

# Laravel CRUD Step by step

<https://www.youtube.com/watch?v=6uWwJK1ehsM>

Docs by Norizna

1. Create new project cmd(composer create-project laravel/laravel CRUD).

```
C:\laragon\www
λ composer create-project laravel/laravel CRUD2
Creating a "laravel/laravel" project at "./CRUD2"
Installing laravel/laravel (v11.3.3)
- Installing laravel/laravel (v11.3.3): Extracting archive
Created project in C:\laragon\www\CRUD2
> @php -r "file_exists('.env') || copy('.env.example', '.env');" cmd
Loading composer repositories with package information
Updating dependencies
Lock file operations: 107 installs, 0 updates, 0 removals
```

2. Tunggu sampai selesai.

```
INFO Application key set successfully.
> @php -r "file_exists('database/database.sqlite') || touch('database/database.sqlite');"
> @php artisan migrate --graceful --ansi

INFO Preparing database.

Creating migration table ..... 14.74ms DONE

INFO Running migrations.

0001_01_01_000000_create_users_table ..... 33.09ms DONE
0001_01_01_000001_create_cache_table ..... 10.06ms DONE
0001_01_01_000002_create_jobs_table ..... 26.65ms DONE

C:\laragon\www
λ |
```

3. Navigasi kedalam folder, kemudian buka code editor.

```
C:\laragon\www
λ cd crud2

C:\laragon\www\CRUD2
λ code .
```

4. Saat sudah terbuka, navigasi ke folder database, kemudian cari DB\_Connection. Set ke database yang digunakan dan uncomment semua yg ada digambar.

```
23
24 DB_CONNECTION=mysql
25 DB_HOST=127.0.0.1
26 DB_PORT=3306
27 DB_DATABASE=laravel
28 DB_USERNAME=root
29 DB_PASSWORD=
30
```

5. Kemudian dibawahnya ganti baris "SESSION\_DRIVER=database" menjadi "SESSION\_DRIVER=file".

```
31 SESSION_DRIVER=database
32 III
33 vvv
34 SESSION_DRIVER=file
35 SESSION_LIFETIME=120
36 SESSION_ENCRYPT=false
37 SESSION_PATH=/
38 SESSION_DOMAIN=null
39
```

6. Coba jalankan web menggunakan cmd(PHP artisan serve).

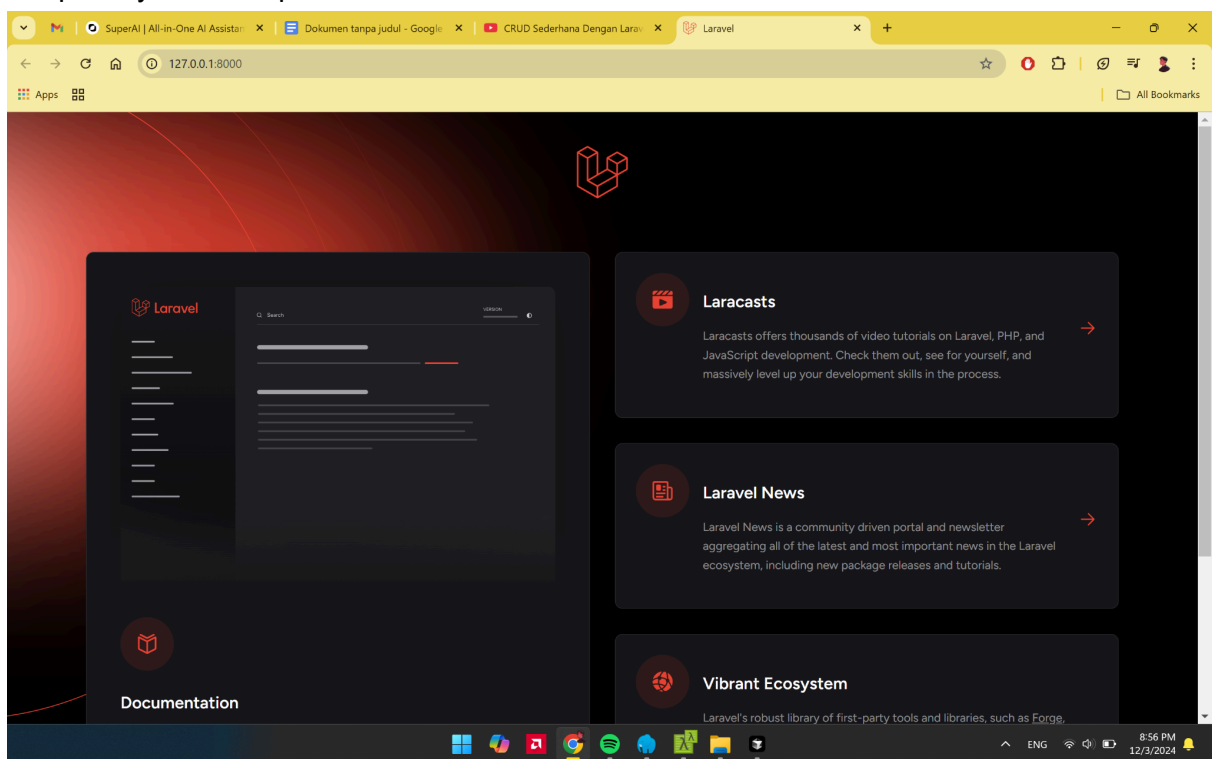
```
C:\laragon\www\CRUD2
λ php artisan serve
forking is not supported on this platform

INFO Server running on [http://127.0.0.1:8000].

Press Ctrl+C to stop the server

2024-12-03 20:52:49 / .....
```

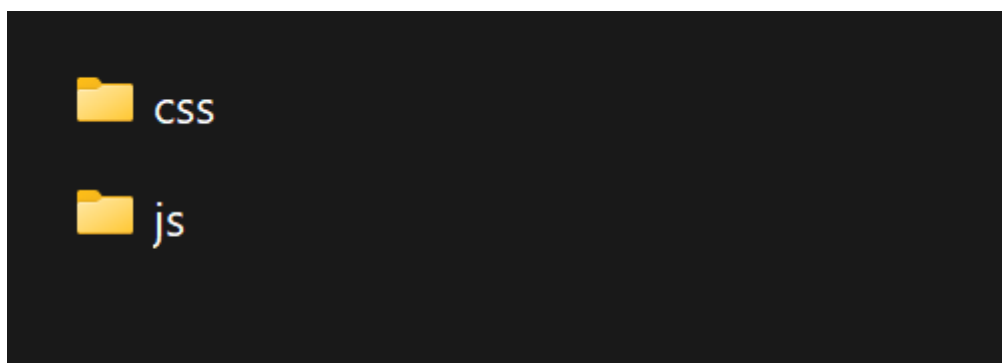
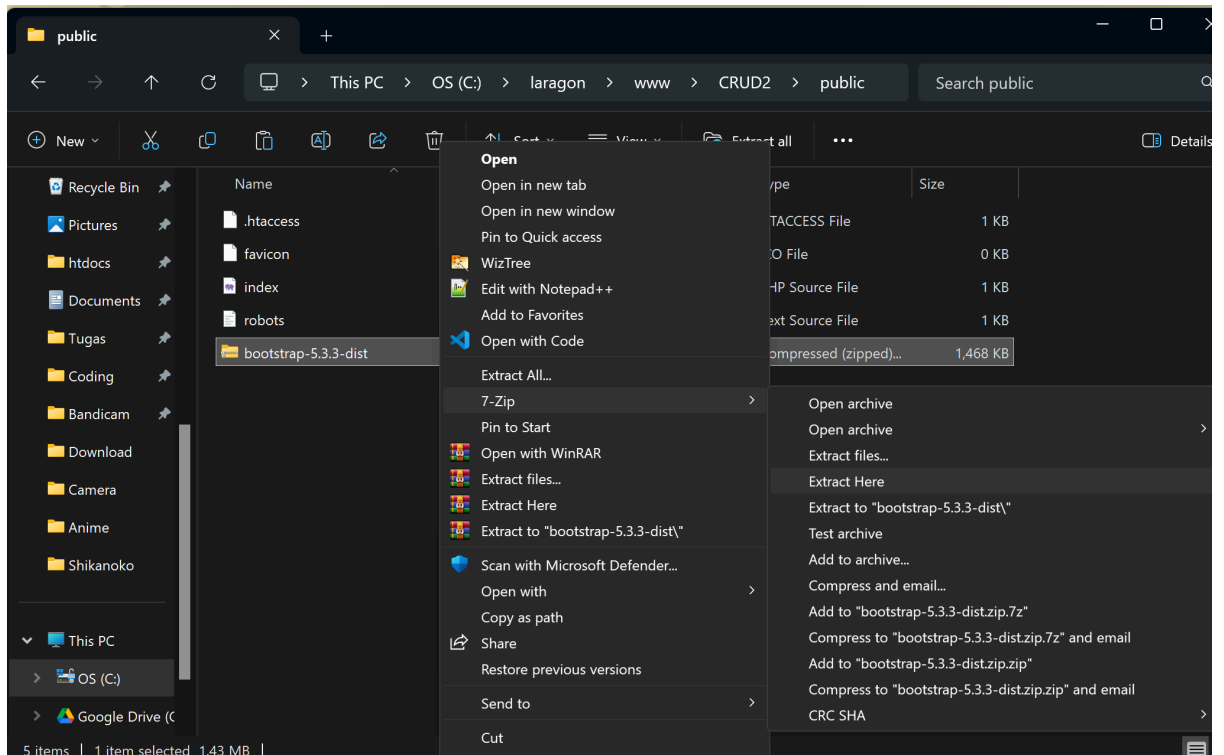
7. Tampilannya akan seperti ini



8. Kemudian karena disini kita akan menggunakan bootstrap, navigasi ke web <https://getbootstrap.com/docs/5.3/getting-started/download/> , kemudian download



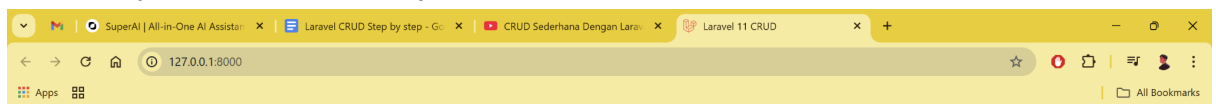
9. Navigasi ke folder codingnya. Kemudian masuk ke *public* paste file tersebut disitu, kemudian extract files dimana saat setelah di extract itu dan dibuka akan ada 2 folder yaitu css dan js.



10. Navigasi ke (resource->views->welcome.blade.php), ini adalah code dari tampilan awal saat masuk laravel
11. Ubah codenya menjadi berikut:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>Laravel 11 CRUD</title>
7     <link rel="stylesheet" href="{{asset(path: 'bootstrap/css/bootstrap.css')}}">
8 </head>
9 <body>
10     <h1 class="text-center mt-3">
11         CRUD Laravel 11
12     </h1>
13 </body>
14 </html>
```

12. Tampilannya akan berubah menjadi berikut:



CRUD Laravel 11



13. Selanjutnya kita akan membuat sebuah controller. Buka cmd didalam direktori folder kodingannya, masukkan (php artisan make:controller SiswaController)

```
C:\laragon\www\CRUD2
λ php artisan make:controller SiswaController

INFO Controller [C:\laragon\www\CRUD2\app\Http\Controllers\SiswaController.php] created successfully.

C:\laragon\www\CRUD2
λ
```

14. Kemudian kita akan membuat Route baru, navigasi ke “routes->web.php”. Kemudian kita masukkan baris code baru:

```
Route::get('/siswa', [SiswaController::class,
'tampil'])->name('siswa.tampil');
```

```
1  <?php
2
3  use App\Http\Controllers\SiswaController;
4  use Illuminate\Support\Facades\Route;
5
6  Route::get(uri: '/', action: function (): Factory|View {
7      return view(view: 'welcome');
8  });
9
10 Route::get(uri: '/siswa', action: [SiswaController::class, 'tampil'])->name(name: 'siswa.tampil');
11  Ctrl+L to chat, Ctrl+K to generate
```

Jangan lupa untuk mengimport packagenya si Controller.

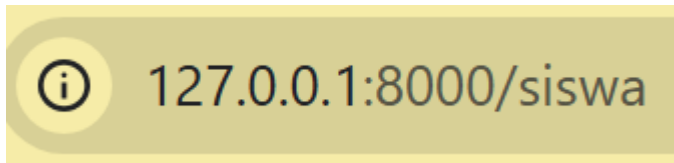
15. Navigasi ke “app => SiswaController => Http => SiswaController”. Kemudian kita akan tambahkan function baru yaitu:

```
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  2 references | 0 implementations
8  class SiswaController extends Controller
9  {
10     1 reference | 0 overrides
11     function tampil(): Factory|View {
12         return view(view: 'siswa.tampil');
13     }
14 }
```

Code ini akan membuka view folder viewsnya siswa, file tampil.

16. Buat file baru di “resource=>views” dengan nama folder “siswa”. Didalamnya folder “siswa” itu kita bikin file baru yaitu “tampil.blade.php”.

17. Kita coba cek di live server kita dengan memasukan url berikut:



18. Apabila tidak terdapat error page maka tidak ada masalah pada codingannya.

19. Next kita akan membuat sebuah layout, yaitu dengan cara membuat file baru di “resource=>views” dengan nama “layout.blade.php”.

20. Pada file layout ini akan kita buat seperti ini.

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Siswa</title>
7      <link rel="stylesheet" href="{{asset(path: 'bootstrap/css/bootstrap.css')}}">
8  </head>
9  <body>
10     <h1 class="text-center mt-3">CRUD Siswa dengan Laravel 11</h1>
11
12     <div class="mt-3 container">|
13         @yield(section: 'konten')
14     </div>
15 </body>
16 </html>
```

@yield("konten") digunakan untuk menentukan tempat di mana konten dari bagian lain (seperti dari file template lain yang menggunakan @section('konten')) akan dimasukkan.

21. Buka file “tampil.blade.php” dan buat seperti berikut

```
1  @extends('view: 'layout')
2
3  @section(section: 'konten')
4
5      <h4 class="text-center">List Siswa</h4>
6
7      <table class="table">
8          <tr>
9              <th>NIS</th>
10             <th>Nama</th>
11             <th>Alamat</th>
12             <th>No Hp</th>
13             <th>Jenis Kelamin</th>
14             <th>Hobi</th>
15         </tr>
16     </table>
17
18 @endsection
```

@extends('layout'): Ini menunjukkan bahwa template ini memperluas (atau menggunakan) layout utama yang disebut layout. Layout ini biasanya berisi struktur HTML dasar yang digunakan di seluruh aplikasi, seperti header, footer, dan elemen navigasi.

@section('konten'): Ini mendefinisikan bagian dari template yang disebut konten. Bagian ini akan diisi dengan konten spesifik yang didefinisikan di antara @section dan @endsection.

22. Kemudian kita perlu membuat model terlebih dahulu di terminal/cmd:

```
C:\laragon\www\CRUD2
λ php artisan make:model Siswa
INFO Model [C:\laragon\www\CRUD2\app\Models\Siswa.php] created successfully.
```

23. Kemudian kita akan perlu membuat tablenya:

```
C:\laragon\www\CRUD2
λ php artisan make:migration buat_tabel_siswa --create=siswa
INFO Migration [C:\laragon\www\CRUD2\database\Migrations\2024_12_04_032430_buat_tabel_siswa.php] created successfully.
```

24. Kemudian navigasi ke file yang baru di migrate tadi  
“database=>migrations=>buat\_tabel\_siswa.php”



25. Next kita akan membuat datanya di dalam tabelnya, yakni:

```
public function up(): void
{
    Schema::create(table: 'siswa', callback: function (Blueprint $table): void {
        $table->id();
        $table->integer(column: 'nis');
        $table->string(column: 'nama');
        $table->string(column: 'alamat');
        $table->string(column: 'no_hp');
        $table->string(column: 'jenis_kelamin');
        $table->string(column: 'hobi');
        $table->timestamps();
    });
}
```

26. Kemudian kita jalankan “php artisan migrate” di terminal:

```
C:\laragon\www\CRUD2
λ php artisan migrate
```

kalau ada message seperti berikut berarti databasenya belum dibuat, jadi kita tinggal type “yes”:

```
C:\laragon\www\CRUD2
λ php artisan migrate

[WARN] The database 'db_laravel_11_2' does not exist on the 'mysql' connection.

Would you like to create it? (yes/no) [yes]
```

27. Berikut adalah gambar yang tampil jika telah berhasil migrate databasenya:

```
C:\laragon\www\CRUD2
λ php artisan migrate

[WARN] The database 'db_laravel_11_2' does not exist on the 'mysql' connection.

Would you like to create it? (yes/no) [yes]
yes

[INFO] Preparing database.

Creating migration table ..... 18.27ms DONE

[INFO] Running migrations.

0001_01_01_000000_create_users_table ..... 60.58ms DONE
0001_01_01_000001_create_cache_table ..... 13.53ms DONE
0001_01_01_000002_create_jobs_table ..... 51.69ms DONE
2024_12_04_032430_buat_tabel_siswa ..... 7.31ms DONE
```

28. Next kita akan meng configure model yang telah kita buat yaitu “Siswa”. Navigasi ke (app/models/Siswa.php).

29. Next kita tambahkan codingnya yaitu:

```
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  0 references | 0 implementations
9  class Siswa extends Model
10 {
11     use HasFactory;
12
13     0 references
14     protected $table = 'siswa';
15 }
```

`protected $table = 'siswa';`: Menentukan bahwa model ini berhubungan dengan tabel siswa di database. Jika nama tabel mengikuti konvensi penamaan Laravel (plural dari nama model), deklarasi ini tidak diperlukan, tetapi dalam kasus ini, mungkin digunakan untuk kejelasan atau jika nama tabel tidak mengikuti konvensi.

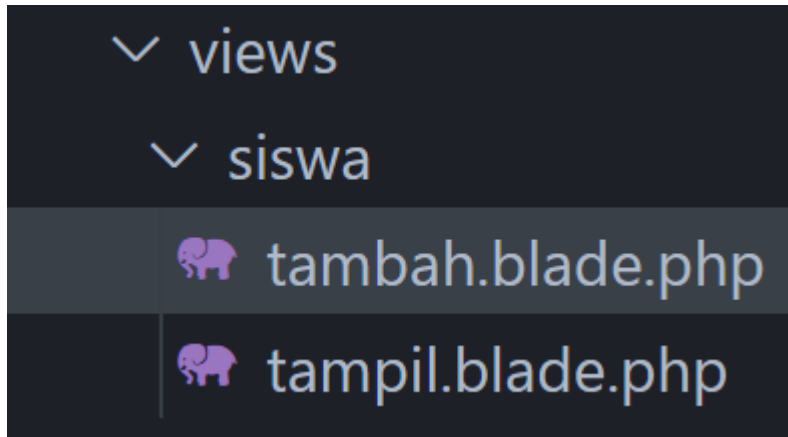
30. Buat *routes* baru, yaitu:

```
10 Route::get(uri: '/siswa', action: [SiswaController::class, 'tampil'])->name(name: 'siswa.tampil');
11 Route::get(uri: '/siswa/tambah', action: [SiswaController::class, 'tambah'])->name(name: 'siswa.tambah');
```

31. Kemudian masuk ke “SiswaController”, dan tambahkan function baru:

```
1 reference | 0 overrides
function tambah(): Factory|View{
    return view(view: 'siswa.tambah');
}
```

32. Buat view baru di folder (views/siswa) dengan nama file (tambah.blade.php).



33. Buat codingan seperti berikut:

```
1  @extends(view: 'layout')
2
3  @section(section: 'konten')
4
5  <h4>Tambah Siswa</h4>
6
7  <form action="" method="post">
8      <label>NIS</label>
9      <input type="number" name="nis" class="form-control mb-2" >
10     <label>Nama</label>
11     <input type="text" name="nama" class="form-control mb-2" >
12     <label>Alamat</label>
13     <input type="text" name="alamat" class="form-control mb-2" >
14     <label>No Hp</label>
15     <input type="text" name="no_hp" class="form-control mb-2" >
16     <label>Jenis Kelamin</label>
17     <input type="text" name="jenis_kelamin" class="form-control mb-2" >
18     <label>Hobi</label>
19     <input type="text" name="hobi" class="form-control mb-2" >
20
21     <button class="btn btn-primary">Tambah</button>
22
23 </form>
24
25 @endsection
```

34. Modifikasi sedikit file “tampil.blade.php”. Bagian di atas tabel kita tambahkan button dan sedikit modif tag h4nya.

```
1  @extends('view: 'layout')
2
3  @section(section: 'konten')
4
5      <div class="d-flex">
6          <h4 class="text-center">List Siswa</h4>
7          <div class="ms-auto">
8              <a class="btn btn-success" href="{{route(name: 'siswa.tambah')}}">Tambah</a>
9          </div>
10     </div>
11
12     <table class="table">
13         <tr>
14             <th>NIS</th>
15             <th>Nama</th>
16             <th>Alamat</th>
17             <th>No Hp</th>
18             <th>Jenis Kelamin</th>
19             <th>Hobi</th>
20         </tr>
21     </table>
22
23 @endsection
```

35. Next kita akan buat *route* baru di folder “routes”:

```
8  });
9
10 Route::get(uri: '/siswa', action: [SiswaController::class, 'tampil'])->name(name: 'siswa.tampil');
11 Route::get(uri: '/siswa/tambah', action: [SiswaController::class, 'tambah'])->name(name: 'siswa.tambah');
12 Route::post(uri: '/siswa/submit', action: [SiswaController::class, 'submit'])->name(name: 'siswa.submit');
```

36. Selanjutnya navigasi ke “StudentController” dan tambahkan function baru:

```
1 reference | 0 overrides
function submit(Request $request): void{
}
}
```

37. Terus navigasi ke views=>siswa=>tambah.blade.php, buat formnya seperti ini:

```
<h4>Tambah Siswa</h4>
<form action="{{route(name: 'siswa.submit')}}" method="post">
    @csrf
    <label>NIS</label>
    <input type="number" name="nis" class="form-control mb-2" >
    <label>Nama</label>
```

38. Lengkapi function submitnya tadi menjadi berikut:

```
1 reference | 0 overrides
function submit(Request $request): mixed | RedirectResponse {
    $siswa = new Siswa();
    $siswa->nis = $request->nis;
    $siswa->nama = $request->nama;
    $siswa->alamat = $request->alamat;
    $siswa->no_hp = $request->no_hp;
    $siswa->jenis_kelamin = $request->jenis_kelamin;
    $siswa->hobi = $request->hobi;
    $siswa->save();

    return redirect()->route(route: 'siswa.tampil');
}
```

39. Kemudian untuk mengecek apakah data yg kita kirim tadi masuk atau tidak, kita bisa melakukan seperti berikut:

```
1 reference | 0 overrides
function tampil(): Factory | View {
    $siswa = Siswa::get();
    dd(vars: $siswa);
    return view(view: 'siswa.tampil');
}
```

Kemudian halaman web view “siswa.tampil”nya akan menjadi seperti ini dan ini menandakan bahwa datanya berhasil dikirim.

```
Illuminate\Database\Eloquent\Collection (#262 // app/Http/Controllers/SiswaController.php:12)
#items: array:1 (*)
0 => App\Model\Siswa (#263 //
#connection: "mysql"
#table: "siswa"
#primaryKey: "id"
#keyType: "int"
#incrementing: true
#with: []
#withCount: []
#preventsLazyLoading: false
#perPage: 15
#exists: true
#wasRecentlyCreated: false
#escapeWhenCastingToString: false
#attributes: array:9 (*)
#original: array:9 (*)
#changes: []
#casts: []
#classCastCache: []
#attributeCastCache: []
#dateFormat: null
#appends: []
#dispatchesEvents: []
#observables: []
#relations: []
#touches: []
#timestamps: true
#usesUniqueIds: false
#hidden: []
#visible: []
#fillable: []
#guarded: array:1 (*)
]
#escapeWhenCastingToString: false
```

40. Selanjutnya delete saja baris yang diblok karena disini kita hanya mengecek apakah datanya berhasil dikirim atau tidak

```
1 reference | 0 overrides
function tampil(): Factory
    $siswa = Siswa::get();
    .....dd(vars: $siswa);
    return view(view: 'siswa.tampil');
}
```

41. Kemudian kita modifikasi lagi function tampil di controllernya:

```
1 reference | 0 overrides
function tampil(): Factory|View {
    $siswa = Siswa::get();
    return view(view: 'siswa.tampil', data: compact(['var_name: 'siswa']));
}
```

42. Next buka view siswa.tampil dan tambahkan codingan baru

```
11
12     <table class="table">
13         <tr>
14             <th>No</th>
15             <th>NIS</th>
16             <th>Nama</th>
17             <th>Alamat</th>
18             <th>No Hp</th>
19             <th>Jenis Kelamin</th>
20             <th>Hobi</th>
21         </tr>
22
23         @foreach ($siswa as $no=>$data)
24             <tr>
25                 <td>{{ $no+1 }}</td>
26                 <td>{{ $data->nis }}</td>
27                 <td>{{ $data->nama }}</td>
28                 <td>{{ $data->alamat }}</td>
29                 <td>{{ $data->no_hp }}</td>
30                 <td>{{ $data->jenis_kelamin }}</td>
31                 <td>{{ $data->hobi }}</td>
32             </tr>
33         @endforeach
34     </table>
35
36 @endsection
```

yaitu dari @foreach sampai @endforeach

43. Maka tampilan pada view “tampil.blade.php” akan menjadi seperti ini:

CRUD Siswa dengan Laravel 11						
List Siswa						
No	NIS	Nama	Alamat	No Hp	Jenis Kelamin	Hobi
1	123213	Noor	Jogja	10239091230	Laki-laki	Lari
2	30322027	Noor Rizky Permana	Jogja	081254284547	Laki-laki	Sepeda

44. Next kita akan menambahkan action edit dan delete.

```
<table class="table">
    <tr>
        <th>No</th>
        <th>NIS</th>
        <th>Nama</th>
        <th>Alamat</th>
        <th>No Hp</th>
        <th>Jenis Kelamin</th>
        <th>Hobi</th>
        <th>Aksi</th>
    </tr>

    @foreach ($siswa as $no=>$data)
        <tr>
            <td>{{ $no+1 }}</td>
            <td>{{ $data->nis }}</td>
            <td>{{ $data->nama }}</td>
            <td>{{ $data->alamat }}</td>
            <td>{{ $data->no_hp }}</td>
            <td>{{ $data->jenis_kelamin }}</td>
            <td>{{ $data->hobi }}</td>
            <td>
                <a href="#">Edit</a>
            </td>
        </tr>
    @endforeach
</table>
```

diatas ini sudah menambahkan baris baru yaitu aksi dan edit

45. Next kita akan modifikasi routesnya

```
13 Route::get(uri: '/siswa/edit', action: [SiswaController::class, 'edit'])->name(name: 'siswa.edit');
```

46. Tambahkan routesnya ke a href yg ada di tampil.blade.php

```
@foreach ($siswa as $no=>$data)
    <tr>
        <td>{{ $no+1 }}</td>
        <td>{{ $data->nis }}</td>
        <td>{{ $data->nama }}</td>
        <td>{{ $data->alamat }}</td>
        <td>{{ $data->no_hp }}</td>
        <td>{{ $data->jenis_kelamin }}</td>
        <td>{{ $data->hobi }}</td>
        <td>
            <a href="{{ route('siswa.edit') }}" class="btn btn-sm btn-warning">Edit</a>
        </td>
    </tr>
@endforeach
```



47. Untuk di routsnya edit itu tambahin yg diblok

```
Route::get(uri: '/siswa/edit/{id}',
```

48. Bagian siswa.tampilnya itu tambahin yg diblok

```
@foreach ($siswa as $no=>$data)
<tr>
    <td>{{$no+1}}</td>
    <td>{{$data->nis}}</td>
    <td>{{$data->nama}}</td>
    <td>{{$data->alamat}}</td>
    <td>{{$data->no_hp}}</td>
    <td>{{$data->jenis_kelamin}}</td>
    <td>{{$data->hobi}}</td>
    <td>
        <a href="{{route('siswa.edit', parameters: $data->id)}}" class="btn btn-sm btn-warning">Edit</a>
    </td>
</tr>
@endforeach
```

49. Next buat function buat editnya:

```
1 reference | 0 overrides
function edit($id): Factory|View{
    $siswa = Siswa::find(id: $id);
    return view(view: 'siswa.edit', data: compact(var_name: 'siswa'));
}
```

50. Tambahkan view baru didalam folder views/siswa yaitu "edit.blade.php"

```
views
└── siswa
    ├── edit.blade.php
    ├── tambah.blade.php
    ├── tampil.blade.php
    ├── layout.blade.php
    └── welcome.blade.php
```

51. Tambahkan code berikut:

```
1  @extends(view: 'layout')
2
3  @section(section: 'konten')
4
5  <h4>Edit Siswa</h4>
6  <form action="{{route(name: 'siswa.update',parameters: $siswa->id)}}" method="post">
7      @csrf
8      <label>NIS</label>
9      <input type="number" name="nis" value="{{ $siswa->nis }}" class="form-control mb-2" >
10     <label>Nama</label>
11     <input type="text" name="nama" value="{{ $siswa->nama }}" class="form-control mb-2" >
12     <label>Alamat</label>
13     <input type="text" name="alamat" value="{{ $siswa->alamat }}" class="form-control mb-2" >
14     <label>No Hp</label>
15     <input type="text" name="no_hp" value="{{ $siswa->no_hp }}" class="form-control mb-2" >
16     <label>Jenis Kelamin</label>
17     <input type="text" name="jenis_kelamin" value="{{ $siswa->jenis_kelamin }}" class="form-control mb-2" >
18     <label>Hobi</label>
19     <input type="text" name="hobi" value="{{ $siswa->hobi }}" class="form-control mb-2" >
20
21     <button class="btn btn-primary">Edit</button>
22
23 </form>
24
25 @endsection
```

52. Tambahkan route baru untuk update

```
Route::post(uri: '/siswa/update/{id}', action: [SiswaController::class, 'update'])->name(name: 'siswa.update');
```

53. Bikin function untuk update

```
function update(Request $request, $id): mixed|RedirectResponse{
    $siswa = Siswa::find(id: $id);
    $siswa->nis = $request->nis;
    $siswa->nama = $request->nama;
    $siswa->alamat = $request->alamat;
    $siswa->no_hp = $request->no_hp;
    $siswa->jenis_kelamin = $request->jenis_kelamin;
    $siswa->hobi = $request->hobi;
    $siswa->update();

    return redirect()->route(route: 'siswa.tampil');
}
```

54. Selanjutnya bikin untuk aksi deletenya. Tambahkan route baru:

```
Route::post(uri: '/siswa/delete/{id}', action: [SiswaController::class, 'delete'])->name(name: 'siswa.delete');
```

55. Tambahkan function baru untuk delete:

```
1 reference | 0 overrides
function delete($id): mixed|RedirectResponse{
    $siswa = Siswa::find(id: $id);
    $siswa->delete();
    return redirect()->route(route: 'siswa.tampil');
}
```

56. Terakhir kita tambahkan buttonnya di “views/siswa/tampil.blade.php”:

```
@foreach ($siswa as $no=>$data)
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| <td>{{ $no+1 }}</td>  <td>{{ $data->nis }}</td>  <td>{{ $data->nama }}</td>  <td>{{ $data->alamat }}</td>  <td>{{ $data->no_hp }}</td>  <td>{{ $data->jenis_kelamin }}</td>  <td>{{ $data->hobi }}</td>  <a href="{{ route('siswa.edit', parameters: $data->id) }}" class="btn btn-sm btn-warning">Edit</a> <form action="{{ route('siswa.delete', parameters: $data->id) }}" method="post"> .....@csrf .....<button class="btn btn-danger btn-sm">Hapus</button> .....</form> | | | | | | | |

</tr>
@endforeach
</table>
```

57. Jejang udah jadi guis prok prok prok 🙌🙌🙌