



PERTEMUAN 6



1. Pembuatan Database

Buat database dengan nama : **toko_api**

Kemudian buat tabel-tabel dengan perintah sebagai berikut :


a. SQL membuat tabel member

```
CREATE table member (  
    id INT NOT NULL AUTO_INCREMENT,  
    nama VARCHAR(255) NOT NULL,  
    email VARCHAR(255) NOT NULL,  
    password VARCHAR(255) NOT NULL,  
    PRIMARY KEY(id)  
);
```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 	int(11)			No	None		AUTO_INCREMENT
2	nama	varchar(255)	utf8mb4_general_ci		No	None		
3	email	varchar(255)	utf8mb4_general_ci		No	None		
4	password	varchar(255)	utf8mb4_general_ci		No	None		


b. SQL membuat tabel member_token

```
CREATE table member_token (  
  id INT NOT NULL AUTO_INCREMENT,  
  member_id INT NOT NULL,  
  auth_key VARCHAR(255) NOT NULL,  
  FOREIGN KEY (member_id) REFERENCES member(id) on update cascade on delete no action,  
  PRIMARY KEY(id)  
);
```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 	int(11)			No	None		AUTO_INCREMENT
2	member_id 	int(11)			No	None		
3	auth_key	varchar(255)	utf8mb4_general_ci		No	None		

c. SQL membuat tabel produk

```
CREATE table produk (  
  id INT NOT NULL AUTO_INCREMENT,  
  kode_produk VARCHAR(255) NOT NULL,  
  nama_produk VARCHAR(255) NOT NULL,  
  harga INT NOT NULL,  
  PRIMARY KEY(id)  
);
```

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 	int(11)			No	None		AUTO_INCREMENT
2	kode_produk	varchar(255)	utf8mb4_general_ci		No	None		
3	nama_produk	varchar(255)	utf8mb4_general_ci		No	None		
4	harga	int(11)			No	None		

2. Instalasi Lumen sebagai Restful API

Selanjutnya install projek lumen pada web server (pada folder C:\xampp\htdocs)

Untuk membuat projek lumen dapat menggunakan composer pada command prompt dengan menjalankan perintah

```
composer create-project --prefer-dist laravel/lumen <nama_projek>
```


Sekarang kita akan membuat proyek Restful API dengan nama **toko-api**. Silahkan buka Command Prompt kemudian ketikkan perintah

```
C:\Users\BSI> cd c:\xampp\htdocs
```

```
C:\xampp\htdocs> composer create-project --prefer-dist laravel/lumen toko-api
```

3. Konfigurasi Proyek

a. Koneksi ke database

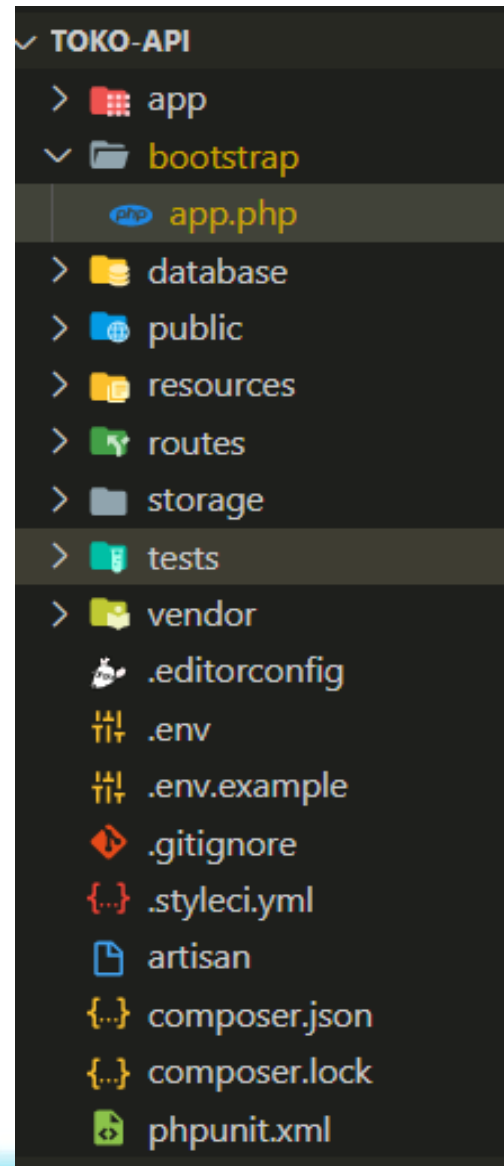
Buka file **.env** kemudian ubah kode berikut

```
DB_DATABASE=homestead  
DB_USERNAME=homestead  
DB_PASSWORD=secret
```

Menjadi sebagai berikut

```
DB_DATABASE=toko_api  
DB_USERNAME=root  
DB_PASSWORD=
```


Kemudian buka file
app.php pada folder
“bootstrap”



Kemudian hilangkan komentar pada kode berikut (pada baris 26 dan 28)

```
// $app->withFacades();
```

```
// $app->withEloquent();
```

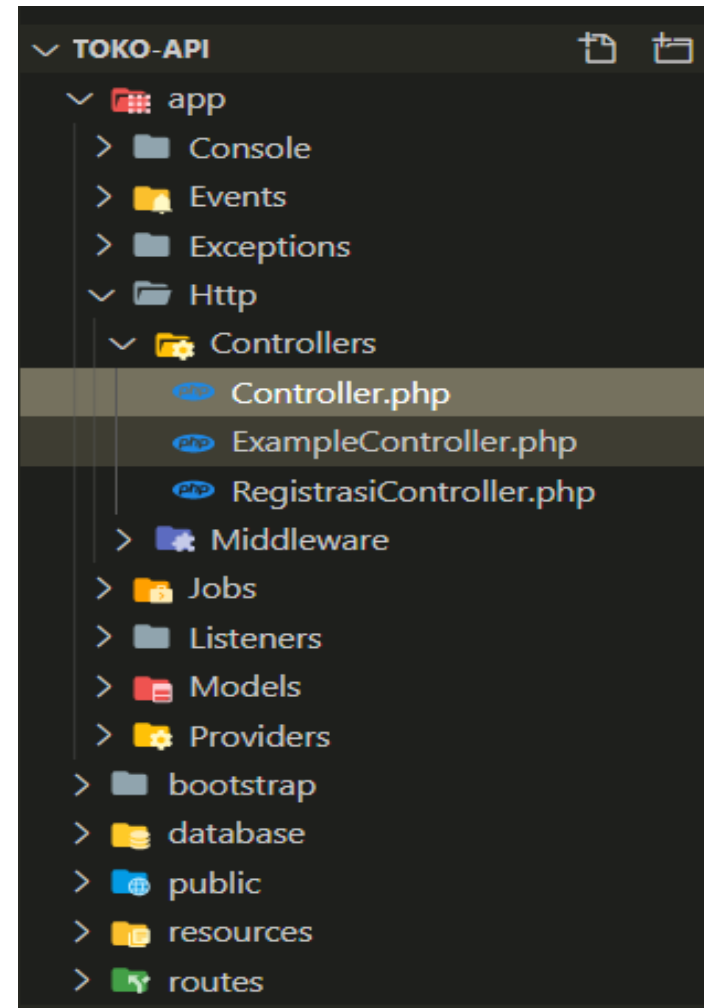
Menjadi

```
$app->withFacades();
```

```
$app->withEloquent();
```

b. Membuat hasil response

Buka file **Controller.php**
pada folder
“app\Http\Controllers”



Kemudian tambahkan kode pada file tersebut sehingga menjadi

```
<?php

namespace App\Http\Controllers;

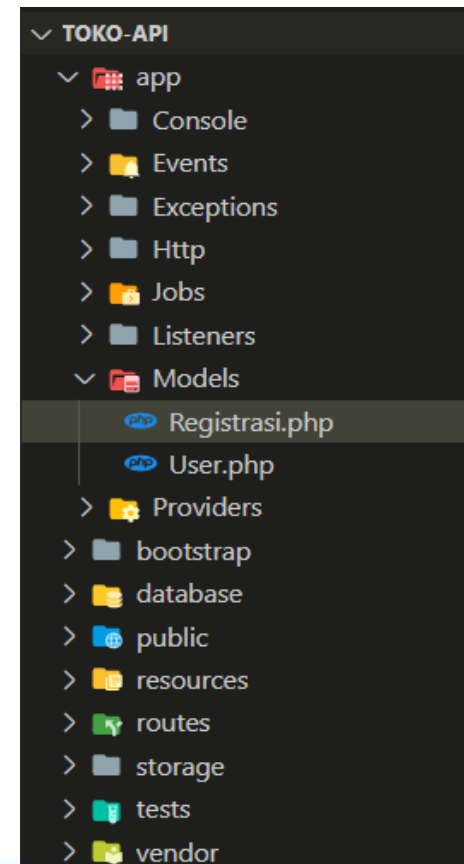
use Laravel\Lumen\Routing\Controller as BaseController;

class Controller extends BaseController
{
    protected function responseHasil($code, $status, $data)
    {
        return \response()->json([
            'code' => $code,
            'status' => $status,
            'data' => $data
        ]);
    }
}
```

4. Registrasi

a. Membuat model Registrasi

Buat sebuah file dengan nama **Registrasi.php** pada folder “app/Models”



Kemudian pada file **Registrasi.php** ketikkan kode berikut

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Model;

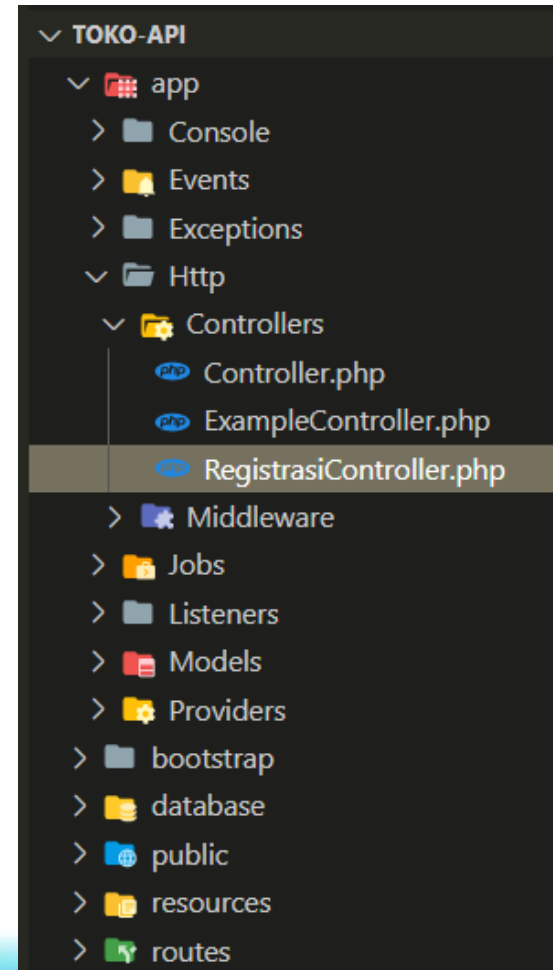
class Registrasi extends Model
{
    protected $table = 'member';

    protected $fillable = ['nama', 'email', 'password'];

    public $timestamps = false;
}
```


b. Membuat controller Registrasi

Buat sebuah file dengan nama
RegistrasiController.php
pada folder
“app\Http\Controller”



Kemudian pada file **RegistrasiController.php** tersebut masukkan kode berikut

```
<?php

namespace App\Http\Controllers;

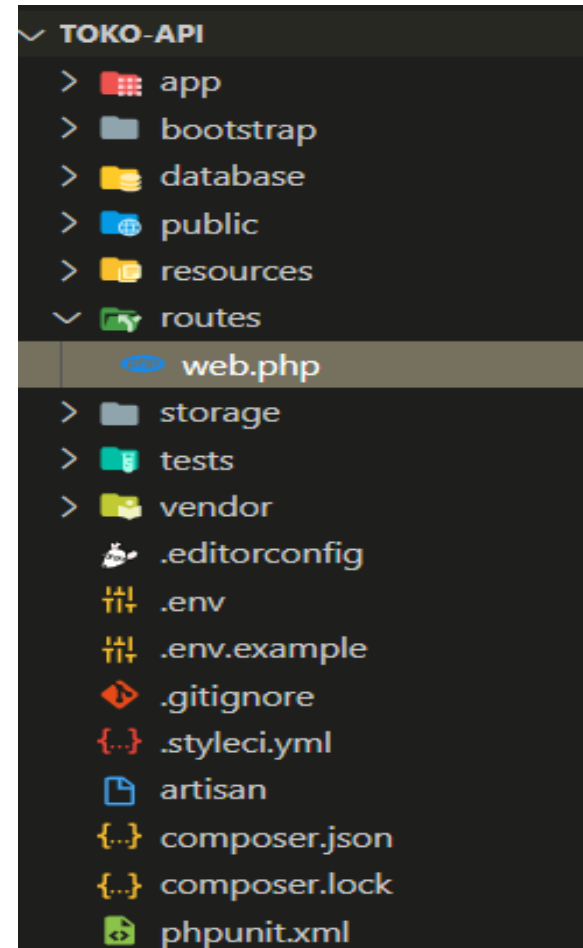
use App\Models\Registrasi;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Hash;

class RegistrasiController extends Controller
{
    public function registrasi(Request $request)
    {
        $nama = $request->input('nama');
        $email = $request->input('email');
        $password = Hash::make($request->input('password'));

        Registrasi::create([
            'nama' => $nama,
            'email' => $email,
            'password' => $password,
        ]);
        return $this->responseHasil(200, true, "Registrasi Berhasil");
    }
}
```

c. Menambah route Registrasi

Buka file **web.php** pada folder “routes”



Kemudian tambahkan kode berikut:

```
<?php

/** @var \Laravel\Lumen\Routing\Router $router */

/*
|-----
| Application Routes
|-----
|
| Here is where you can register all of the routes for an application.
| It is a breeze. Simply tell Lumen the URIs it should respond to
| and give it the Closure to call when that URI is requested.
|
*/

$router->get('/', function () use ($router) {
    return $router->app->version();
});

$router->post('/registrasi', ['uses' => 'RegistrasiController@registrasi']);
```

Pada baris terakhir kita menambahkan routing registrasi agar dapat diakses. Untuk mengakses registrasi, kita gunakan Postman dengan alamat url **localhost/toko-api/public/registrasi** dengan method **POST**

Adapun langkah menggunakan postman untuk menguji Rest API yang telah dibuat adalah sebagai berikut:

1. Buka aplikasi Postman yang telah terinstall
2. Masukkan alamat url untuk melakukan registrasi toko-api yang telah kita buat yaitu <http://localhost/toko-api/public/registrasi>
3. Selanjutnya pilih pengujian dengan type POST
4. Kemudian klik **Body** yang berada pada bagian bawah inputan url, kemudian pilih **x-www-form-urlencoded**
5. Kemudian isi key sesuai dengan field request pada **RegistrasiController** pada fungsi registrasi yaitu *nama, email, password* kemudian klik tombol **Send**


```
$nama = $request->input('nama');  
$email = $request->input('email');  
$password = Hash::make($request->input('password'));
```

POST localhost/toko-api/public/registrasi

Body ▾

form-data ▾

	KEY	VALUE	...	Bulk Edit
<input checked="" type="checkbox"/>	nama	Administrator		
<input checked="" type="checkbox"/>	email	admin@admin.com		
<input checked="" type="checkbox"/>	password	admin		
	Key	Value		

Body ▾

Pretty Raw Preview Visualize JSON

```
1 {  
2   "code": 200,  
3   "status": true,  
4   "data": "Registrasi Berhasil"  
5 }
```