

	MODUL PRAKTIKUM PEMROGRAMAN WEB LANJUT ROUTING	JTI Hal 1 of 10
---	---	---------------------------

TUJUAN PRAKTIKUM

1. Mahasiswa dapat memahami konsep routing dalam framework php
2. Mahasiswa dapat memahami cara pembuatan dan penulisan route dalam framework
3. Mahasiswa dapat mengetahui jenis-jenis route dalam framework

TEORI SINGKAT

Routing pada Laravel merupakan cara mengakses suatu halaman pada aplikasi melalui URL. Misalnya untuk membuka halaman awal aplikasi dapat dilakukan dengan mengetik URL <http://localhost:8000>. Berarti dalam menentukan route, kita menentukan bagaimana struktur URL untuk mengakses halaman tertentu.

File yang digunakan untuk melakukan penyetelan route terdapat pada folder **routes**. Pada folder ini terdapat empat file php untuk pembuatan aplikasi, file yang digunakan untuk pembuatan route adalah **web.php**.

CARA MEMBUAT ROUTE

Untuk pembuatan suatu aplikasi, route dibuat pada file routes/web.php. kita tinggal menambahkan script route pada bagian bawah file tersebut. Adapun format penulisan route yaitu sebagai berikut:

```
Route::get($url,$callback);
```

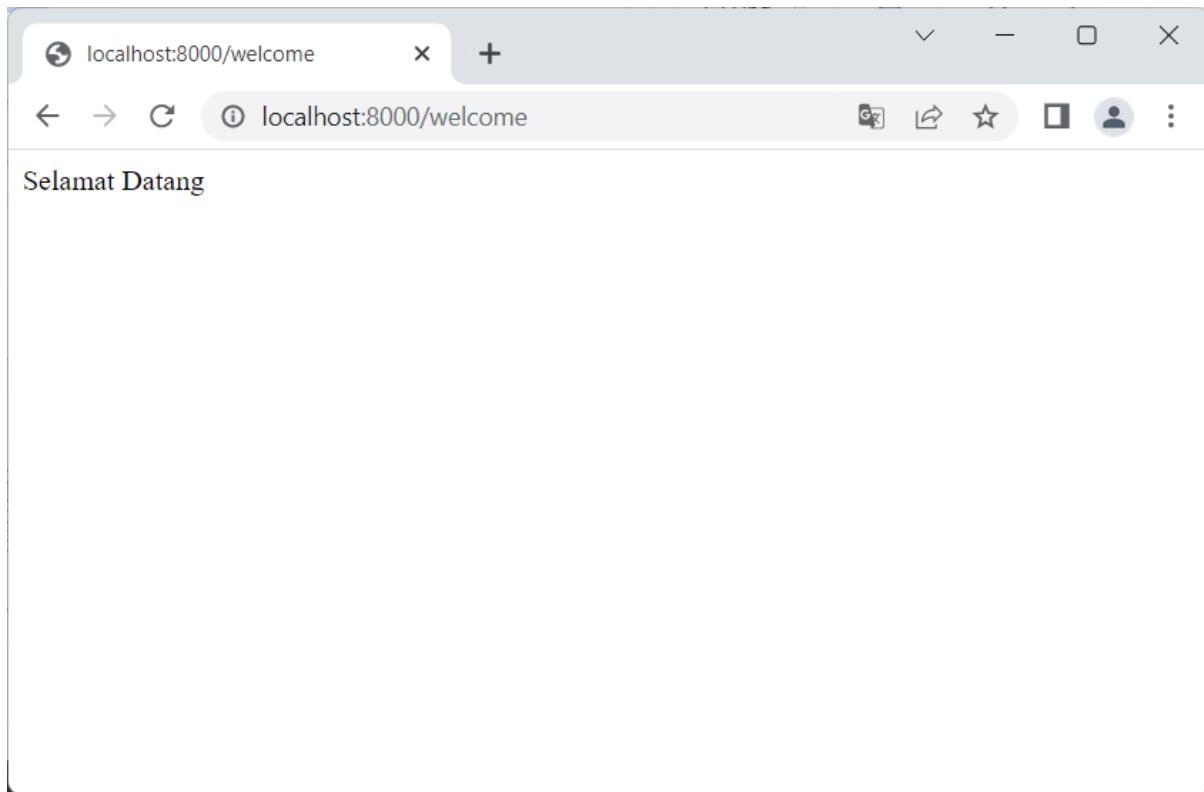
Pada format penulisan di atas, \$url diisi dengan format URL dari route, sedangkan \$callback diisi dengan script yang akan dijalankan ketika membuka URL tersebut. Callback dapat berupa controller atau fungsi. Contoh penulisan route sederhana adalah sebagai berikut:

```
Route::get('/welcome', function(){
    echo "Selamat Datang";
});
```

Untuk menguji route diatas pertama-pertama pastikan server artisan sudah berjalan, untuk menjalankan server artisan itu sendiri sudah dibahas pada modul sebelumnya yaitu melalui cmd atau comand prompt dengan cara mengarahkan direktori ke dalam projek laravel yang dalam kasus ini adalah projek websaya dan kemudian mengetikan php artisan serve.



Lalu jika kita mengetikan pada browser <http://localhost:8000/welcome> maka hasilnya akan seperti gambar diabawah:



JENIS-JENIS ROUTE METHOD

Ada berbagai method dalam route yang dapat digunakan sesuai dengan HTTP request. Pada contoh berikut ada beberapa contoh route yang menggunakan route method:

1. Untuk menampilkan atau mengambil data dapat menggunakan method GET. Route dengan method ini dapat langsung kita akses melalui link pada browser. Contohnya dapat dilihat pada latihan dibawah:

```
Route::get('/index', function(){
    echo "Uji coba route dengan method GET";
});
```

2. Untuk mengirim data dari form dengan dapat menggunakan method POST, biasasnya digunakan untuk menambah data. Adapun penulisan dari route ini adalah sebagai berikut:



```
Route::post('/save', function(){
    echo "Uji coba route dengan method POST";
});
```

3. Untuk mengirim data dari form dengan tujuan untuk memperbarui data dapat menggunakan method PUT, contoh penulisan route dengan method PUT adalah sebagai berikut:

```
Route::put('/update', function(){
    echo "Uji coba route PUT";
});
```

4. Untuk mengirim data dari form dengan tujuan untuk menghapus data dapat menggunakan method DELETE. Contoh dari penulisan route dengan method delete adalah sebagai berikut:

```
Route::delete('/delete', function(){
    echo "Uji coba route DELETE";
});
```

5. Untuk route yang dapat merespons beberapa HTTP request, kita dapat menggunakan method match().

```
Route::match(['get','post'],'/kirim', function(){
    echo "Uji coba route method GET dan POST";
});
```

6. Sedangkan untuk route yang dapat merespons semua HTTP request, dapat menggunakan method any(). Contohnya pada script berikut:

```
Route::any('/home', function(){
    echo "Uji coba route ANY";
});
```

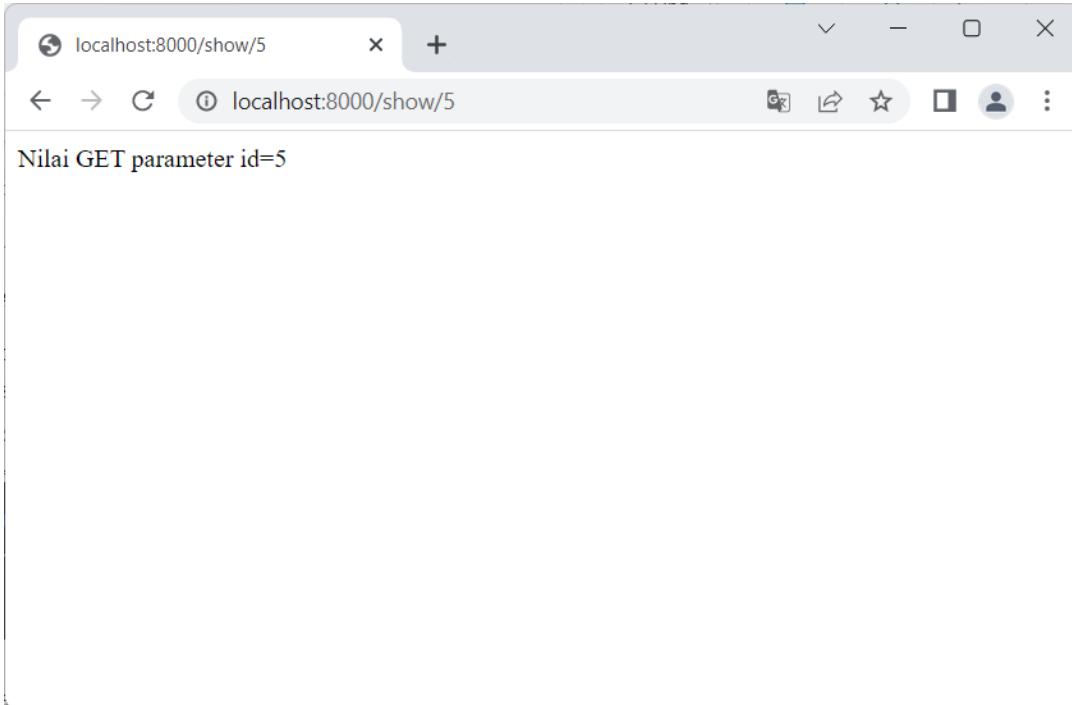


KEGIATAN PRAKTIKUM BERBAGAI CARA PENULISAN ROUTE

1. Route dengan parameter Parameter pada route ditandai dengan tanda { }. Artinya kata pada URL yang berada pada posisi ini akan menjadi nilai dari parameter. Untuk penulisannya buatlah satu route baru pada file web.php yang ada pada folder routes/web.php dengan skrip sebagai berikut:

```
Route::get('/show/{id}', function($id){  
    echo "Nilai GET parameter id=". $id;  
});
```

Kemudian bukalah browser dan ketikan pada link <http://localhost:8000/show/5> angka 5 tersebut akan menjadi parameter dari route yang dibuat yang artinya bebas diinputkan berapa saja dan jika dilihat pada browser maka hasilnya akan menjadi seperti gambar dibawah:

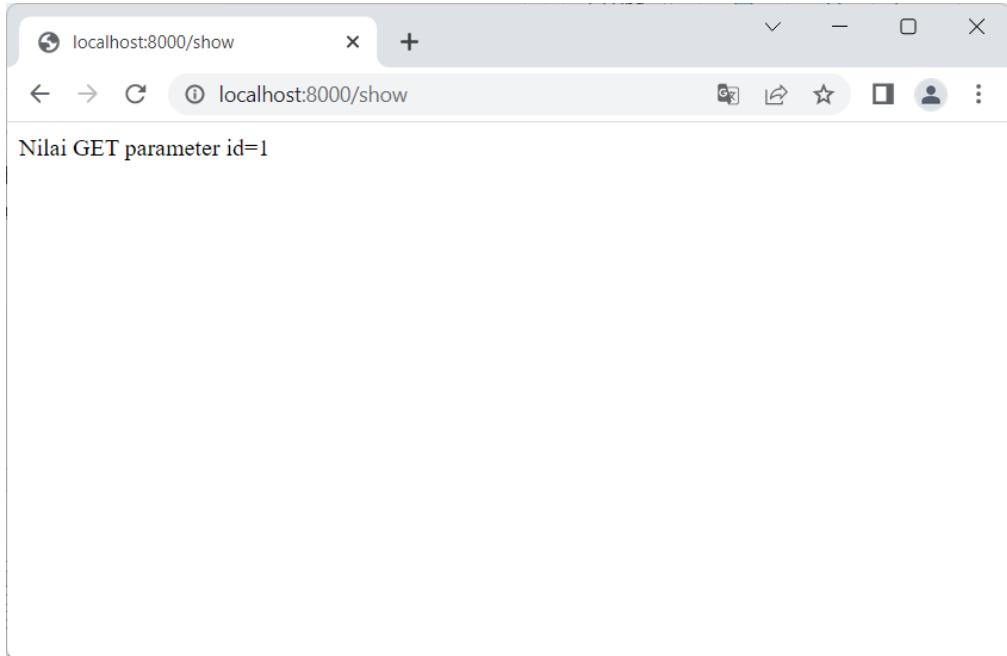


Jika parameter tidak wajib diisi maka tambahkan tanda tanya (?) dibelakang nama parameter dan pada fungsi kita beri default parameter pada contoh dibawah jika parameter tidak disebutkan misalnya pada URL <http://localhost:8000/show> maka parameter \$id akan dianggap bernilai 1 sesuai dengan default parameter yang diberikan.



```
Route::get('/show/{id?}', function($id=1){  
    echo "Nilai GET parameter id=". $id;  
});
```

Jika route diatas dijalankan maka hasilnya akan menjadi seperti gambar dibawah:



2. Route dengan reguler expression

Route ini akan dijalankan hanya jika nilai dari parameter pada URL memenuhi syarat tertentu. Misalnya pada baris pertama contoh di bawah, parameter judul harus berupa karakter A sampai Z atau a sampai z. Dengan demikian, maka URL <http://localhost:8000/edit/udin> memenuhi syarat dari route tersebut, sedangkan URL <http://localhost:8000/edit/123> tidak dapat menggunakan route tersebut. Adapun contoh penulisan route dengan prasyarat dapat dapat dipraktekan dengan menambahkan route dibawah pada file **web.php** pada folder **routes/web.php**.

```
Route::get('/edit/{nama}', function($nama){  
    echo "Nilai GET parameter nama=". $nama;  
})->where('nama', '[A-Za-z]+');
```

Kemudian jika dijalankan pada browser dengan alamat link <http://localhost:8000/edit/udin> maka hasilnya akan menjadi seperti berikut:



**MODUL PRAKTIKUM
PEMROGRAMAN WEB LANJUT
ROUTING**

JTI

Hal 6 of 10

localhost:8000/edit/udin

Nilai GET parameter nama=udin

Sedangkan jika nilai pada bagian parameternya kita ganti menjadi angka maka Laravel tidak akan mengeksekusi route tersebut. Dapat dicoba dengan mengetikan pada link browser alamat <http://localhost:8000/edit/123> maka laravel akan mengeluarkan error seperti berikut:

Not Found

localhost:8000/edit/123

404 | NOT FOUND



Error diatas bertuliskan 404 | NOT FOUND yang berarti Route untuk link yang diakses tidak ditemukan.

3. Route dengan nama

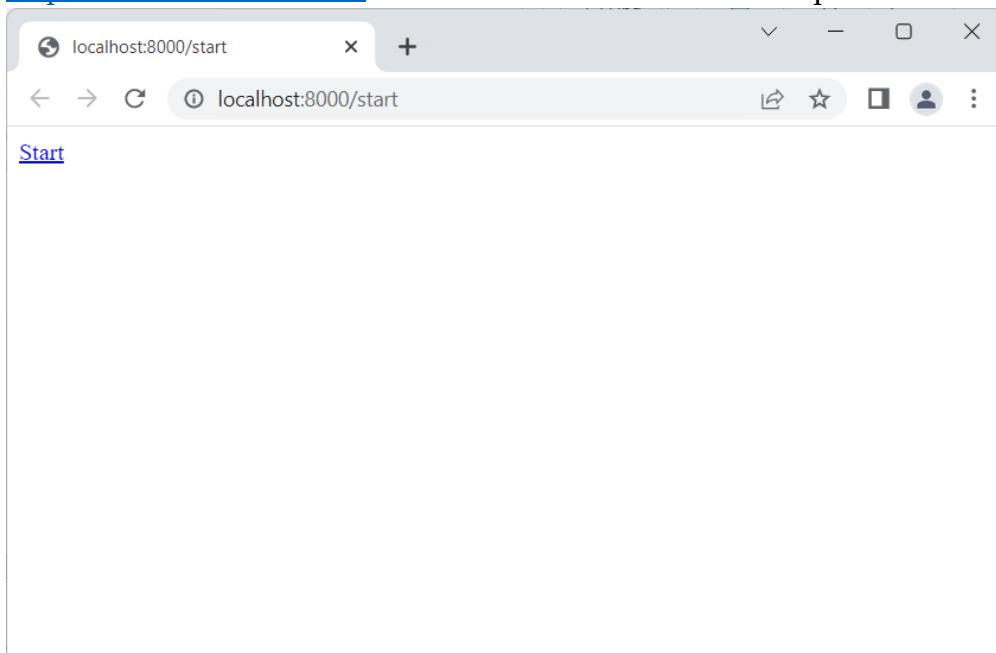
Route dengan pendefinisian nama atau identitas memiliki kelebihan yang dapat digunakan hanya dengan menyebutkan nama yang diberikan. Contohnya buatlah 2 route baru pada file **web.php** yang ada pada folder **routes/web.php**

```
Route::get('/start', function(){
    echo "<a href='".$route('coba')."'>Start</a>";
})->name('start');
```

Route diatas akan digunakan untuk menampilkan tag dalam atribut href nya akan diisikan nama dari route yang akan diakses

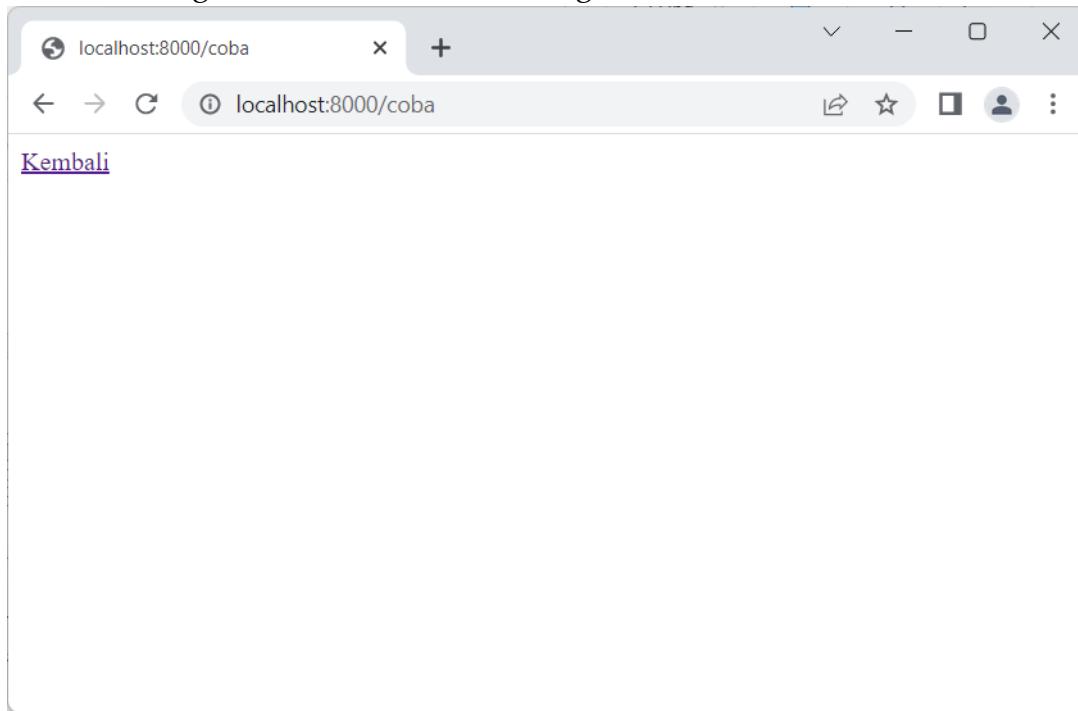
```
Route::get('/coba', function(){
    echo "<a href='".$route('start')."'>Kembali</a>";
})->name('coba');
```

Route diatas adalah route dengan pendefinisian nama yang akan coba kita akses dari route diatasnya. Untuk melakukan uji coba pada Route dengan pendefinisian nama bukalah browser dan ketikan link <http://localhost:8000/start> maka akan muncul halaman seperti berikut:





Jika link Akses route dengan nama di klik maka kita akan diarahkan ke halaman dengan route name create sebagai berikut:



4. Route dengan aksi controller

Route seperti ini ketika dijalankan akan mengakses controller yang disebutkan pada parameter kedua. Jika ingin mengakses method atau function tertentu pada controller, maka parameter yang dapat digunakan [namaController::class, namaFungsi]. Untuk mempraktekan route ini buatlah satu controller dengan menggunakan perintah artisan sebagai berikut:

```
php artisan make:controller mhsController
```

Perintah artisan diatas akan menghasilkan satu file baru bernama **produkController.php** yang terletak di folder **app\Http\Controller**. Bukanlah file controller tersebut dan tambahkan satu fungsi bernama index seperti pada contoh skrip dibawah:



```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

class mhsController extends Controller
{
    public function index(){
        return "Mengakses fungsi di controller menggunakan route";
    }
}
```

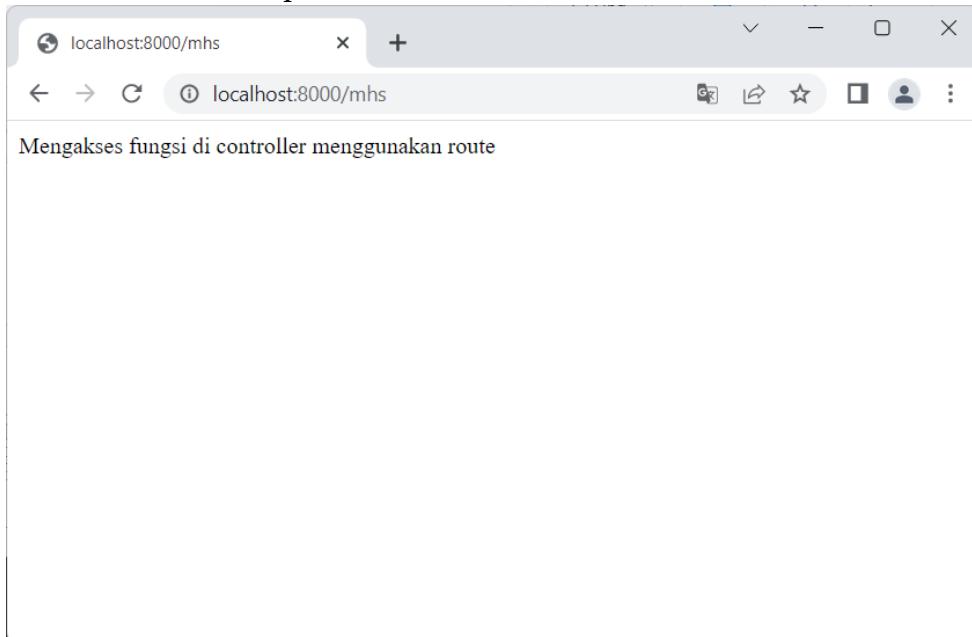
Kemudian pada file **web.php** yang ada pada folder **routes/web.php** tambahkan satu route baru dengan bentuk seperti berikut:

```
<?php

use Illuminate\Support\Facades\Route;
use App\Http\Controllers\mhsController;

Route::get('/mhs', [mhsController::class, 'index'])->name('mhs');
```

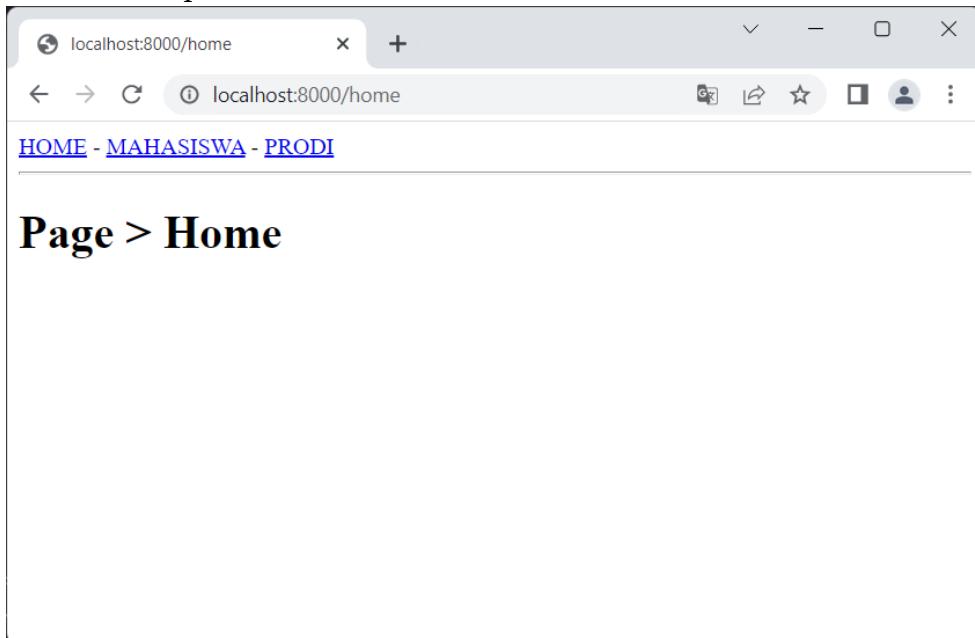
Lakukan uji coba pada route diatas dengan cara membuka browser dan ketikan <http://localhost:8000/mhs> pada link browser yang digunakan maka akan muncul halaman seperti berikut:





TUGAS

- Buatlah sebuah halaman utama dengan mempunyai 3 hyperlink yaitu HOME, MAHASISWA, PRODI
- Masing-masing halaman mempunyai controller
- Lakukan routing sesuai hyperlink yang di *request*
- Contoh tampilan halaman home:



- Tampilan disesuaikan dengan hyperlink yang di klik

Catatan:

- Langkah-langkah pembuatannya dapat mengikuti praktikum
- Laporan praktikum tugas dan sourcode dikumpulkan ke elearning