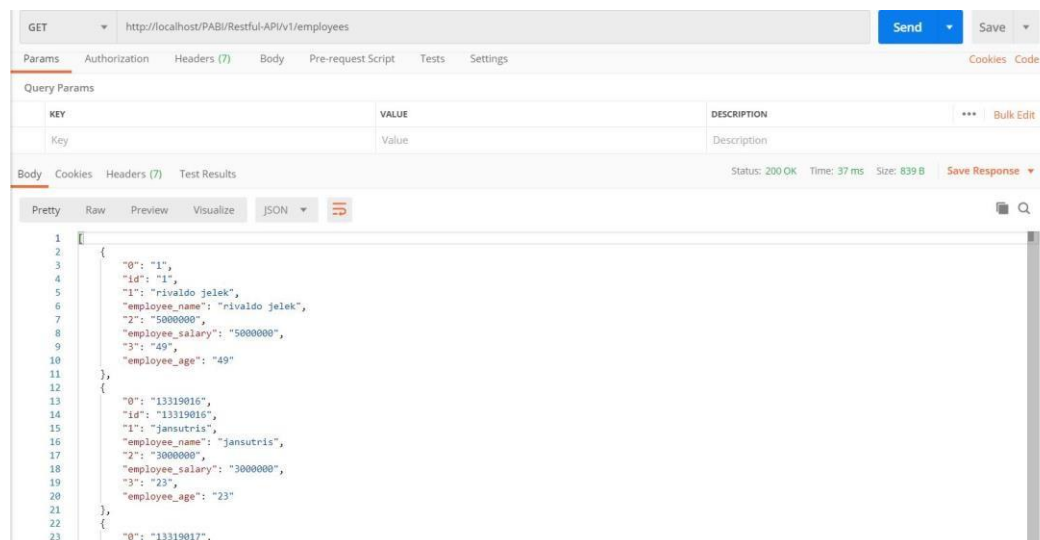
metode **mysqli_query ()** mengambil data dari tabel karyawan MySQL dan disimpan sebagai variabel ke dalam variabel '**result**'. Metode **json_encode ()** mengubah array data menjadi string json.

Sekarang akses <http://localhost/PABI/Restful-API/v1/employees> . Ini adalah REST API url dan anda akan mendapatkan semua catatan karyawan dari tabel karyawan mysql.



```
[{"0": "1", "id": "1", "1": "rivaldo jelek", "employee_name": "rivaldo jelek", "2": "5000000", "employee_salary": "5000000", "3": "49", "employee_age": "49"}, {"0": "13319016", "id": "13319016", "1": "jansutris", "employee_name": "jansutris", "2": "3000000", "employee_salary": "3000000", "3": "23", "employee_age": "23"}, {"0": "13319017", "id": "13319017", "1": "elisabeth", "employee_name": "elisabeth", "2": "4000000", "employee_salary": "4000000", "3": "34", "employee_age": "34"}, {"0": "13319018", "id": "13319018", "1": "Arta Hutapea", "employee_name": "Arta Hutapea", "2": "9000000", "employee_salary": "9000000", "3": "34", "employee_age": "34"}]
```

Pengujian menggunakan tools Postman



GET http://localhost/PABI/Restful-API/v1/employees

Send Save

Params Authorization Headers (7) Body Pre-request Script Tests Settings

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (7) Test Results

Status: 200 OK Time: 37 ms Size: 839 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   {
3     "0": "1",
4     "id": "1",
5     "1": "rivaldo jelek",
6     "employee_name": "rivaldo jelek",
7     "2": "5000000",
8     "employee_salary": "5000000",
9     "3": "49",
10    "employee_age": "49"
11  },
12  {
13    "0": "13319016",
14    "id": "13319016",
15    "1": "jansutris",
16    "employee_name": "jansutris",
17    "2": "3000000",
18    "employee_salary": "3000000",
19    "3": "23",
20    "employee_age": "23"
21  },
22  {
23    "0": "13319017",
24    "id": "13319017",
25    "1": "elisabeth",
26    "employee_name": "elisabeth",
27    "2": "4000000",
28    "employee_salary": "4000000",
29    "3": "34",
30    "employee_age": "34"
31  },
32  {
33    "0": "13319018",
34    "id": "13319018",
35    "1": "Arta Hutapea",
36    "employee_name": "Arta Hutapea",
37    "2": "9000000",
38    "employee_salary": "9000000",
39    "3": "34",
40    "employee_age": "34"
41  }
42 }
```

Jika kita menggunakan HTTP request GET untuk mengambil semua data dari database lewat **URI/endpoint** rest server dan kita klik **SEND**, maka akan menghasilkan output berupa data yang sama. Artinya, REST-API dengan fungsi GET semua record sudah berhasil kita ciptakan dan kita uji menggunakan postman sebagai REST client.

Restful API Untuk Mendapatkan Single record/data tunggal Dari MySQL Menggunakan PHP

kita akan membuat rest request dengan jenis HTTP GET untuk mengakses single/satu record karyawan dari database MySQL menggunakan php. sama seperti pemanggilan REST API sebelumnya dengan melakukan passing/melewatkan employee id sebagai argumen/parameter fungsi.

```
function get_All_employees($id=0)
{
    global $connection;
    $query="SELECT * FROM tb_employee";
    if($id != 0)
    {
        $query.=" WHERE id=".$id." LIMIT 1";
    }
    $response=array();
    $result=mysqli_query($connection, $query);
    while($row=mysqli_fetch_array($result))
    {
        $response[]=$row;
    }
    header('Content-Type: application/json');
    echo json_encode($response);
}
```

Sekarang akses <http://localhost/PABI/Restful-API/v1/employees/1> sebagai REST API endpoint dan anda akan mendapatkan satu record employee/karyawan dari tabel **tb_employee** dengan **id=1**.

← → ↺

[{"0":"1","id":"1","1":"rivaldo jelek","employee_name":"rivaldo jelek","2":"5000000","employee_salary":"5000000","3":"49","employee_age":"49"}]

Pengujian menggunakan tools Postman

GET <http://localhost/PABI/Restful-API/v1/employees/1> Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (7) Test Results Status: 200 OK Time: 23 ms Size: 389 B Save

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "0": "1",
4     "id": "1",
5     "1": "rivaldo jelek",
6     "employee_name": "rivaldo jelek",
7     "2": "5000000",
8     "employee_salary": "5000000",
9     "3": "49",
10    "employee_age": "49"
11  }
12 ]
```

Rest Api Untuk Membuat Record Baru Ke MySQL

Kita akan membuat Rest Api baru untuk memasukkan Record karyawan baru ke dalam MySQL menggunakan php. Saya akan membuat REST Request dengan jenis POST karena kita akan memposting beberapa data JSON ke server php.

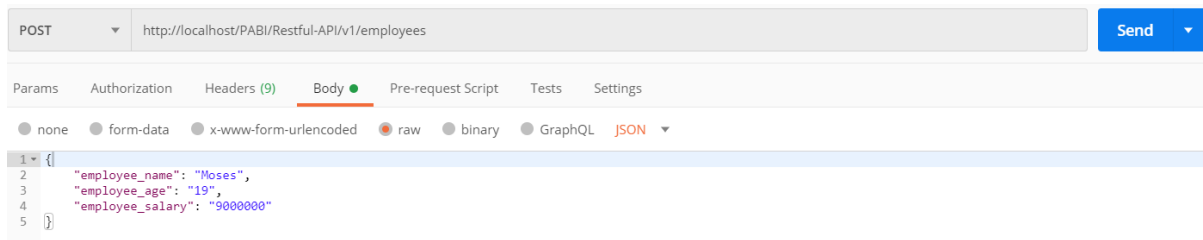
kita akan menambahkan **CASE** baru ke dalam metode **SWITCH** seperti di bawah ini.

```
case 'POST':  
    // Insert Product  
    insert_employee();  
    break;
```

Now we will create **insert_employee()** method into **employees.php** file.

```
<?  
function insert_employee()  
{  
    global $connection;  
  
    $data = json_decode(file_get_contents('php://input'), true);  
    $employee_name=$data["employee_name"];  
    $employee_salary=$data["employee_salary"];  
    $employee_age=$data["employee_age"];  
    echo $query="INSERT INTO tb_employee SET employee_name='".$employee_name."  
' , employee_salary='".$employee_salary."', employee_age='".$employee_age.'";  
    if(mysqli_query($connection, $query))  
    {  
        $response=array(  
            'status' => 1,  
            'status_message' =>'Employee Added Successfully.'  
        );  
    }  
    else  
    {  
        $response=array(  
            'status' => 0,  
            'status_message' =>'Employee Addition Failed.'  
        );  
    }  
    header('Content-Type: application/json');  
    echo json_encode($response);  
}  
?>
```

Pengujian menggunakan tools Postman

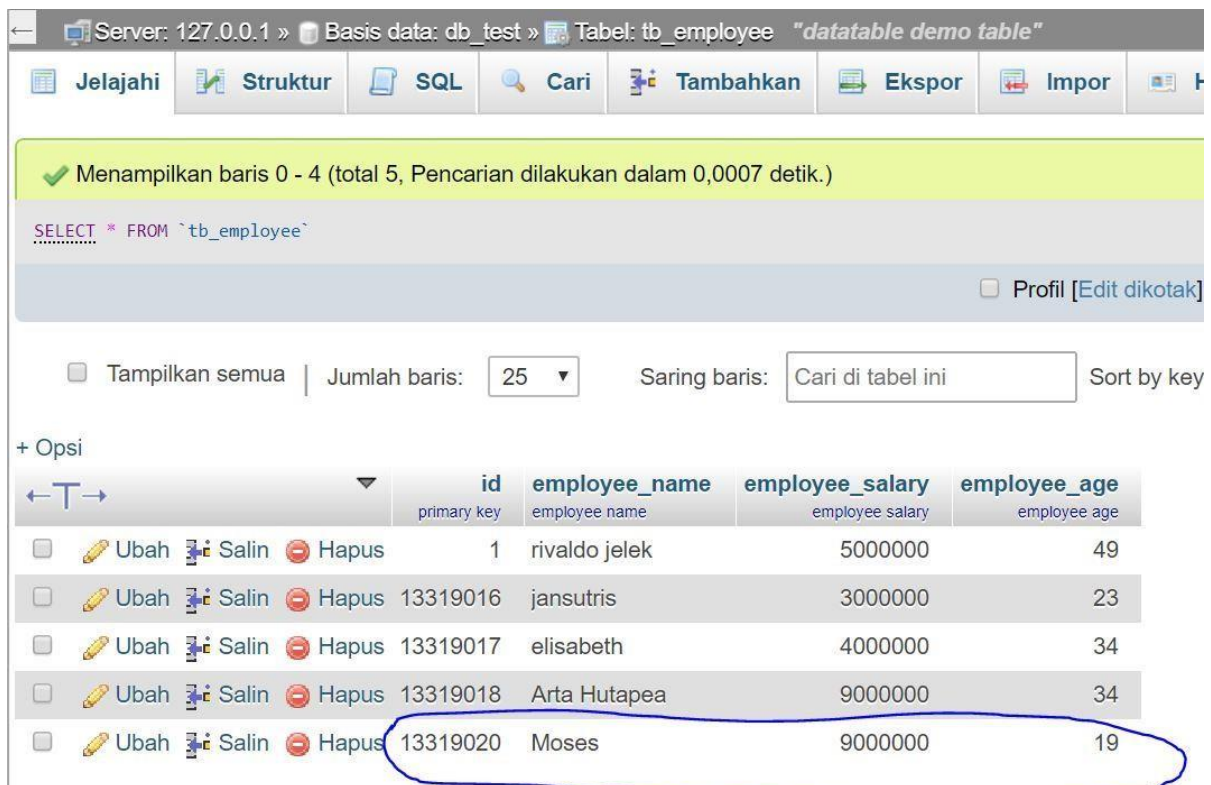


Jika anda kirimkan request POST ke endpoint rest-server yaitu request yang menambahkan sebuah data bar ke database dengan klik tombol **Send**, maka Response body akan menghasilkan output sebagai berikut:



Perubahan data pada database

Silahkan anda cek database, maka anda akan melihat data baru telah berhasil ditambahkan di database anda:



Rest Api Untuk Update Record Ke MySQL Menggunakan PHP

kita akan membuat jenis HTTP PUT Request Restful API baru untuk memperbarui data ke dalam basis data MySQL. Kita harus *passing* id karyawan sebagai parameter yang akan digunakan untuk memperbarui catatan ke tabel karyawan.

Kita akan menambahkan **CASE** baru ke dalam metode **SWITCH** seperti di bawah ini.

```
case 'PUT':  
    // Update Product  
    $id=intval($_GET["id"]);  
    update_employee($id);  
    break;
```

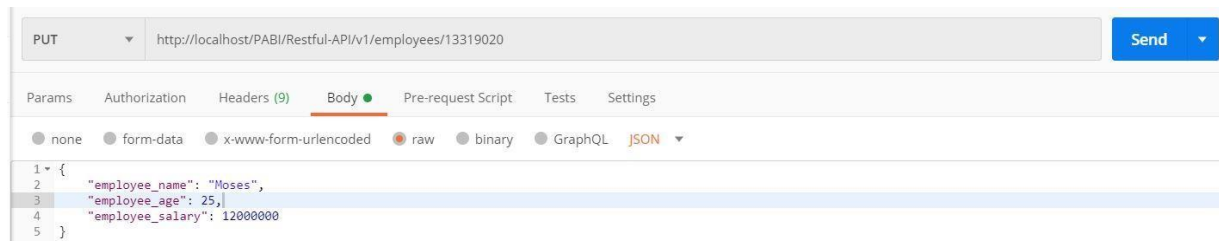
Sekarang kita akan membuat metode **update_employee ()** ke dalam file **employee.php**. Kita akan menggunakan **employee_id** yang ingin kita perbarui dan yang kedua adalah data yang diperbarui dalam format json.

```
function update_employee($id)  
{  
    global $connection;  
    $post_vars = json_decode(file_get_contents("php://input"),true);  
    $employee_name=$post_vars["employee_name"];  
    $employee_salary=$post_vars["employee_salary"];  
    $employee_age=$post_vars["employee_age"];  
    $query="UPDATE tb_employee SET employee_name='".$employee_name."', employee  
e_salary='".$employee_salary."', employee_age='".$employee_age.'" WHERE id=".$  
id;  
    if(mysqli_query($connection, $query))  
    {  
        $response=array(  
            'status' => 1,  
            'status_message' =>'Employee Updated Successfully.'  
        );  
    }  
    else  
    {  
        $response=array(  
            'status' => 0,  
            'status_message' =>'Employee Updation Failed.'  
        );  
    }  
    header('Content-Type: application/json');  
    echo json_encode($response);  
}
```

Kami telah mengambil data **employee_id** sebagai parameter dan json untuk memperbarui record ke dalam tabel mysql, jika semuanya benar kemudian mengirim '**updated message**' jika tidak, mengirim **error message**.

Pengujian menggunakan tools Postman

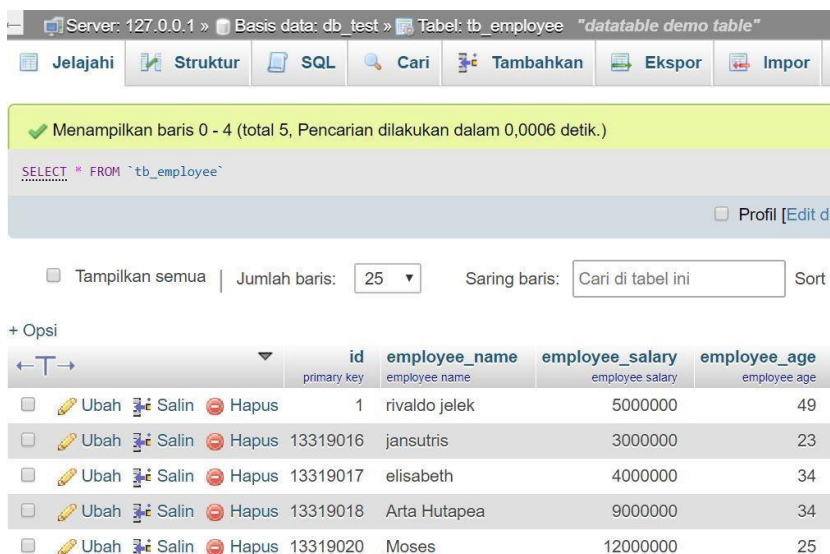
Misalnya moses sudah bekerja bertahun-tahun dan usia moses sekarang sudah 25 tahun dan karena moses adalah karyawan yang cerdas maka gaji moses juga sekarang sudah 12.000.000. Sehingga kita wajib melakukan update data moses (employee_age dan employee_salary) pada database. Kita akan menguji fungsi update tersebut pada data moses. Pilih method PUT dan ketikkan endpoint sesuai dengan **id moses = 13319020**.



Response Body



Perubahan data pada database



Rest API Untuk Menghapus Record Dari MySQL

kita akan membuat DELETE Type baru permintaan api restful menggunakan php untuk menghapus record karyawan dari database MySQL. Kami akan passing/melewatkan **employee_id** sebagai parameter yang ingin Anda hapus dari tabel karyawan.

Kami akan menambahkan **CASE** baru ke dalam metode **SWITCH** seperti di bawah ini

```
case 'DELETE':  
    // Delete Product  
    $id=intval($_GET["id"]);  
    delete_employee($id);  
    break;
```

Sekarang kita akan membuat metode **delete_employee ()** ke dalam file **employee.php**. Kita akan menggunakan **employee_id** yang ingin kita hapus record dari tabel **tb_employee**.

```
function delete_employee($id)  
{  
    global $connection;  
    $query="DELETE FROM tb_employee WHERE id=".$id;  
    if(mysqli_query($connection, $query))  
    {  
        $response=array(  
            'status' => 1,  
            'status_message' =>'Employee Deleted Successfully.'  
        );  
    }  
    else  
    {  
        $response=array(  
            'status' => 0,  
            'status_message' =>'Employee Deletion Failed.'  
        );  
    }  
    header('Content-Type: application/json');  
    echo json_encode($response);  
}
```

Kode program yang lengkap di **employee.php** adalah berikut.

```
<?php  
// Connect to database  
include("../connection.php");  
$connection = $db;  
$request_method=$_SERVER["REQUEST_METHOD"];  
  
switch($request_method)  
{
```

```

        case 'GET':
            // Retrive Products
            if(!empty($_GET["id"]))
            {
                $id=intval($_GET["id"]);
                get_single_employees($id);
            }
            else
            {
                get_All_employees();
            }
            break;

        case 'POST':
            // Insert Product
            insert_employee();
            break;

        case 'PUT':
            // Update Product
            $id=intval($_GET["id"]);
            update_employee($id);
            break;

        case 'DELETE':
            // Delete Product
            $id=intval($_GET["id"]);
            delete_employee($id);
            break;

        default:
            // Invalid Request Method
            header("HTTP/1.0 405 Method Not Allowed");
            break;
    }

function get_All_employees()
{
    global $connection;
    $query="SELECT * FROM tb_employee";
    $response=array();
    $result=mysqli_query($connection, $query);

    while($row=mysqli_fetch_array($result))
    {
        $response[]=$row;
    }
    header('Content-Type: application/json');
}

```

```

        echo json_encode($response);
    }

function get_single_employees($id=0)
{
    global $connection;
    $query="SELECT * FROM tb_employee";
    if($id != 0)
    {
        $query.=" WHERE id=".$id." LIMIT 1";
    }
    $response=array();
    $result=mysqli_query($connection, $query);
    while($row=mysqli_fetch_array($result))
    {
        $response[]=$row;
    }
    header('Content-Type: application/json');
    echo json_encode($response);
}

function insert_employee()
{
    global $connection;

    $data = json_decode(file_get_contents('php://input'), true);
    $employee_name=$data["employee_name"];
    $employee_salary=$data["employee_salary"];
    $employee_age=$data["employee_age"];
    echo $query="INSERT INTO tb_employee SET employee_name='".$employee_name."', employee_salary='".$employee_salary."', employee_age='".$employee_age.'"
";
    if(mysqli_query($connection, $query))
    {
        $response=array(
            'status' => 1,
            'status_message' => 'Employee Added Successfully.'
        );
    }
    else
    {
        $response=array(
            'status' => 0,
            'status_message' => 'Employee Addition Failed.'
        );
    }
    header('Content-Type: application/json');
}

```

```

        echo json_encode($response);
    }

function update_employee($id)
{
    global $connection;
    $post_vars = json_decode(file_get_contents("php://input"),true);
    $employee_name=$post_vars["employee_name"];
    $employee_salary=$post_vars["employee_salary"];
    $employee_age=$post_vars["employee_age"];
    $query="UPDATE tb_employee SET employee_name='".$employee_name."', employee_salary='".$employee_salary."', employee_age='".$employee_age.'" WHERE id=".$id;
    if(mysqli_query($connection, $query))
    {
        $response=array(
            'status' => 1,
            'status_message' =>'Employee Updated Successfully.'
        );
    }
    else
    {
        $response=array(
            'status' => 0,
            'status_message' =>'Employee Updation Failed.'
        );
    }
    header('Content-Type: application/json');
    echo json_encode($response);
}

function delete_employee($id)
{
    global $connection;
    $query="DELETE FROM tb_employee WHERE id=".$id;
    if(mysqli_query($connection, $query))
    {
        $response=array(
            'status' => 1,
            'status_message' =>'Employee Deleted Successfully.'
        );
    }
    else
    {
        $response=array(
            'status' => 0,

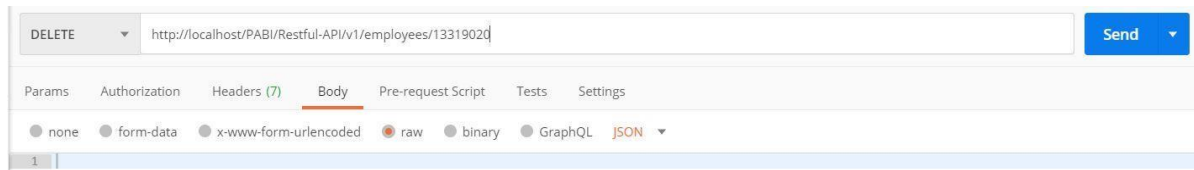
```

```

        'status_message' => 'Employee Deletion Failed.'
    );
}
header('Content-Type: application/json');
echo json_encode($response);
}
?>

```

Pengujian menggunakan tools Postman



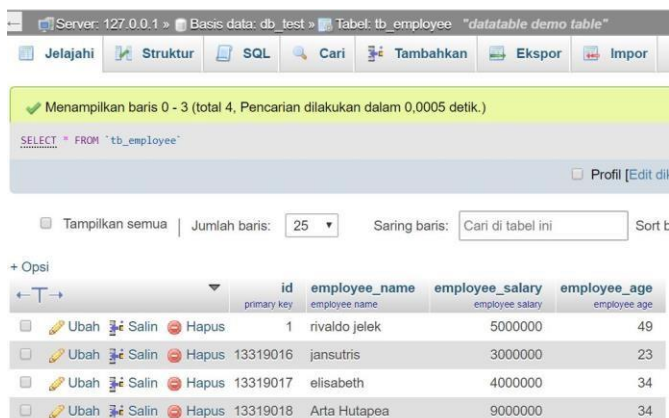
Output

Response body menghasilkan status code 200. Artinya proses delete berhasil. Silahkan anda cek dan refresh langsung ke database untuk memastikan bahwa data sudah terhapus.



Perubahan data pada database

Data moses sudah tidak ada. Artinya fungsi delete telah berhasil dibuktikan.



Kesimpulan

Kami telah membuat RESTful API untuk mengambil semua karyawan, mendapatkan record karyawan tunggal, membuat entri karyawan baru, memperbarui data karyawan dan menghapus data karyawan dari database MySQL. Anda dapat menambahkan opsi terkait keamanan seperti API token untuk membatasi akses rest api dari pengguna yang tidak diotorisasi.

Tugas

1. Pada week sebelumnya Anda sudah membuat aplikasi pendaftaran mahasiswa baru yang memiliki fungsi CRUD (CREATE, READ, UPDATE dan DELETE). Silahkan ciptakan RESTful API server pada aplikasi pendaftaran mahasiswa tersebut dan pastikan API yang anda buat lolos uji menggunakan Postman.
2. Buatlah sebuah aplikasi dengan spesifikasi sebagai berikut:
nama aplikasi: Pantau Korona
Jenis Web service: Rest API
Menggunakan charts/grafik realtime (anda dapat menggunakan AJAX atau JQuery)
Fungsi: JSON time-series of coronavirus cases (confirmed, deaths and recovered) per country which updated daily
Contoh aplikasi yang sudah ada:
<http://kawalcovid.tobasamosirkab.go.id/home>
<https://kawalcorona.com/>
<https://www.arcgis.com/apps/opsdashboard/index.html#/bda7594740fd40299423467b48e9ecf6>

Silahkan gunakan rest-api data corona yang bersifat opensource. Jadi anda tidak perlu membuat database anda sendiri. Cukup gunakan rest-api covid-19 yang sudah ada di internet sehingga hanya anda bertindak sebagai consumer data dan endpoint/rest-server anda bertindak sebagai producer data.

Contoh Referensi API:

<https://documenter.getpostman.com/view/10808728/SzS8rjbc?version=latest>
<https://api.kawalcorona.com/>