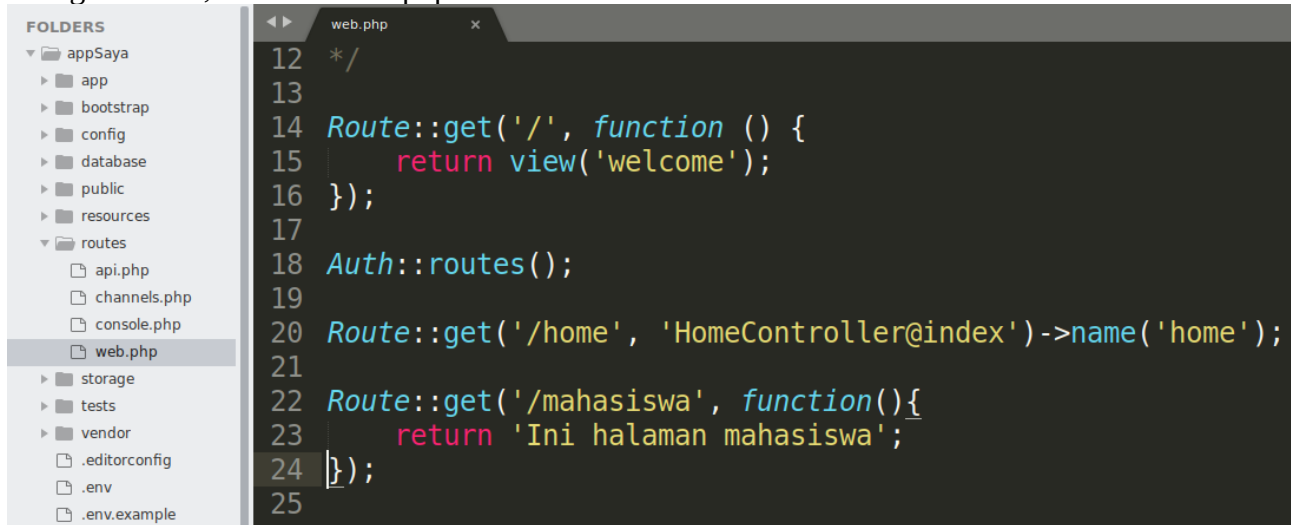


## Routing

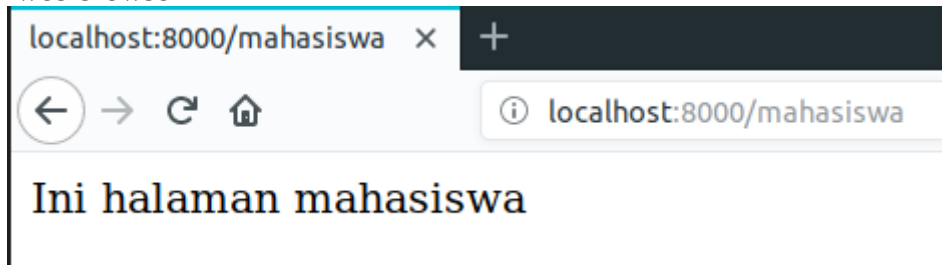
Route URL diatur pada file routes/web.php

Sebagai contoh, buka file web.php kemudian tambahkan kode berikut:



```
12 */
13
14 Route::get('/', function () {
15     return view('welcome');
16 });
17
18 Auth::routes();
19
20 Route::get('/home', 'HomeController@index')->name('home');
21
22 Route::get('/mahasiswa', function(){
23     return 'Ini halaman mahasiswa';
24 });
25
```

Coba buka di web browser



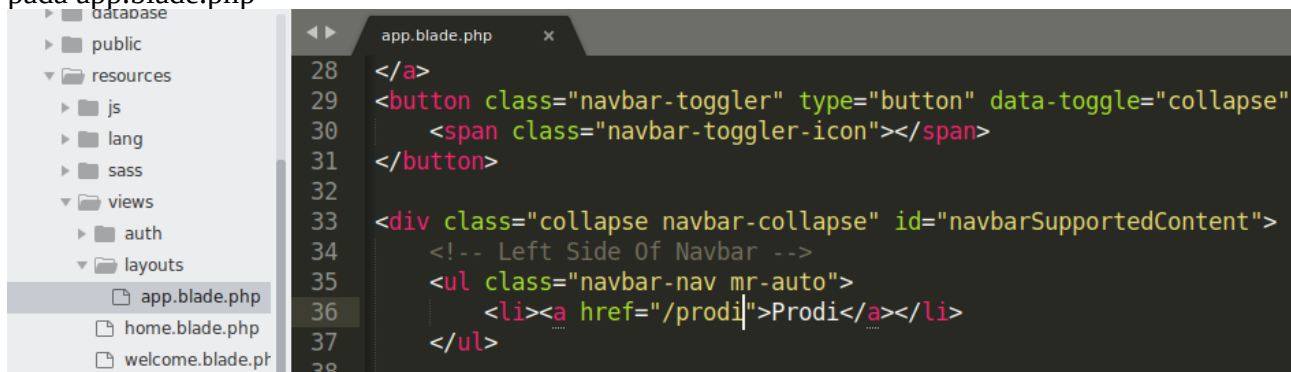
Untuk melihat daftar route yang ada, coba ketikkan command berikut:

```
alexwawo@alexwawo:~/Code/appSaya$ php artisan route:list
```

Domain	Method	URI	Name	Action	Middleware
	GET HEAD	/		Closure	web
	GET HEAD	api/user		Closure	api,auth:api
	GET HEAD	home	home	App\Http\Controllers\HomeController@index	web,auth
	GET HEAD	login	login	App\Http\Controllers\Auth\LoginController@showLoginForm	web,guest
	POST	login	login	App\Http\Controllers\Auth\LoginController@login	web,guest
	POST	logout	logout	App\Http\Controllers\Auth\LoginController@logout	web
	GET HEAD	mahasiswa		Closure	web
	POST	password/email	password.email	App\Http\Controllers\Auth\ForgotPasswordController@sendResetLinkEmail	web,guest
	GET HEAD	password/reset	password.request	App\Http\Controllers\Auth\ForgotPasswordController@showLinkRequestForm	web,guest
	POST	password/reset	password.update	App\Http\Controllers\Auth\ResetPasswordController@reset	web,guest
	GET HEAD	password/reset/{token}	password.reset	App\Http\Controllers\Auth\ResetPasswordController@showResetForm	web,guest
	GET HEAD	register	register	App\Http\Controllers\Auth\RegisterController@showRegistrationForm	web,guest
	POST	register	register	App\Http\Controllers\Auth\RegisterController@register	web,guest

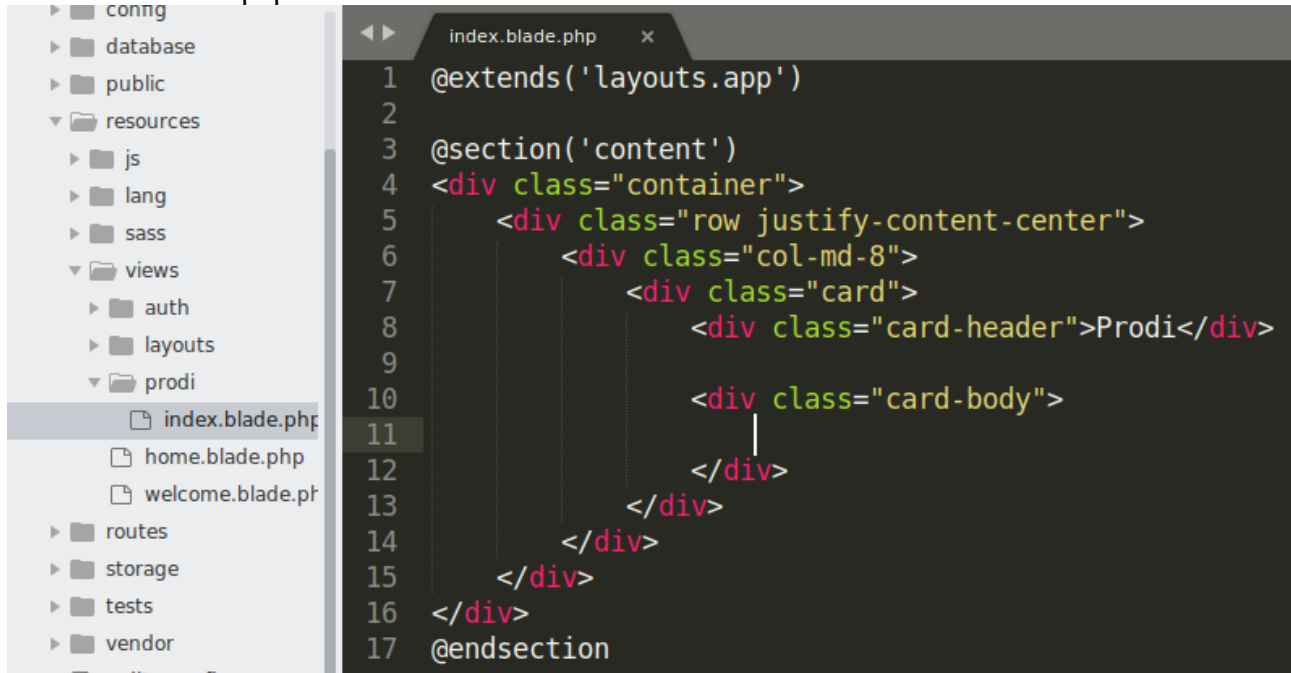
## Blade Templating Engine

Laravel menggunakan blade templating engine untuk membuat tampilan. Tambahkan kode berikut pada app.blade.php



```
28 </a>
29 <button class="navbar-toggler" type="button" data-toggle="collapse"
30     <span class="navbar-toggler-icon"></span>
31 </button>
32
33 <div class="collapse navbar-collapse" id="navbarSupportedContent">
34     <!-- Left Side Of Navbar -->
35     <ul class="navbar-nav mr-auto">
36         <li><a href="/prod" >Prod</a></li>
37     </ul>
38
```

Buatlah sebuah folder baru dengan nama “prodi” di dalam folder views. Kemudian buat file dengan nama index.blade.php kemudian tuliskan kode berikut.



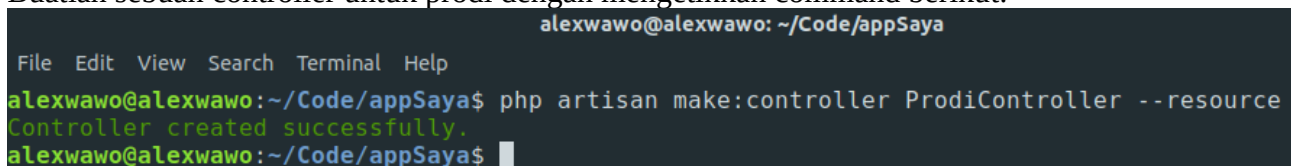
The screenshot shows a file explorer on the left with the following structure:

- config
- database
- public
- resources
  - js
  - lang
  - sass
  - views
    - auth
    - layouts
    - prodi
- routes
- storage
- tests
- vendor

The main editor shows the content of `index.blade.php`:

```
1 @extends('layouts.app')
2
3 @section('content')
4 <div class="container">
5     <div class="row justify-content-center">
6         <div class="col-md-8">
7             <div class="card">
8                 <div class="card-header">Prodi</div>
9
10                <div class="card-body">
11                    |
12                </div>
13            </div>
14        </div>
15    </div>
16 </div>
17 @endsection
```

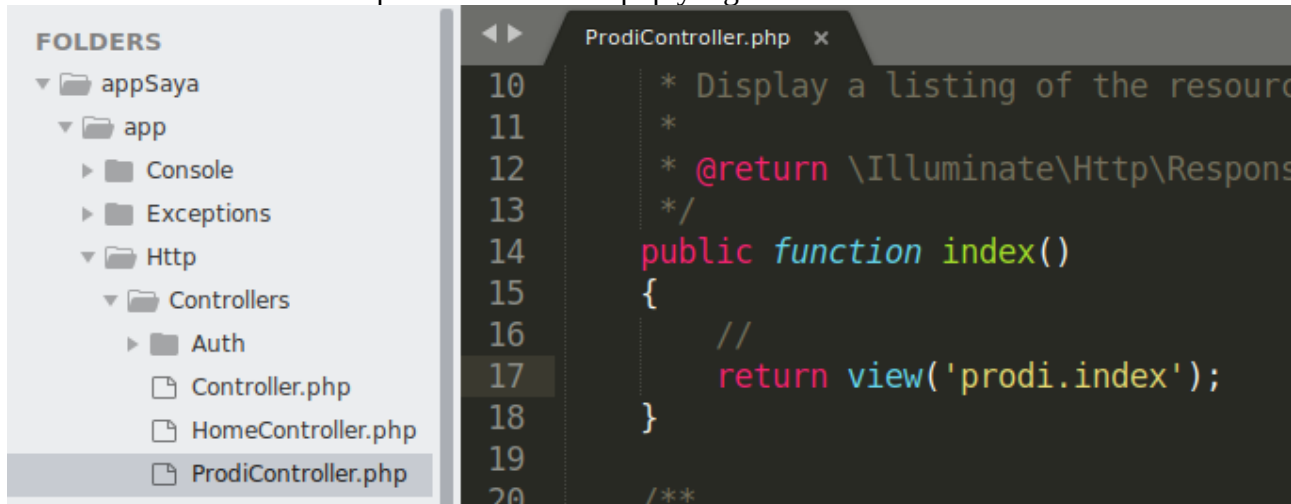
Buatlah sebuah controller untuk prodi dengan mengetikkan command berikut:



The terminal window shows the following command and output:

```
alexwawo@alexwawo: ~/Code/appSaya
File Edit View Search Terminal Help
alexwawo@alexwawo:~/Code/appSaya$ php artisan make:controller ProdiController --resource
Controller created successfully.
alexwawo@alexwawo:~/Code/appSaya$
```

Akan terbentuk sebuah file ProdiController.php, tambahkan kode berikut untuk menginstruksikan function index untuk menampilkan index.blade.php yang tadi dibuat.



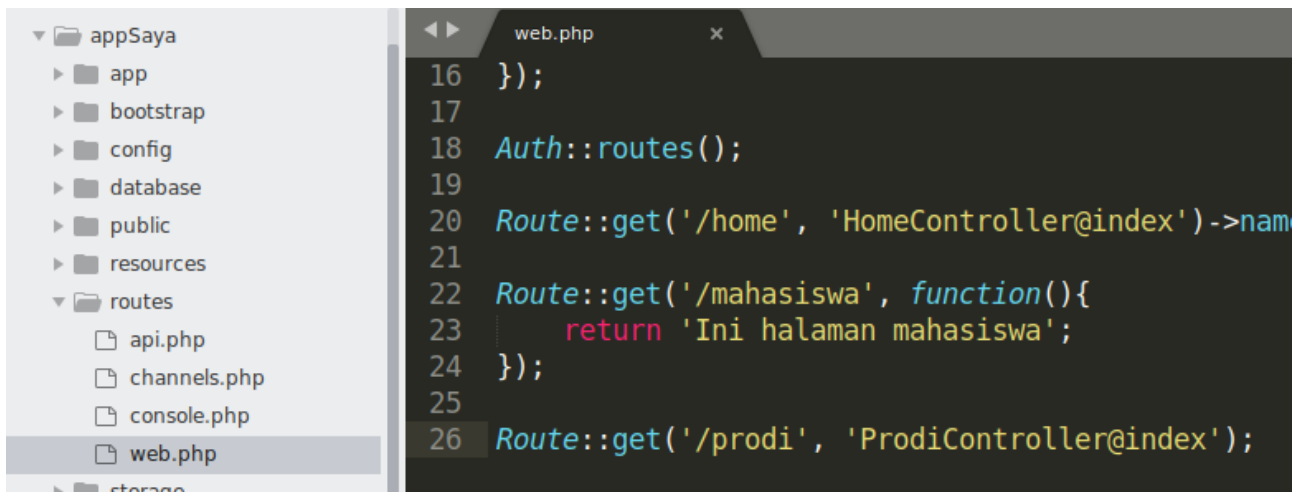
The screenshot shows a file explorer on the left with the following structure:

- appSaya
  - app
    - Console
    - Exceptions
    - Http
      - Controllers
        - Auth
        - Controller.php
        - HomeController.php
        - ProdiController.php

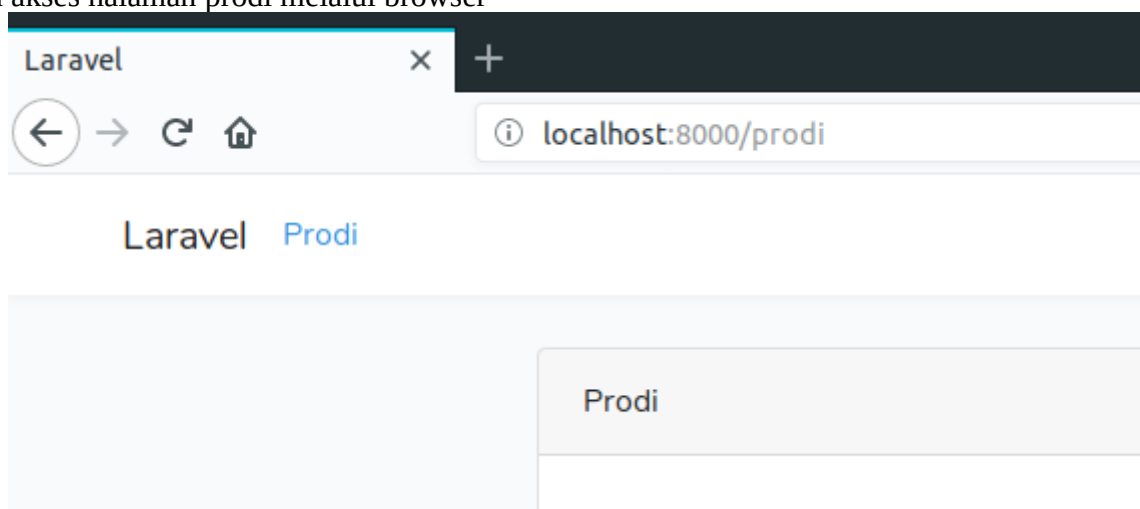
The main editor shows the content of `ProdiController.php`:

```
10 * Display a listing of the resource
11 *
12 * @return \Illuminate\Http\Response
13 */
14 public function index()
15 {
16     //
17     return view('prodi.index');
18 }
19
20 /**
```

Buat route sehingga halaman prodi bisa diakses.

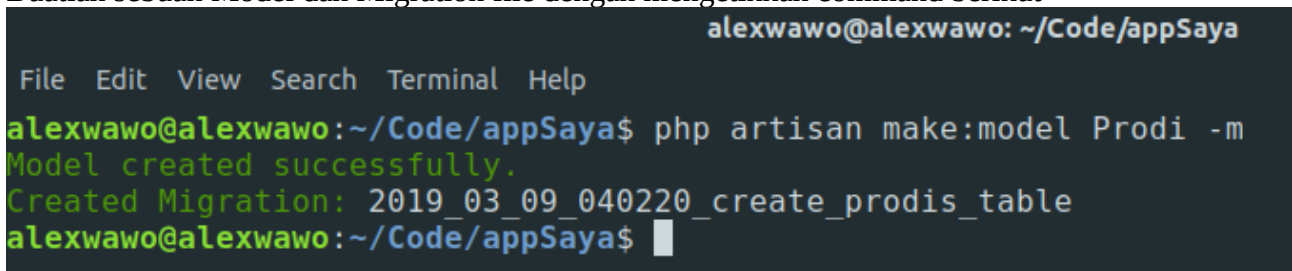


Coba akses halaman prodi melalui browser

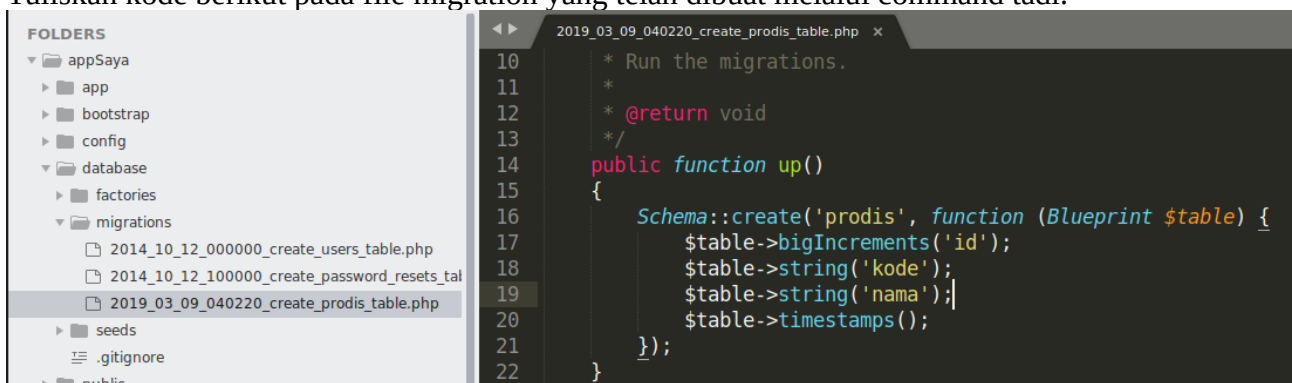


Kita akan membuat tabel baru untuk menyimpan data prodi.

Buatlah sebuah Model dan Migration file dengan mengetikkan command berikut



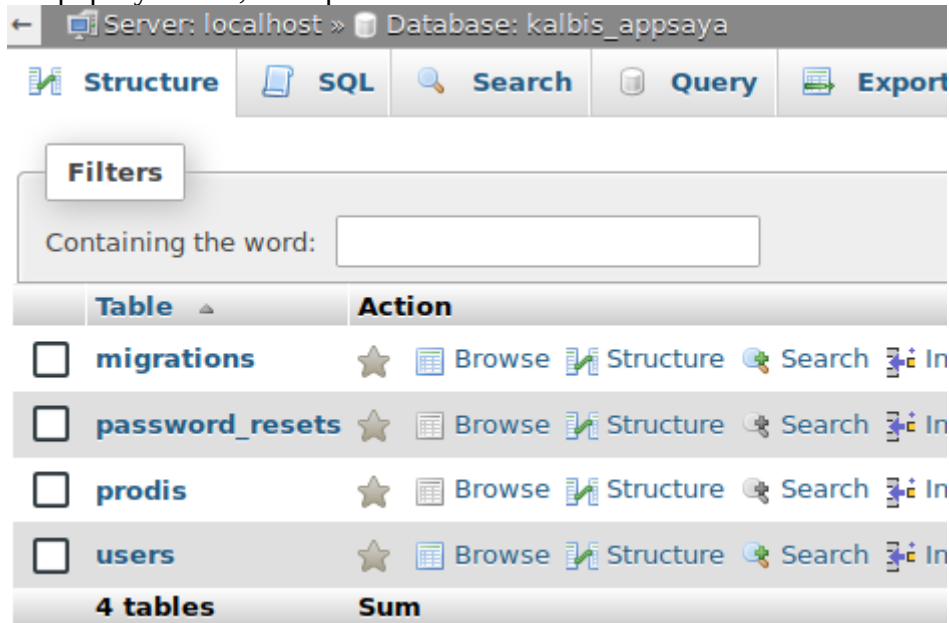
Tuliskan kode berikut pada file migration yang telah dibuat melalui command tadi.



Lakukan migration dengan mengetikkan command berikut

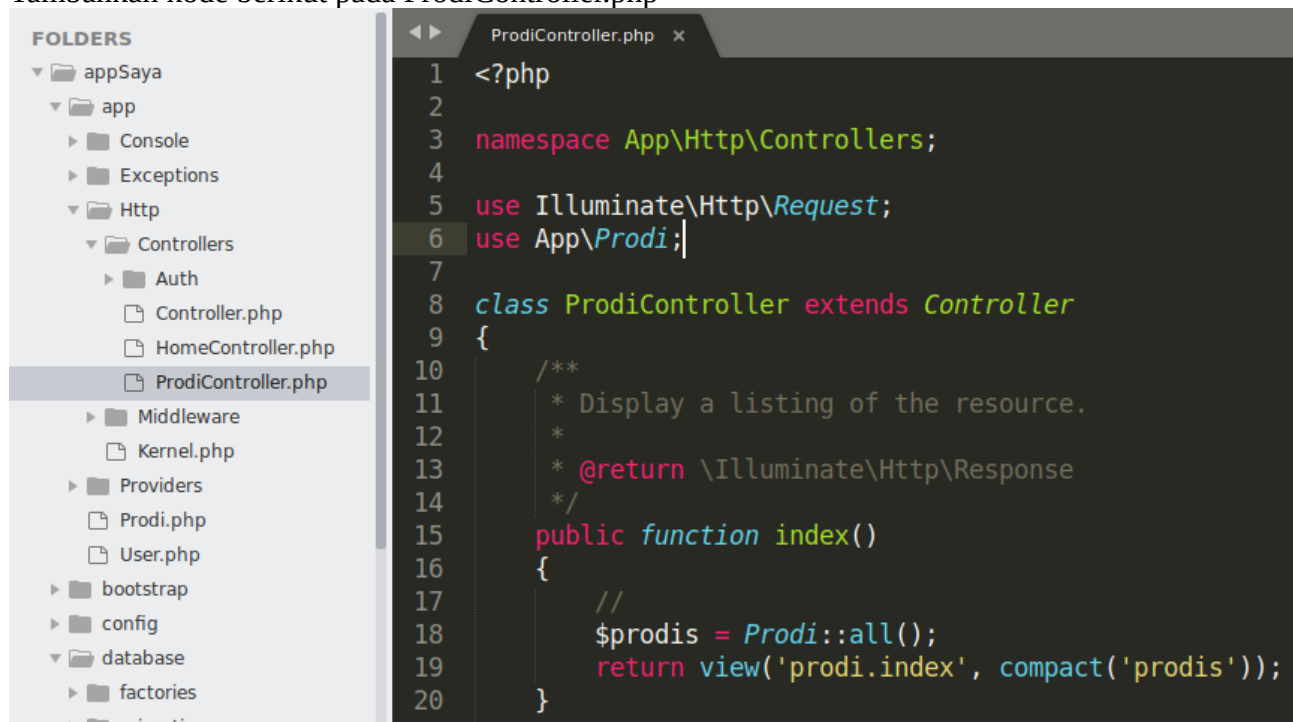
```
alexwawo@alexwawo: ~/Code
File Edit View Search Terminal Help
alexwawo@alexwawo:~/Code/appSaya$ php artisan migrate
Migrating: 2019_03_09_040220_create_prodis_table
Migrated: 2019_03_09_040220_create_prodis_table
alexwawo@alexwawo:~/Code/appSaya$
```

Lihat hasilnya di phpMyAdmin, tabel prodis sudah terbentuk.



Coba insert beberapa data secara langsung melalui phpMyAdmin. Kemudian kita akan mencoba menampilkan datanya pada halaman index yang tadi kita buat.

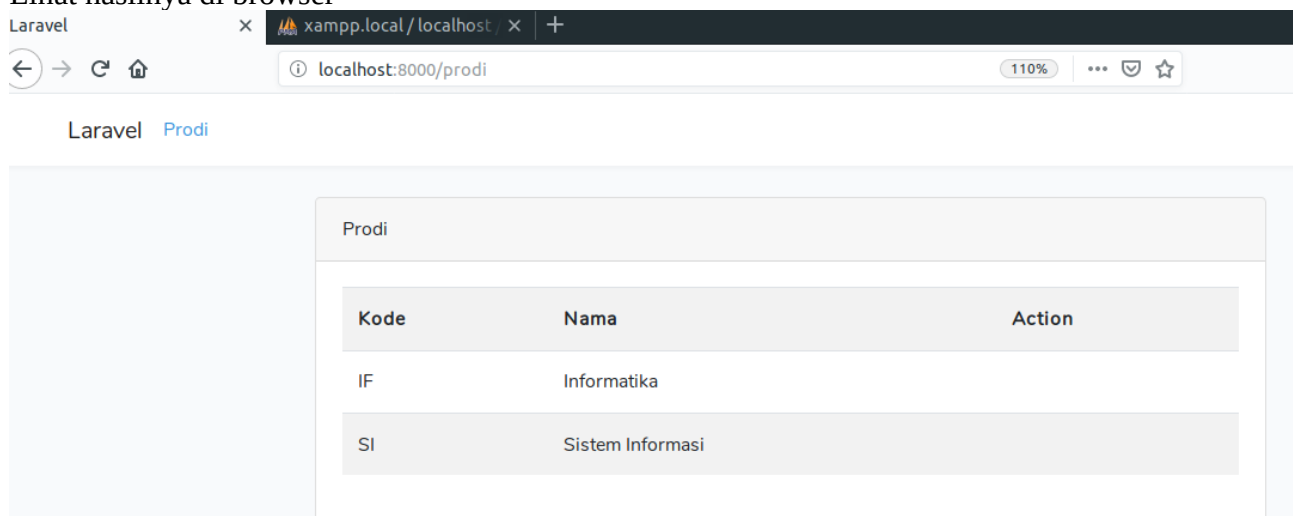
Tambahkan kode berikut pada ProdiController.php



Tambahkan kode berikut pada index.blade.php

```
7      <div class="card">
8          <div class="card-header">Prodi</div>
9
10         <div class="card-body">
11             <table class="table table-striped">
12                 <tr>
13                     <th>Kode</th>
14                     <th>Nama</th>
15                     <th>Action</th>
16                 </tr>
17                 @foreach($prodis as $prodi)
18                     <tr>
19                         <td>{{ $prodi->kode }}</td>
20                         <td>{{ $prodi->nama }}</td>
21                         <td>
22
23                         </td>
24                     </tr>
25                 @endforeach
26             </table>
27         </div>
28     </div>
29 </div>
30 </div>
31 </div>
```

Lihat hasilnya di browser



Tambahkan sebuah link di index.blade.php untuk membuka halaman Buat Prodi Baru

```
<div class="card-header">Prodi</div>
<div class="card-body">
  <a href="/prodi/new" class="btn btn-primary">Buat Prodi Baru</a>
  <table class="table table-striped">
    <tr>
      <th>Kode</th>
```

Prodi

Buat Prodi Baru

Kode	Nama
IF	Informatika

Buat file create.blade.php, tuliskan kode berikut

```
1 @extends('layouts.app')
2
3 @section('content')
4 <div class="container">
5   <div class="row justify-content-center">
6     <div class="col-md-8">
7       <div class="card">
8         <div class="card-header">Buat Prodi Baru</div>
9         <div class="card-body">
10          <form action="{{ url('/prodi/new') }}" method="post">
11            {{ csrf_field() }}
12            <div class="form-group">
13              <label>Kode Prodi</label>
14              <input type="text" name="kode" class="form-control">
15            </div>
16            <div class="form-group">
17              <label>Nama Prodi</label>
18              <input type="text" name="nama" class="form-control">
19            </div>
20            <button type="submit" class="btn btn-primary">Simpan</button>
21          </form>
22        </div>
23      </div>
24    </div>
25  </div>
26 </div>
27 @endsection
```

Tambahkan kode berikut pada file ProdiController.php

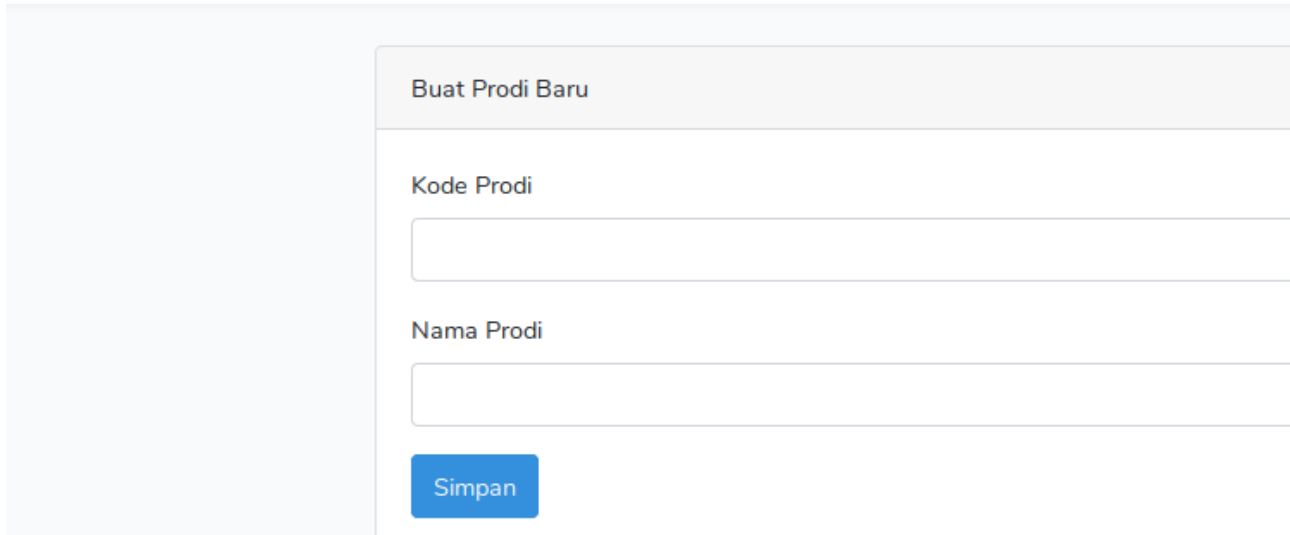
```
27 public function create()
28 {
29   //
30   return view('prodi.create');
31 }
```

Tambahkan route agar halaman buat prodi baru bisa diakses

```
Route::get('/prodi', 'ProdiController@index');  
Route::get('/prodi/new', 'ProdiController@create');
```

Coba akses halaman buat prodi baru

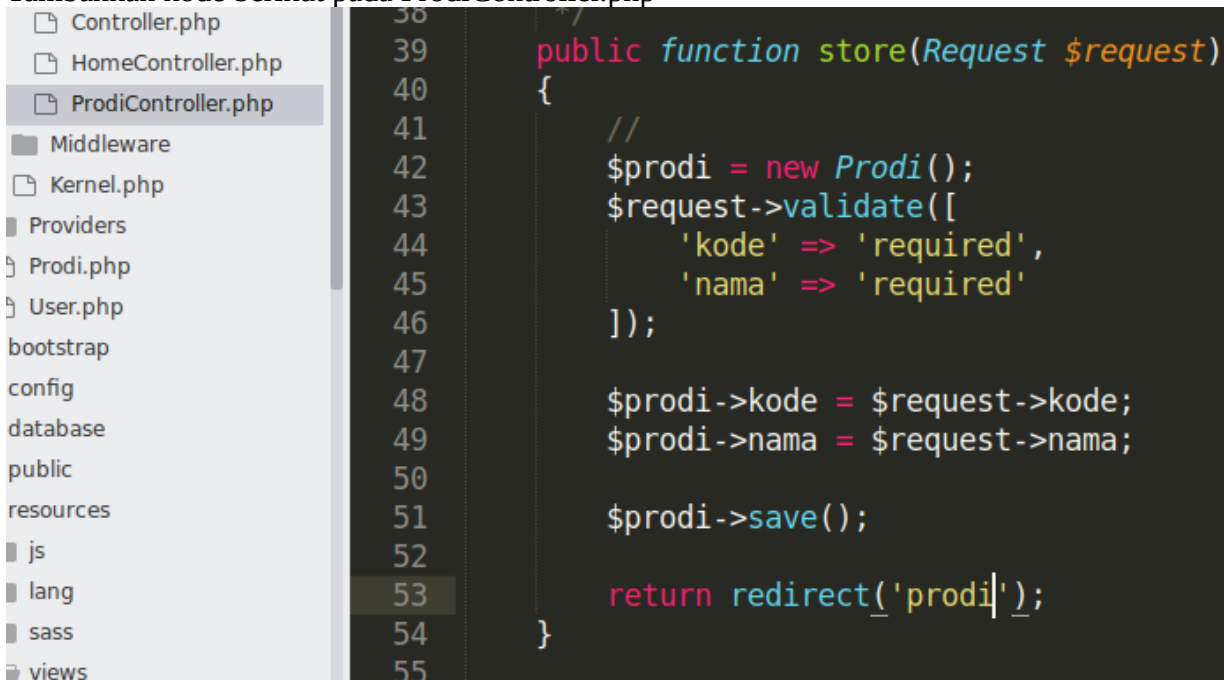
Laravel Prodi



Tambahkan route berikut untuk memproses form tadi.

```
Route::get('/prodi', 'ProdiController@index');  
Route::get('/prodi/new', 'ProdiController@create');  
Route::post('/prodi/new', 'ProdiController@store');
```

Tambahkan kode berikut pada ProdiController.php



```
38  
39 public function store(Request $request)  
40 {  
41     //  
42     $prodi = new Prodi();  
43     $request->validate([  
44         'kode' => 'required',  
45         'nama' => 'required'  
46     ]);  
47  
48     $prodi->kode = $request->kode;  
49     $prodi->nama = $request->nama;  
50  
51     $prodi->save();  
52  
53     return redirect('prodi');  
54 }  
55
```

Coba tambah prodi baru melalui web browser.



Tambahkan link berikut pada index.blade.php

```
<td>{{ $prodi->kode }}</td>
<td>{{ $prodi->nama }}</td>
<td>
    <a href="/prodi/edit/{{ $prodi->id }}">Edit</a>
</td>
</tr>
<endforeach>
```

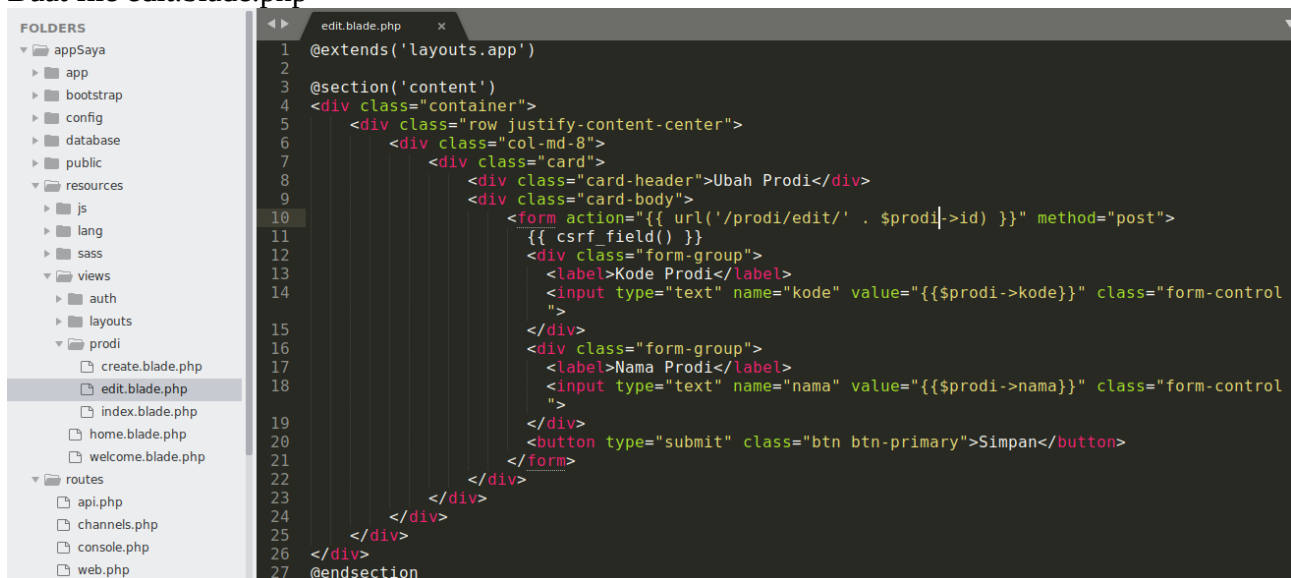
Buat route untuk halaman form edit

```
Route::get('/prodi', 'ProdiController@index');
Route::get('/prodi/new', 'ProdiController@create');
Route::post('/prodi/new', 'ProdiController@store');
Route::get('/prodi/edit/{id}', 'ProdiController@edit');
```

Tambahkan kode berikut pada ProdiController.php

```
public function edit($id)
{
    //
    $prodi = Prodi::find($id);
    return view('prodi.edit', compact('prodi'));
}
```

Buat file edit.blade.php



```
1 @extends('layouts.app')
2
3 @section('content')
4 <div class="container">
5 <div class="row justify-content-center">
6 <div class="col-md-8">
7 <div class="card">
8 <div class="card-header">Ubah Prodi</div>
9 <div class="card-body">
10 <form action="{{ url('/prodi/edit/' . $prodi->id) }}" method="post">
11 {{ csrf_field() }}
12 <div class="form-group">
13 <label>Kode Prodi</label>
14 <input type="text" name="kode" value="{{ $prodi->kode }}" class="form-control">
15 </div>
16 <div class="form-group">
17 <label>Nama Prodi</label>
18 <input type="text" name="nama" value="{{ $prodi->nama }}" class="form-control">
19 </div>
20 <button type="submit" class="btn btn-primary">Simpan</button>
21 </form>
22 </div>
23 </div>
24 </div>
25 </div>
26 </div>
27 @endsection
```

Coba klik link edit di halaman prodi melalui browser. Apakah data prodi yang mau diedit tampil?



Tambahkan route berikut untuk proses update prodi.

```
Route::get('/prodi', 'ProdiController@index');
Route::get('/prodi/new', 'ProdiController@create');
Route::post('/prodi/new', 'ProdiController@store');
Route::get('/prodi/edit/{id}', 'ProdiController@edit');
Route::post('/prodi/edit/{id}', 'ProdiController@update');
```

Tambahkan kode berikut pada ProdiController.php

```
public function update(Request $request, $id)
{
    //
    $request->validate([
        'kode' => 'required',
        'nama' => 'required'
    ]);

    $prodi = Prodi::find($id);
    $prodi->kode = $request->kode;
    $prodi->nama = $request->nama;

    $prodi->save();

    return redirect('/prodi');
}
```

Coba edit salah satu prodi dan lihat hasilnya.

Tambahkan kode berikut pada index.blade.php untuk menambahkan tombol hapus.

```
<tr>
    <td>{{ $prodi->kode }}</td>
    <td>{{ $prodi->nama }}</td>
    <td>
        <a href="/prodi/edit/{{ $prodi->id }}">Edit</a>
        <form action="/prodi/delete/{{ $prodi->id }}" method="post">
            {{ csrf_field() }}
            {{ method_field('DELETE') }}
            <button type="submit" class="btn btn-danger btn-sm">Hapus</button>
        </form>
    </td>
</tr>
@endforeach
```

Tambahkan route untuk memproses hapus

```
Route::get('/prodi', 'ProdiController@index');
Route::get('/prodi/new', 'ProdiController@create');
Route::post('/prodi/new', 'ProdiController@store');
Route::get('/prodi/edit/{id}', 'ProdiController@edit');
Route::post('/prodi/edit/{id}', 'ProdiController@update');
Route::delete('/prodi/delete/{id}', 'ProdiController@destroy');
```

Tambahkan kode berikut pada ProdiController.php

```
public function destroy($id)
{
    //
    $prodi = Prodi::find($id);
    $prodi->delete();

    return redirect('/prodi');
}
```

Coba lakukan hapus data.

Perhatikan bahwa halaman prodi dapat diakses walaupun user belum login. Agar halaman prodi hanya bisa diakses jika user sudah login, ubah route menjadi seperti berikut.

```
Route::get('/prodi', ['middleware' => 'auth', 'uses' => 'ProdiController@index']);
Route::get('/prodi/new', ['middleware' => 'auth', 'uses' => 'ProdiController@create']);
Route::post('/prodi/new', ['middleware' => 'auth', 'uses' => 'ProdiController@store']);
Route::get('/prodi/edit/{id}', ['middleware' => 'auth', 'uses' => 'ProdiController@edit']);
Route::post('/prodi/edit/{id}', ['middleware' => 'auth', 'uses' => 'ProdiController@update']);
Route::delete('/prodi/delete/{id}', ['middleware' => 'auth', 'uses' => 'ProdiController@destroy']);
```