

Nama : Daniel Susilo  
NIM : G.231.23.0150  
Kelas : Sore A2  
Matkul : Pemrograman Framework

## PRAKTIKUM

### 1. Download Laravel melalui command prompt

```
Command Prompt
Microsoft Windows [Version 10.0.26100.3775]
(c) Microsoft Corporation. All rights reserved.

C:\Users\62821>composer create-project --prefer-dist laravel/laravel data-pegawai
```

### 2. Buka file Laravel data-pegawai yang sudah di unduh

```
Command Prompt
Microsoft Windows [Version 10.0.26100.3775]
(c) Microsoft Corporation. All rights reserved.

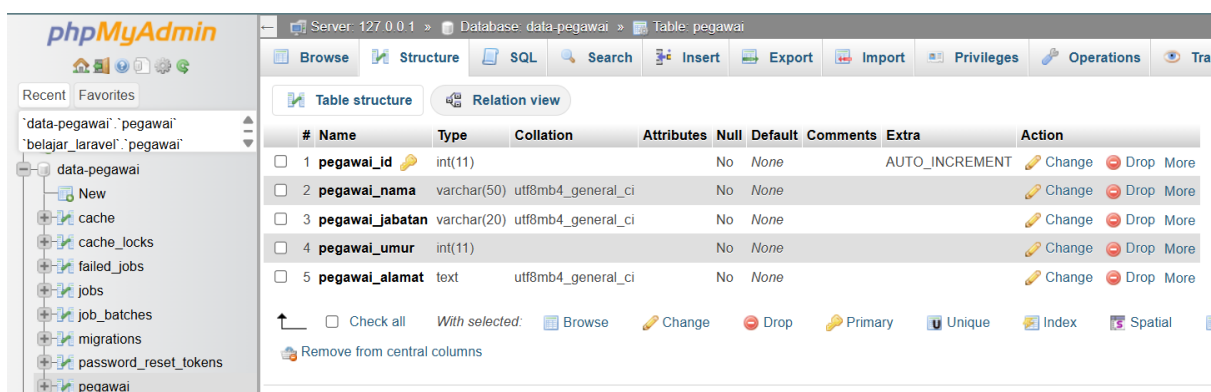
C:\Users\62821>cd C:\xampp\htdocs\data-pegawai
C:\xampp\htdocs\data-pegawai>code .
```

### 3. Konfigurasi database pada .env

```
22
23 DB_CONNECTION=mysql
24 DB_HOST=127.0.0.1
25 DB_PORT=3306
26 DB_DATABASE=data-pegawai
27 DB_USERNAME=root
28 DB_PASSWORD=
29
```

### 4. Siapkan tabel database pada localhost

- Siapkan database dengan nama **data-pegawai**
- Buat tabel dengan nama **pegawai** pengisian tabel pegawai sesuai dengan gambar dibawah



5. Buat route pada `web.php`

```
routes > web.php
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use App\Http\Controllers\PegawaiController;
5
6 Route::get('/', function (): mixed {
7     return view('welcome');
8 });
9
10 //route CRUD
11 Route::get('/pegawai', [PegawaiController::class, 'index']);
```

- Buat `PegawaiController.php` pada Terminal

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\xampp\htdocs\data-pegawai>
PS C:\xampp\htdocs\data-pegawai> php artisan make:controller PegawaiController
```

- Lalu ketik lagi pada terminal untuk membuat `migrate`

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\xampp\htdocs\data-pegawai>
PS C:\xampp\htdocs\data-pegawai> php artisan migrate
```

6. Buka file `PegawaiController.php`, lalu tambahkan perintah yang di beri tanda untuk menjalankan method indeks() pada `Controller`. Hal ini dilakukan untuk mengambil data dari tabel pegawai serta menampilkan data pegawai

```
app > Http > Controllers > PegawaiController.php > PHP > PegawaiController
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use Illuminate\Support\Facades\DB; // ← Tambahkan baris ini di sini
7
8 class PegawaiController extends Controller
9 {
10     1 reference | 0 overrides
11     public function index(): View
12     {
13         // mengambil data dari table pegawai
14         $pegawai = DB::table('pegawai')->get();
15
16         // mengirim data pegawai ke view index
17         return view('index', data: ['pegawai' => $pegawai]);
18     }
19 }
```

7. Buat file `indeks.blade.php` untuk membuat tampilan pada view

```
resources > views > index.blade.php
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Tutorial Membuat CRUD Pada Laravel</title>
5 </head>
6 <body>
7 <h3>Data Pegawai</h3>
8
9 <a href="/pegawai/tambah">+ Tambah Pegawai Baru</a>
10
11 <br/><br/>
12
13 <table border="1">
14 <tr>
15 <th>Nama</th>
16 <th>Jabatan</th>
17 <th>Umur</th>
18 <th>Alamat</th>
19 <th>Ops</th>
20 </tr>
21 @foreach($pegawai as $p)
22 <tr>
23 <td>{{ $p->pegawai_nama }}</td>
24 <td>{{ $p->pegawai_jabatan }}</td>
25 <td>{{ $p->pegawai_umur }}</td>
26 <td>{{ $p->pegawai_alamat }}</td>
27 <td>
28 <a href="/pegawai/edit/{{ $p->pegawai_id }}">Edit</a>
29 |
30 <a href="/pegawai/hapus/{{ $p->pegawai_id }}">Hapus</a>
31 </td>
32 </tr>
33 @endforeach
34 </table>
35
36 </body>
37 </html>
```

8. Akses dari web browser <http://localhost:8000/pegawai>

- Sebelum membuka web pastikan sudah mengaktifkan server php pada terminal

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\xampp\htdocs\data-pegawai>
PS C:\xampp\htdocs\data-pegawai> php artisan serve
```

- Hasil view yang dibuat

localhost / 127.0.0.1 / data-pegawai x Tutorial Membuat CRUD Pada Lar x +

localhost:8000/pegawai

**Data Pegawai**

[+ Tambah Pegawai Baru](#)

Nama	Jabatan	Umur	Alamat	Ops
Dicky	Manager	30	Semarang	<a href="#">Edit</a> <a href="#">Hapus</a>
Andreas	Staff IT	25	Jakarta	<a href="#">Edit</a> <a href="#">Hapus</a>
Irawan	Staff IT	40	Semarang	<a href="#">Edit</a> <a href="#">Hapus</a>

9. Untuk mengakses tambah, edit , dan hapus pegawai perlu menambahkan

- Tambah

```
routes > web.php
1  <?php
2
3  use Illuminate\Support\Facades\Route;
4  use App\Http\Controllers\PegawaiController;
5
6  Route::get(uri: '/', action: function () { View {
7      return view(view: 'welcome');
8  });
9
10 //route CRUD
11 Route::get(uri: '/pegawai', action: [PegawaiController::class, 'index']);
12 //route CRUD Tambah Pegawai
13 Route::get(uri: '/pegawai/tambah', action: [PegawaiController::class, 'tambah']);
```

```
app > Http > Controllers > PegawaiController.php > PHP > PegawaiController
8  class PegawaiController extends Controller
18 // method untuk menampilkan view form tambah pegawai
1 reference | 0 overrides
19 public function tambah(): View
20 {
21     // memanggil view tambah
22     return view(view: 'tambah');
23 }
```

Buat file `tambah.blade.php` untuk views pada menu tambah

```
resources > views > tambah.blade.php
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <title>Tutorial Membuat CRUD Pada Laravel - www.usm.ac.id</title>
5  </head>
6  <body>
7      <h2><a href="https://www.usm.ac.id">www.usm.ac.id</a></h2>
8      <h3>Data Pegawai</h3>
9
10     <a href="/pegawai">kembali</a>
11
12     <br/>
13     <br/>
14
15     <form action="/pegawai/store" method="post">
16         {{ csrf_field() }}
17         Nama <input type="text" name="nama" required="required"><br/>
18         Jabatan <input type="text" name="jabatan" required="required"><br/>
19         Umur <input type="text" name="umur" required="required"><br/>
20         Alamat <textarea name="alamat" required="required"></textarea><br/>
21         <input type="submit" value="Simpan Data">
22     </form>
23
24 </body>
25 </html>
```

1 fitur dari laravel, yaitu csrf protection. csrf protection semacam fitur keamanan untuk pencegahan penginputan data dari luar aplikasi atau sistem kita. Fitur ini sudah include secara otomatis hanya dengan menambahkan perintah `{{ csrf field() }}` seperti di atas, maka laravel akan otomatis meng-generate kode token otomatis, yang di buat dalam bentuk form hidden.

- Tambahkan Store pada web php dan controller untuk menampilkan form tambah data sehingga masuk ke database menggunakan method post

```

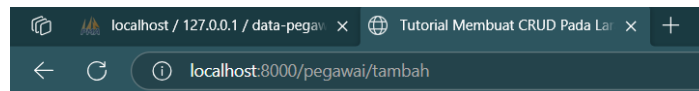
routes > web.php
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use App\Http\Controllers\PegawaiController;
5
6 Route::get(uri: '/', action: function (): View {
7     return view(view: 'welcome');
8 });
9
10 //route CRUD
11 Route::get(uri: '/pegawai', action: [PegawaiController::class, 'index']);
12 //route CRUD Tambah Pegawai
13 Route::get(uri: '/pegawai/tambah', action: [PegawaiController::class, 'tambah']);
14 //route CRUD Pegawai --> store
15 Route::post(uri: '/pegawai/store', action: [PegawaiController::class, 'store']);

```

```

app > Http > Controllers > PegawaiController.php > PHP > PegawaiController
8 class PegawaiController extends Controller
24 // method untuk insert data ke table pegawai
25 public function store(Request $request): RedirectResponse
26 {
27     // insert data ke table pegawai
28     DB::table(table: 'pegawai')->insert(values: [
29         'pegawai_nama' => $request->nama,
30         'pegawai_jabatan' => $request->jabatan,
31         'pegawai_umur' => $request->umur,
32         'pegawai_alamat' => $request->alamat
33     ]);
34
35     // alihkan halaman ke halaman pegawai
36     return redirect(to: '/pegawai');
37 }

```



[www.usm.ac.id](http://www.usm.ac.id)

### Data Pegawai

[kembali](#)

Nama

Jabatan

Umur

Alamat

### • Edit

```

routes > web.php
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use App\Http\Controllers\PegawaiController;
5
6 Route::get(uri: '/', action: function (): View {
7     return view(view: 'welcome');
8 });
9
10 //route CRUD
11 Route::get(uri: '/pegawai', action: [PegawaiController::class, 'index']);
12 //route CRUD Tambah Pegawai
13 Route::get(uri: '/pegawai/tambah', action: [PegawaiController::class, 'tambah']);
14 //route CRUD Pegawai --> store
15 Route::post(uri: '/pegawai/store', action: [PegawaiController::class, 'store']);
16 //route CRUD Edit Pegawai
17 Route::get(uri: '/pegawai/edit/{id}', action: [PegawaiController::class, 'edit']);
18 //route CRUD Update Pegawai

```

```

app > Http > Controllers > PegawaiController.php > PHP > PegawaiController
8 class PegawaiController extends Controller
38 // method untuk edit data pegawai
1 reference|0 overrides
39 public function edit($id): View
40 {
41     // mengambil data pegawai berdasarkan id yang dipilih
42     $pegawai = DB::table('pegawai')->where('pegawai_id', $id)->get();
43     // passing data pegawai yang didapat ke view edit.blade.php
44     return view('edit', ['pegawai' => $pegawai]);
45 }

```

## Buat file edit.blade.php

```

resources > views > edit.blade.php
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Tutorial Membuat CRUD Pada Laravel - www.usm.ac.id</title>
5 </head>
6 <body>
7
8 <h2><a href="https://www.usm.ac.id">www.usm.ac.id</a></h2>
9 <h3>Edit Pegawai</h3>
10
11 <a href="/pegawai">Kembali</a>
12
13 <br/>
14 <br/>
15
16 @foreach($pegawai as $p)
17 <form action="/pegawai/update" method="post">
18     {{ csrf_field() }}
19     <input type="hidden" name="id" value="{{ $p->pegawai_id }}"> <br/>
20     Nama <input type="text" required="required" name="nama" value="{{ $p->pegawai_nama }}"> <br/>
21     Jabatan <input type="text" required="required" name="jabatan" value="{{ $p->pegawai_jabatan }}"> <br/>
22     Umur <input type="number" required="required" name="umur" value="{{ $p->pegawai_umur }}"> <br/>
23     Alamat <textarea required="required" name="alamat" value="{{ $p->pegawai_alamat }}"></textarea> <br/>
24     <input type="submit" value="Simpan Data">
25 </form>
26 @endforeach
27
28 </body>
29 </html>

```

- Agar file dapat terupdate perlu menambahkan update pada web dan controller

```

routes > web.php
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use App\Http\Controllers\PegawaiController;
5
6 Route::get(uri: '/', action: function (): View {
7     return view('welcome');
8 });
9
10 //route CRUD
11 Route::get(uri: '/pegawai', action: [PegawaiController::class, 'index']);
12 //route CRUD Tambah Pegawai
13 Route::get(uri: '/pegawai/tambah', action: [PegawaiController::class, 'tambah']);
14 //route CRUD Pegawai --> store
15 Route::post(uri: '/pegawai/store', action: [PegawaiController::class, 'store']);
16 //route CRUD Edit Pegawai
17 Route::get(uri: '/pegawai/edit/{id}', action: [PegawaiController::class, 'edit']);
18 //route CRUD Update Pegawai
19 Route::post(uri: '/pegawai/update', action: [PegawaiController::class, 'update']);

```

```

app > Http > Controllers > PegawaiController.php > PHP > PegawaiController
8 class PegawaiController extends Controller
46 // update data pegawai
1 reference|0 overrides
47 public function update(Request $request): RedirectResponse
48 {
49     // update data pegawai
50     DB::table('pegawai')
51         ->where('column: 'pegawai_id', operator: $request->id)
52         ->update(values: [
53             'pegawai_nama' => $request->nama,
54             'pegawai_jabatan' => $request->jabatan,
55             'pegawai_umur' => $request->umur,
56             'pegawai_alamat' => $request->alamat
57         ]);
58
59     // alihkan halaman ke halaman pegawai
60     return redirect(to: '/pegawai');
61 }

```

localhost / 127.0.0.1 / data-pegaw x Tutorial Membuat CRUD Pada Laravel x +

localhost:8000/pegawai/edit/1

[www.usm.ac.id](http://www.usm.ac.id)

**Edit Pegawai**

[Kembali](#)

Nama

Jabatan

Umur

Alamat

- **Hapus**  
Pada penggunaan fitur hapus pada laravel cukup menambahkan pada web dan controller saja.

```
routes > web.php
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use App\Http\Controllers\PegawaiController;
5
6 Route::get(uri: '/', action: function () { View {
7     return view(view: 'welcome');
8 }});
9
10 //route CRUD
11 Route::get(uri: '/pegawai', action: [PegawaiController::class, 'index']);
12 //route CRUD Tambah Pegawai
13 Route::get(uri: '/pegawai/tambah', action: [PegawaiController::class, 'tambah']);
14 //route CRUD Pegawai --> store
15 Route::post(uri: '/pegawai/store', action: [PegawaiController::class, 'store']);
16 //route CRUD Edit Pegawai
17 Route::get(uri: '/pegawai/edit/{id}', action: [PegawaiController::class, 'edit']);
18 //route CRUD Update Pegawai
19 Route::post(uri: '/pegawai/update', action: [PegawaiController::class, 'update']);
20 //route CRUD Delete Pegawai
21 Route::get(uri: '/pegawai/hapus/{id}', action: [PegawaiController::class, 'hapus']);
```

```
app > Http > Controllers > PegawaiController.php > PHP > PegawaiController
8 class PegawaiController extends Controller
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63 // method untuk menghapus data pegawai
64 // 1 reference | 0 overrides
65 public function hapus($id): RedirectResponse
66 {
67     // menghapus data pegawai berdasarkan id yang dipilih
68     DB::table('pegawai')->where('pegawai_id', $id)->delete();
69
70     // alihkan halaman ke halaman pegawai
71     return redirect(to: '/pegawai');
72 }
73 }
```