

# **LAPORAN PRAKTIKUM JOBSHEET 3**

**MATA KULIAH PEMROGRAMAN WEB LANJUT  
MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan  
ELOQUENT ORM**

**Dosen Pengampu : Dimas Wahyu Wibowo, S.T., M.T.**



**Di Susun Oleh :**

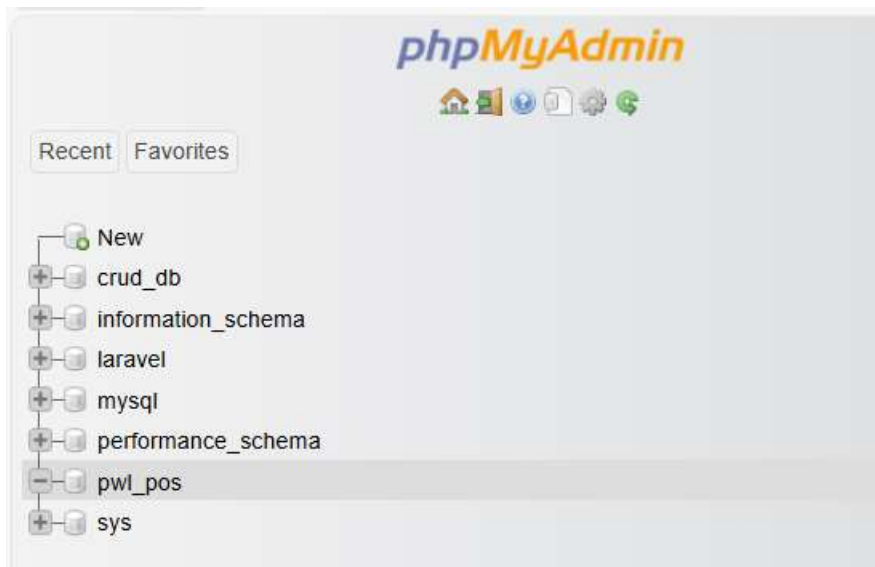
**Louise Nazarossa (2341760117)**

**SIB-2A/18**

**PROGRAM STUDI D4 SISTEM INFORMASI BISNIS  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI MALANG**

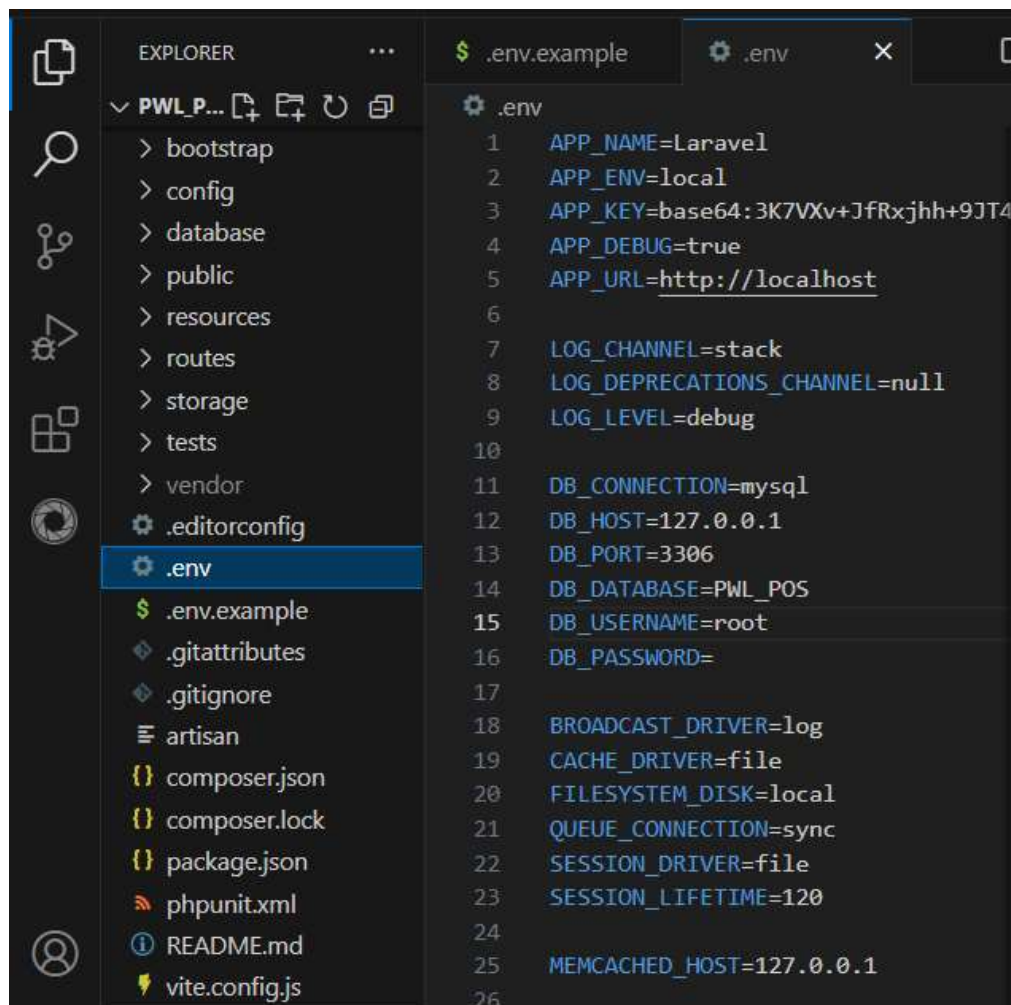
**2024**

## Pengaturan Database



```
C:\laragon\www\PWL_2025\Minggu3>cd PWL_POS
C:\laragon\www\PWL_2025\Minggu3\PWL_POS>php artisan key:generate

INFO Application key set successfully.
```



## Migration

### Membuat file migrasi tanpa relasi

- a.) Membuat file migasi

```
C:\laragon\www\PWL_2025\Minggu3\PWL_POS>php artisan make:migration create_m_level_tabel --create=m_level

INFO Migration [C:\laragon\www\PWL_2025\Minggu3\PWL_POS\database\Migrations\2025_03_05_175458_create_m_level_tabel.php] created successfully.
```

- b.) Menambahkan kode

```
14 Schema::create('m_level', function (Blueprint $table)
15     $table->id('level_id');
16     $table->String('level_kode', 10)->unique();
17     $table->String('level_nama', 100);
18     $table->timestamps();
19 });
```

- c.) Menjalankan perintah pada terminal

```
PS C:\laragon\www\PWL_2025\Minggu3\PWL_POS> php artisan migrate

INFO Preparing database.

Creating migration table ..... 151ms DONE

INFO Running migrations.

2014_10_12_000000_create_users_table ..... 110ms DONE
2014_10_12_100000_create_password_reset_tokens_table 51ms DONE
2019_08_19_000000_create_failed_jobs_table ..... 73ms DONE
2019_12_14_000001_create_personal_access_tokens_table 92ms DONE
E
2025_03_05_175458_create_m_level_tabel ..... 97ms DONE
```

Table	Action
<input type="checkbox"/> failed_jobs	★ Browse Structure Search Insert Empty
<input type="checkbox"/> migrations	★ Browse Structure Search Insert Empty
<input type="checkbox"/> m_level	★ Browse Structure Search Insert Empty
<input type="checkbox"/> password_reset_tokens	★ Browse Structure Search Insert Empty
<input type="checkbox"/> personal_access_tokens	★ Browse Structure Search Insert Empty
<input type="checkbox"/> users	★ Browse Structure Search Insert Empty

d.) Membuat tabel m\_kategori dengan migration

```
php artisan make:migration create_m_kategori_table --create=m_kategori

INFO Migration [C:\laragon\www\PWL_2025\Minggu3\PWL_POS\database\Migrations\2025_03_05_180436_create_m_kategori_table.php] created successfully.

PS C:\laragon\www\PWL_2025\Minggu3\PWL_POS> php artisan migrate

INFO Running migrations.

2025_03_05_180436_create_m_kategori_table 73ms DONE

PS C:\laragon\www\PWL_2025\Minggu3\PWL_POS>
```

Table	Action	Rows	Type
<input type="checkbox"/> failed_jobs	★ Browse Structure Search Insert Empty Drop	0	InnoDB
<input type="checkbox"/> migrations	★ Browse Structure Search Insert Empty Drop	6	InnoDB
<input type="checkbox"/> m_kategori	★ Browse Structure Search Insert Empty Drop	0	InnoDB
<input type="checkbox"/> m_level	★ Browse Structure Search Insert Empty Drop	0	InnoDB

## Membuat file migrasi dengan relasi

a.) Membuat file migrasi untuk tabel m\_user

```
PS C:\laragon\www\PWL_2025\Minggu3\PWL_POS> php artisan make:migration create_m_user_table --table=m_user

INFO Migration [C:\laragon\www\PWL_2025\Minggu3\PWL_POS\database\Migrations\2025_03_05_181100_create_m_user_table.php] created successfully.
```

b.) Modifikasi file untuk tabel m\_user

```
11  */
12  public function up(): void
13  {
14      Schema::create('m_user', function (Blueprint $table) {
15          $table->id('user_id');
16          $table->unsignedBigInteger('level_id')->index();
17          $table->string('username', 20)->unique();
18          $table->string('nama', 100);
19          $table->string('password');
20          $table->timestamps();
21
22          $table->foreign('level_id')->references('level_id')->on('m_level');
23      });
24
25  }
```

```
php artisan migrate

INFO Running migrations.

2025_03_05_181100_create_m_user_table ..... 187ms DONE

PS C:\laragon\www\PWL_2025\Minggu3\PWL_POS> |
```

c.) Membuat tabel database dengan migration untuk tabel-tabel yang memiliki foreign key

```
PS C:\laragon\www\PWL_2025\Minggu3\PWL_POS> php artisan make:migration create_m_barang_table --table=m_barang

INFO Migration [C:\laragon\www\PWL_2025\Minggu3\PWL_POS\database\Migrations\2025_03_05_181547_create_m_barang_table.php] created successfully.

PS C:\laragon\www\PWL_2025\Minggu3\PWL_POS> php artisan make:migration create_t_penjualan_table --table=t_penjualan

INFO Migration [C:\laragon\www\PWL_2025\Minggu3\PWL_POS\database\Migrations\2025_03_05_181610_create_t_penjualan_table.php] created successfully.

PS C:\laragon\www\PWL_2025\Minggu3\PWL_POS> php artisan make:migration create_t_stok_table --table=t_stok

INFO Migration [C:\laragon\www\PWL_2025\Minggu3\PWL_POS\database\Migrations\2025_03_05_181628_create_t_stok_table.php] created successfully.

PS C:\laragon\www\PWL_2025\Minggu3\PWL_POS> php artisan make:migration create_t_penjualan_detail_table --table=t_penjualan_detail

INFO Migration [C:\laragon\www\PWL_2025\Minggu3\PWL_POS\database\Migrations\2025_03_05_181652_create_t_penjualan_detail_table.php] created successfully.
```

# Seeder

## Membuat file seeder

```
PS C:\laragon\www\PWL_2025\Minggu3\PWL_POS> php artisan make:seeder LevelSeeder
```

**INFO** Seeder [C:\laragon\www\PWL\_2025\Minggu3\PWL\_POS\database\seeders\LevelSeeder.php] created successfully.

```
6 use Illuminate\Database\Seeder;
7 use Illuminate\Support\Facades\DB;
8
9
10 class LevelSeeder extends Seeder
11 {
12     /**
13      * Run the database seeds.
14      */
15     public function run(): void
16     {
17         $data = [
18             ['level_id' => 1, 'level_kode' => 'ADM', 'level_nama' => 'Administrator'],
19             ['level_id' => 2, 'level_kode' => 'MNG', 'level_nama' => 'Manager'],
20             ['level_id' => 3, 'level_kode' => 'STF', 'level_nama' => 'Staff/Kasir'],
21         ];
22         DB::table('m_level')->insert($data);
23     }
24 }
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

PS C:\laragon\www\PWL\_2025\Minggu3\PWL\_POS> php artisan db:seed --class=LevelSeeder

**INFO** Seeding database.

<div><div><div></div><div></div><div></div></div></div>				level_id	level_kode	level_nama	created_at	updated_at
<div><div><div></div></div></div>	<div><div></div><div>Edit</div></div>	<div><div></div><div>Copy</div></div>	<div><div></div><div>Delete</div></div>	1	ADM	Administrator	NULL	NULL
<div><div><div></div></div></div>	<div><div></div><div>Edit</div></div>	<div><div></div><div>Copy</div></div>	<div><div></div><div>Delete</div></div>	2	MNG	Manager	NULL	NULL
<div><div><div></div></div></div>	<div><div></div><div>Edit</div></div>	<div><div></div><div>Copy</div></div>	<div><div></div><div>Delete</div></div>	3	STF	Staff/Kasir	NULL	NULL

```
PS C:\laragon\www\PWL_2025\Minggu3\PWL_POS> php artisan db:seed --class=UserSeeder
```

**INFO** Seeding database.



```

16 public function run(): void
17 {
18     'user_id' => 1,
19     'level_id' => 1,
20     'username' => 'admin',
21     'nama' => 'Administrator',
22     'password' => Hash::make('12345'),
23 },
24 [
25     'user_id' => 2,
26     'level_id' => 2,
27     'username' => 'manager',
28     'nama' => 'Manager',
29     'password' => Hash::make('12345'),
30 },
31 [
32     'user_id' => 3,
33     'level_id' => 3,
34     'username' => 'staff',
35     'nama' => 'Staff/Kasir',
36     'password' => Hash::make('12345'),
37 },
38 ],
39 ];
40 }

```

				user_id	level_id	username	nama	password
<input type="checkbox"/>	Edit	Copy	Delete	1	1	admin	Administrator	\$2y\$12\$1dI4Grk3Hx21m3/ny/E/a
<input type="checkbox"/>	Edit	Copy	Delete	2	2	manager	Manager	\$2y\$12\$QIJtRWX/GCPP9Vnfq5X
<input type="checkbox"/>	Edit	Copy	Delete	3	3	staff	Staff/Kasir	\$2y\$12\$NX5gvmaoop2pextdJU4

## DB FACADE

### Implementasi DB Facade

```

$ .env.example web.php M LevelController.php U X
app > Http > Controllers > LevelController.php > LevelController >
5
6 use Illuminate\Http\Request;
7
8 class LevelController extends Controller
9 {
10     public function index()
11     {
12         DB::insert('insert into m_level(level_kode,
13             level_nama, created_at) values(?, ?, ?)',
14             ['CUS', 'Pelanggan', now()]);
15         return "Inserts data baru berhasil";
16     }

```

```
$.env.example web.php M X LevelController.php U
routes > web.php > ...
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use App\Http\Controllers\LevelController;
5
6
7 Route::get('/', function () {
8     return view('welcome');
9 });
10
11 Route::get('/level', [LevelController::class,
12 'index']);
13
```

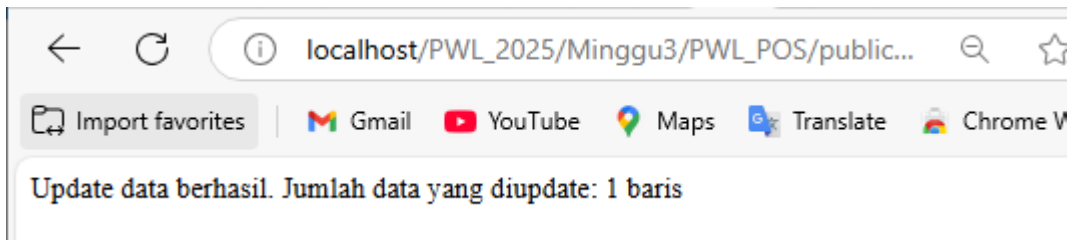
		level_id	level_kode	level_nama	created_at	updated_at
<input type="checkbox"/>	Edit Copy Delete	1	ADM	Administrator	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	2	MNG	Manager	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	3	STF	Staff/Kasir	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	4	CUS	Pelanggan	2025-03-05 18:50:42	NULL

localhost/PWL\_2025/Minggu3/PWL\_POS/public...  
Import favorites Gmail YouTube Maps Translate Chrome Webstore

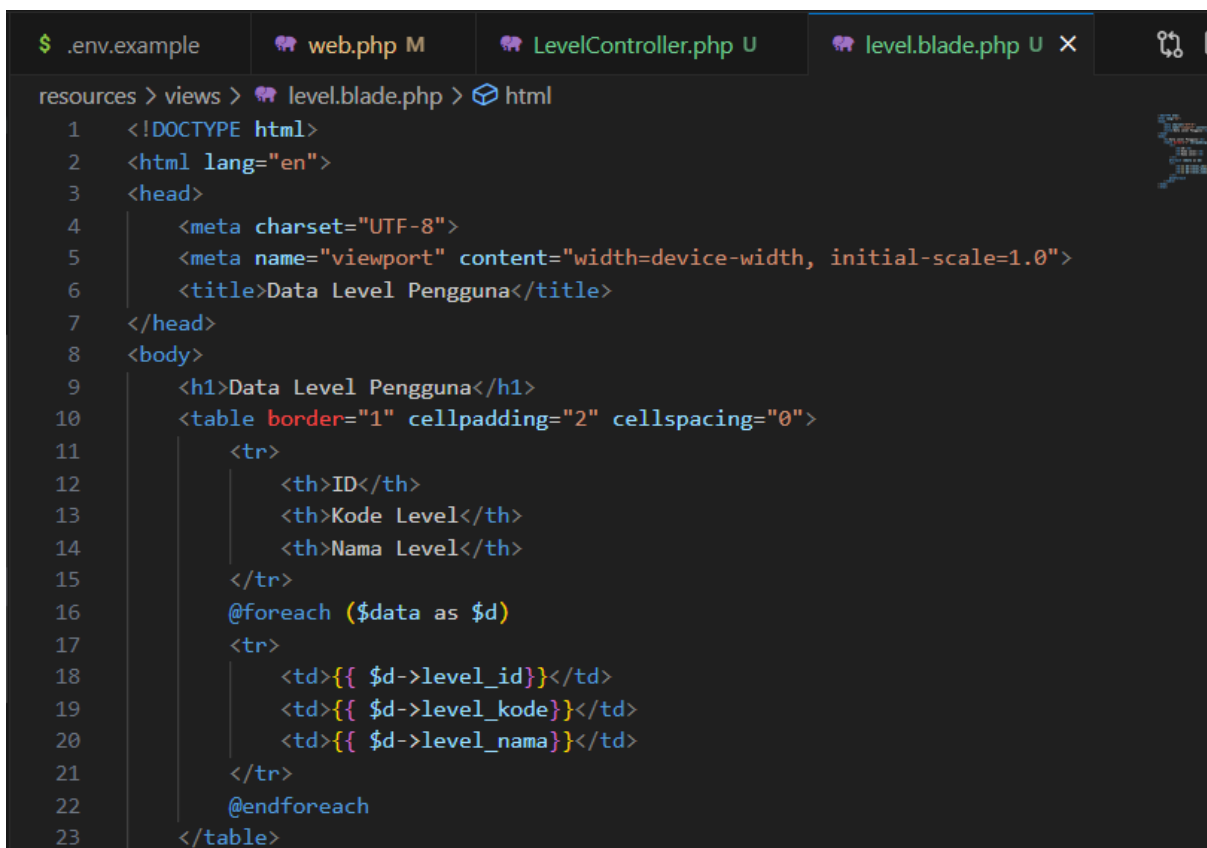
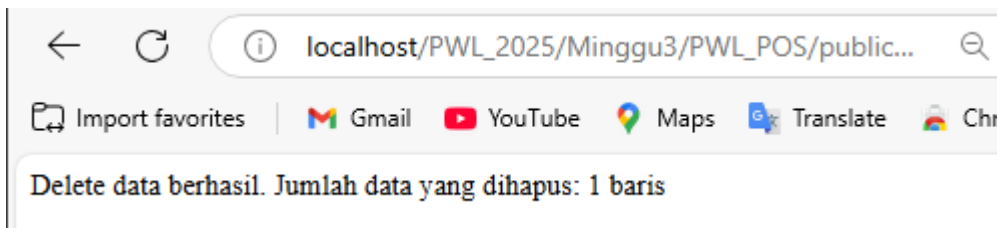
Inserts data baru berhasil

```
17 $row = DB::update('update m_level set
18 level_nama =? where level_kode =?',
19 ['Customer', 'CUS']);
20 return "Update data berhasil.
21 Jumlah data yang diupdate: ". $row.' baris';
22
```





```
18 $row = DB::update('delete from m_level
19 where level_kode = ?', ['CUS']);
20 return 'Delete data berhasil.
21 Jumlah data yang dihapus: ' . $row . ' baris';
```



localhost/PWL\_2025/Minggu3/PWL\_POS/public...

Import favorites | Gmail | YouTube | Maps | Translate | Chrome Webstc

# Data Level Pengguna

ID	Kode Level	Nama Level
1	ADM	Administrator
2	MNG	Manager
3	STF	Staff/Kasir

## Query Builder

### Implementasi Query Builder

web.php M | kategori.blade.php U X | KategoriController.php M | LevelController.php

resources > views > kategori.blade.php > html

```
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>Data Kategori Barang</title>
7 </head>
8 <body>
9     <h1>Data Kategori Barang</h1>
10    <table border="1" cellpadding="2" cellspacing="0">
11        <tr>
12            <th>ID</th>
13            <th>Kode Kategori</th>
14            <th>Nama Kategori</th>
15        </tr>
16        @foreach ($data as $d)
17            <tr>
18                <td>{{ $d->kategori_id }}</td>
19                <td>{{ $d->kategori_kode }}</td>
20                <td>{{ $d->kategori_nama }}</td>
21            </tr>
22        @endforeach
23    </table>
24 </body>
25 </html>
```

```
web.php M x kategori.blade.php U KategoriController.php M x LevelController.php level.blade.php
app > Http > Controllers > KategoriController.php > KategoriController > index
1 <?php
2
3 namespace App\Http\Controllers;
4 use Illuminate\Support\Facades\DB;
5
6 use Illuminate\Http\Request;
7
8 class KategoriController extends Controller
9 {
10     public function index()
11     {
12         //$data = [
13             //'kategori_kode' => 'SNK',
14             //'kategori_nama' => 'Snack/Makanan Ringan',
15             //'created_at' => now()
16         //];
17         //DB::table('m_kategori')->insert($data);
18         //return 'Insert data baru berhasil';
19
20         //$row = DB::table('m_kategori')->where('kategori_kode', 'SNK')->update(['kategori_nama' => 'Cam
21         //return 'Update data berhasil. Jumlah data yang diupdate: ' . $row . ' baris';
22
23         //$row = DB::table('m_kategori')->where('kategori_kode', 'SNK')->delete();
24         //return 'Update data dihapus. Jumlah data yang dihapus: ' . $row . ' baris';
25
26         $data = DB::table('m_kategori')->get();
27         return view('kategori', ['data' => $data]);
28     }
29 }
```

## Floquent ORM

### Implementasi Floquent ORM

```
User.php x UserModel.php U x kategori.blade.php
app > Models > UserModel.php > UserModel
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class UserModel extends Model
9 {
10     use HasFactory;
11
12     protected $table = 'm_user';
13     protected $primaryKey = 'user_id';
14 }
```

PROBLEMS OUTPUT TERMINAL ... powershell + v [ ] [ ] ...

```
PS C:\laragon\www\PWL_2025\Minggu3\PWL_POS>
php artisan make:model UserModel

INFO Model [C:\laragon\www\PWL_2025\Minggu3\PWL_POS\app\Models\UserModel.php] created successfully.
PS C:\laragon\www\PWL_2025\Minggu3\PWL_POS>
```

```
web.php M x UserController.php U x User.php UserModel.php U kategori.blade.php KategoriCon
app > Http > Controllers > UserController.php > UserController
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use App\Models\UserModel;
7  use Illuminate\Support\Facades\Hash;
8
9  class UserController extends Controller
10 {
11     public function index()
12     {
13         // tambah data user dengan Eloquent Model
14         $data = [
15             'nama' => 'Pelanggan Pertama',
16         ];
17         UserModel::where('username', 'customer-1')->update($data); // tambahkan data ke tabel m_user
18
19         //coba akses model UserModel
20         $user = UserModel::all(); // ambil semua data dari tabel m_user
21         return view('user', ['data' => $user]);
22     }
23 }
```

## Penutup

- 1 Pada Praktikum 1 - Tahap 5, apakah fungsi dari APP\_KEY pada file setting .env Laravel?  
**Enkripsi data** seperti sesi, cookie, dan password.  
**Meningkatkan keamanan** dengan memastikan data terenkripsi tidak bisa diubah tanpa key yang valid.
- 2 Pada Praktikum 1, bagaimana kita men-generate nilai untuk APP\_KEY?  
**Menggunakan perintah php artisan key:generate**
- 3 Pada Praktikum 2.1 - Tahap 1, secara default Laravel memiliki berapa file migrasi? dan untuk apa saja file migrasi tersebut?
  - 1 `create_users_table.php` → Untuk membuat tabel pengguna (users).
  - 2 `create_password_resets_table.php` → Untuk menyimpan token reset password.
  - 3 `create_failed_jobs_table.php` → Untuk menyimpan informasi tentang pekerjaan (jobs) yang gagal dijalankan.
- 4 Secara default, file migrasi terdapat kode `$table->timestamps();`, apa tujuan/output dari fungsi tersebut?  
`created_at` → Waktu ketika data pertama kali dibuat.  
`updated_at` → Waktu terakhir kali data diperbarui.  
**Berguna untuk tracking perubahan data secara otomatis.**
- 5 Pada File Migrasi, terdapat fungsi `$table->id();` Tipe data apa yang dihasilkan dari fungsi tersebut?  
**menghasilkan kolom bertipe BIGINT (unsigned) dengan AUTO\_INCREMENT, yang berarti nilai akan bertambah otomatis.**
- 6 Apa bedanya hasil migrasi pada table `m_level`, antara menggunakan `$table->id();` dengan menggunakan `$table->id('level_id');` ?  
`$table->id();` → Laravel otomatis membuat kolom id sebagai primary key dengan tipe BIGINT AUTO\_INCREMENT.  
`$table->id('level_id');` → Laravel membuat primary key dengan nama khusus (`level_id`) tetapi tetap bertipe BIGINT AUTO\_INCREMENT.
- 7 Pada migration, Fungsi `->unique()` digunakan untuk apa?  
**untuk memastikan bahwa nilai dalam kolom tersebut tidak boleh duplikat.**
- 8 Pada Praktikum 2.2 - Tahap 2, kenapa kolom `level_id` pada tabel `m_user` menggunakan `$table->unsignedBigInteger('level_id')`, sedangkan kolom `level_id` pada tabel `m_level` menggunakan `$table->id('level_id')` ?  
**Karena `level_id` di `m_user` adalah foreign key yang merujuk ke `level_id` di `m_level`.**
- 9 Pada Praktikum 3 - Tahap 6, apa tujuan dari Class Hash? dan apa maksud dari kode program `Hash::make('1234');`?  
**Class Hash digunakan untuk mengamankan password dengan teknik hashing.**  
**`Hash::make('1234');` mengubah 1234 menjadi password yang telah dienkripsi menggunakan bcrypt.**
- 10 Pada Praktikum 4 - Tahap 3/5/7, pada query builder terdapat tanda tanya (?), apa kegunaan dari tanda tanya (?) tersebut?  
**Tanda tanya ? digunakan sebagai placeholder untuk parameter dalam query SQL.**

- 11 Pada Praktikum 6 - Tahap 3, apa tujuan penulisan kode `protected $table = 'm_user';` dan `protected $primaryKey = 'user_id';` ?

`protected $table = 'm_user';` → Menentukan nama tabel yang digunakan oleh model.

`protected $primaryKey = 'user_id';` → Menentukan primary key jika tidak menggunakan nama default (id).

- 12 Menurut kalian, lebih mudah menggunakan mana dalam melakukan operasi CRUD ke database (DB Façade / Query Builder / Eloquent ORM) ? jelaskan

Eloquent ORM, karena berbasis objek dan otomatis menangani banyak hal dan mendukung relasi antar tabel