# Jurus 6:

# Membuat Controller

Pertama-tama, masih tetap dengan jendela command, jalankan perintah

php artisan make:controller Berita

Perintah tersebut akan membuat sebuah file controller baru yang bernama Berita. Jika sudah berhasil, maka file controller tersebut dapat kita lihat di direktori **app/Http/Controllers/**.

php artisan make:controller Berita --resource

Jika menambahkan –resource dibelakang maka controller secara otomatis akan memiliki fungsi seperti API (PUT, POST, DELETE, SHOW, EDIT, STORE).

Jika dibuka, maka kita akan menemukan sebuah class kosong yang dapat kita isi dengan sembarang method.

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use Illuminate\Support\Facades\DB;

use App\Models\Mberita;

use App\Models\Mkategori;

use Response;

use Validator;

use Hash;

use Session;

use Carbon;

use File;

class Berita extends Controller

{

/\*\*

\* Display a listing of the resource.

\*

\* @return \Illuminate\Http\Response

\*/

public function index()

{

$beritas=Mberita::get();

return view('berita.index')->with(['beritas'=>$beritas]);

//

}

/\*\*

\* Show the form for creating a new resource.

\*

\* @return \Illuminate\Http\Response

\*/

public function create()

{

$kategoris=Mkategori::get();

return view('berita.create',compact('kategoris'));

//

}

/\*\*

\* Store a newly created resource in storage.

\*

\* @param \Illuminate\Http\Request $request

\* @return \Illuminate\Http\Response

\*/

public function store(Request $request)

{

$simpan=Mberita::create(['judul' =>$request->judul,

'isi' => $request->isi,

'img\_slider' => $request->img\_slider,

'author' => $request->author,

'idpengguna' => $request->idpengguna,

'kdkategori' => $request->kdkategori,

'gambar' => Berita::simpandokumen($request->file('filedokumen')),

'tanggal' =>\Carbon\Carbon::now(),

'views'=>0

]);

return redirect()->route('berita.index')

->with('success', 'upload berita berhasil');

//

}

public function simpandokumen($filedokumen)

{

$file = $filedokumen;

$tujuan\_upload = 'img';

$acak=rand(0,100);

$extension = $file->getClientOriginalExtension();

$iddokumen=$acak.'.'.$extension;

$file->move($tujuan\_upload,$iddokumen);

return $iddokumen;

}

/\*\*

\* Display the specified resource.

\*

\* @param int $id

\* @return \Illuminate\Http\Response

\*/

public function show($id)

{

$berita=Mberita::with('get\_kategori')->where('id',$id)->get();

return view('berita.show',compact('berita')); //

}

/\*\*

\* Show the form for editing the specified resource.

\*

\* @param int $id

\* @return \Illuminate\Http\Response

\*/

public function edit($id)

{

$beritas=Mberita::find($id);

$kategoris=Mkategori::get();

return view('berita.edit',compact('beritas','kategoris'));

//

}

/\*\*

\* Update the specified resource in storage.

\*

\* @param \Illuminate\Http\Request $request

\* @param int $id

\* @return \Illuminate\Http\Response

\*/

public function update(Request $request, $id)

{

$request->validate([

'judul' => 'required',

'isi' => 'required',

'kdkategori' => 'required',

'author' => 'required'

]);

$databerita=Mberita::find($id);

$databerita->judul=$request->judul;

$databerita->isi=$request->isi;

$databerita->img\_slider=$request->img\_slider;

$uploadedFile = $request->file('filedokumen');

if (!empty($request->file('filedokumen'))) {

$databerita->gambar= Berita::simpandokumen($request->file('filedokumen'));

}

else {

$databerita->gambar=$request->filelama;

}

$databerita->kdkategori=$request->kdkategori;

$databerita->author=$request->author;

$databerita->idpengguna=$request->idpengguna;

$databerita->save();

return redirect()->route('berita.index')

->with('success','Post updated successfully');

//

}

/\*\*

\* Remove the specified resource from storage.

\*

\* @param int $id

\* @return \Illuminate\Http\Response

\*/

public function destroy($id)

{

$beritas=Mberita::find($id);

$datagambar=$beritas->gambar;

$beritas->delete();

File::delete('img/'.$datagambar);

return redirect()->route('berita.index')

->with('success','Post deleted successfully');

//

}

}

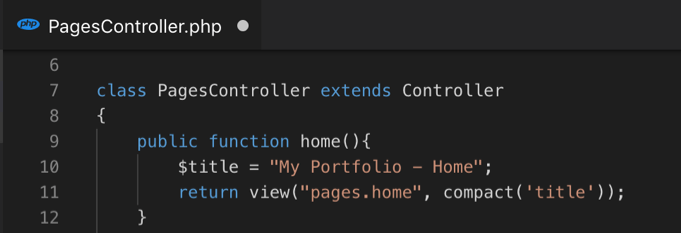
Setelah membuat controller tersebut, di laravel kita WAJIB mendefinisikan route untuk menentukan alamat pemanggilan dan baris kode apa yang akan dipanggil. Route dapat kita buat di  /routes/web.php. Isi route default bisa kita hapus, dan diganti menjadi seperti ini :

Route::resource('berita', Berita::class);

**Mengirim Value ke Blade**

Berikut adalah beberapa cara untuk mengirimkan Value ke Blade:

* Menggunakan compact(‘namavariabel’)



Blade Mneggunakan Compact

* Menggunakan ->with(‘namavar’, ‘vardikirim’). Parameter pertama adalah nama variabel untuk menerima di dalam blade, parameter kedua adalah variable dari value yang akan dikirimkan.



Balade menggunakan With

* Menggunakan ->with(‘namavar’, ‘var\_array\_dikirim’). Parameter pertama adalah nama variabel untuk menerima di dalam blade, parameter kedua adalah variable dari value yang akan dikirimkan berupa array.

