

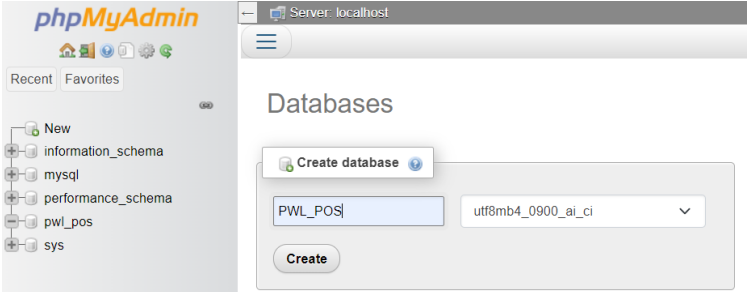


Mata Kuliah : Pemrograman Web Lanjut (PWL)
Program Studi : D4 – Sistem Informasi Bisnis
Semester : 5

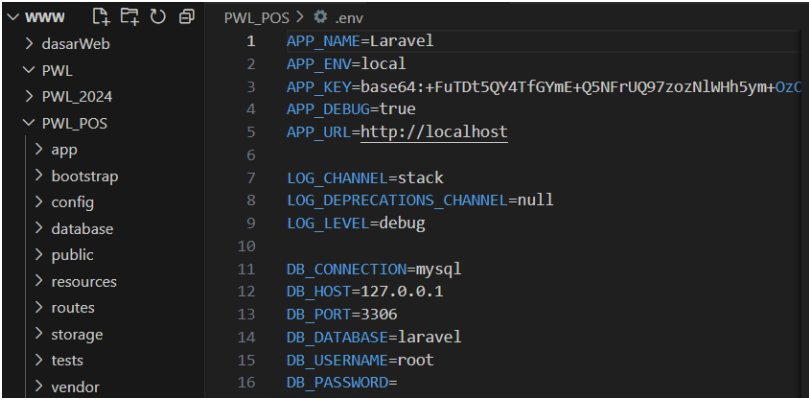
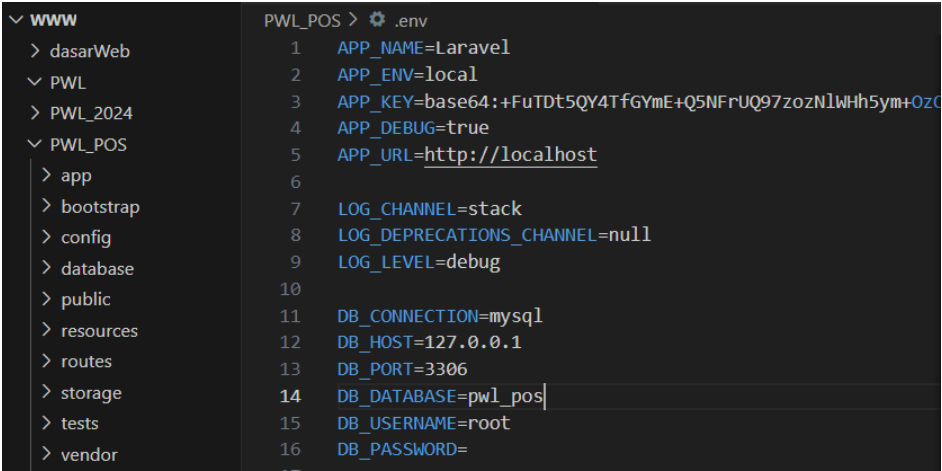
Kelas : SIB
NIM : 2241760007
Nama : Anaradi Octa Lavechia
Jobsheet Ke- : 3 (tiga)

Laporan Jobsheet 03 – MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Praktikum Ke-1

Langkah	Jawaban/Deskripsi
1	Buka aplikasi phpMyAdmin, dan buat database baru dengan nama PWL_POS 
2	Buka aplikasi VSCode dan buka folder project PWL_POS yang sudah kita buat
3	Copy file .env.example menjadi .env
4	Buka file .env, dan pastikan konfigurasi APP_KEY bernilai. Jika belum bernilai silahkan kalian generate menggunakan php artisan.



	
5	<p>Edit file <code>.env</code> dan sesuaikan dengan database yang telah dibuat</p> 
6	<p>Laporkan hasil Praktikum-1 ini dan <code>commit</code> perubahan pada <code>git</code>.</p>

Praktikum Ke-2.1

Langkah	Jawaban/Deskripsi
1	Buka terminal VSCode kalian, untuk yang di kotak merah adalah default dari laravel



2	Kita abaikan dulu yang di kotak merah (jangan di hapus)
3	<p>Kita buat file migrasi untuk table <code>m_level</code> dengan perintah</p> <pre>php artisan make:migration create_m_level_table --create=m_level</pre>

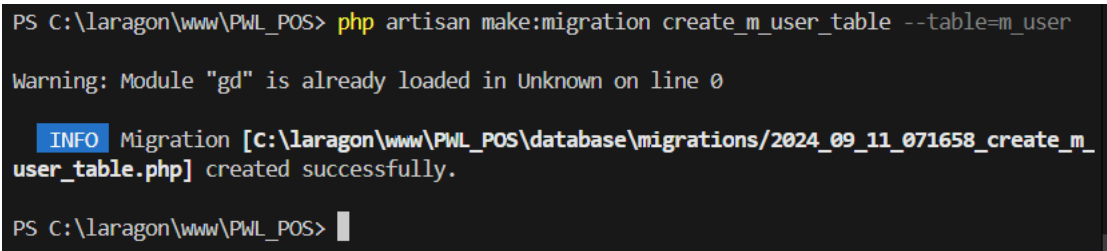


4	<p>Kita perhatikan bagian yang di kotak merah, bagian tersebut yang akan kita modifikasi sesuai desain database yang sudah ada</p> <pre>PWL_POS > database > migrations > 2024_09_11_064710_create_m_level_table.php > class > 1 <?php 2 3 use Illuminate\Database\Migrations\Migration; 4 use Illuminate\Database\Schema\Blueprint; 5 use Illuminate\Support\Facades\Schema; 6 7 return new class extends Migration 8 { 9 /** 10 * Run the migrations. 11 */ 12 public function up(): void 13 { 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 }); 20 }</pre>
5	<p>Simpan kode pada tahapan 4 tersebut, kemudian jalankan perintah ini pada terminal VSCode untuk melakukan migrasi</p> <p>php artisan migrate</p> <pre>▼ TERMINAL powershell + - [x] [x] [x] ... PS C:\laragon\www\PWL_POS> php artisan migrate Warning: Module "gd" is already loaded in Unknown on line 0 [INFO] Preparing database. Creating migration table 37ms DONE [INFO] Running migrations. 2014_10_12_000000_create_users_table 56ms DONE 2014_10_12_100000_create_password_reset_tokens_table 19ms DONE 2019_08_19_000000_create_failed_jobs_table 54ms DONE 2019_12_14_000001_create_personal_access_tokens_table 69ms DONE 2024_09_11_064710_create_m_level_table 37ms DONE PS C:\laragon\www\PWL_POS> </pre>
6	<p>Kemudian kita cek di phpMyAdmin apakah table sudah ter-generate atau belum</p>



	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	5	InnoDB
	<input type="checkbox"/> m_level	★ Browse Structure Search Insert Empty Drop	0	InnoDB
	<input type="checkbox"/> password_reset_tokens	★ Browse Structure Search Insert Empty Drop	0	InnoDB
	<input type="checkbox"/> personal_access_tokens	★ Browse Structure Search Insert Empty Drop	0	InnoDB
	<input type="checkbox"/> users	★ Browse Structure Search Insert Empty Drop	0	InnoDB
7	Ok, table sudah dibuat di database			
8	Buat table database dengan migration untuk table m_kategori dan m_supplier yang sama-sama tidak memiliki foreign key			
9	Laporkan hasil Praktikum-2.1 ini dan commit perubahan pada git.			

Praktikum Ke-2.2

Langkah	Jawaban/Deskripsi
1	<p>Buka terminal VSCode kalian, dan buat file migrasi untuk table m_user</p> <pre>php artisan make:migration create_m_user_table --table=m_user</pre>  <pre>PS C:\laragon\www\PWL_POS> php artisan make:migration create_m_user_table --table=m_user Warning: Module "gd" is already loaded in Unknown on line 0 INFO Migration [C:\laragon\www\PWL_POS\database\migrations\2024_09_11_071658_create_m_user_table.php] created successfully. PS C:\laragon\www\PWL_POS></pre>
2	Buka file migrasi untuk table m_user, dan modifikasi seperti berikut



	<div><div>PWL_POS > database > migrations > 2024_09_11_071658_create_m_user_table.php > class > down</div><div><pre>9 return new class extends Migration 10 { 11 /** 12 * Run the migrations. 13 */ 14 public function up(): void 15 { 16 Schema::create('m_user', function (Blueprint \$table) { 17 \$table->id('user_id'); 18 \$table->unsignedBigInteger('level_id')->index(); //Indexing untuk ForeignKey 19 \$table->string('username', 20)->unique(); //unique untuk memastikan tidak ada use 20 \$table->string('nama', 100); 21 \$table->string('password'); 22 \$table->timestamps(); 23 24 // Mendefinisikan Foreign Key pada kolom level_id mengacu pada kolom level_id di 25 \$table->foreign('level_id')->references('level_id')->on('m_level'); 26 }); 27 } 28 29 /** 30 * Reverse the migrations. 31 */ 32 public function down(): void 33 { 34 Schema::dropIfExists('m_user'); 35 } 36 };</pre></div></div>				
3	<div><p>Simpan kode program Langkah 2, dan jalankan perintah php artisan migrate.</p><p>Amati apa yang terjadi pada database.</p><div><pre>PS C:\laragon\www\PWL_POS> php artisan migrate Warning: Module "gd" is already loaded in Unknown on line 0 [INFO] Running migrations. 2024_09_11_071658_create_m_user_table 158ms DONE PS C:\laragon\www\PWL_POS> </pre></div></div>				
4	<div><p>Buat table database dengan migration untuk table-tabel yang memiliki foreign key</p><table><tr><td>m_barang</td></tr><tr><td>t_penjualan</td></tr><tr><td>t_stok</td></tr><tr><td>t_penjualan_detail</td></tr></table><p>- Table m_kategori</p></div>	m_barang	t_penjualan	t_stok	t_penjualan_detail
m_barang					
t_penjualan					
t_stok					
t_penjualan_detail					



```
PS C:\laragon\www\PWL_POS> php artisan make:migration create_m_kategori_table --table=m_kategori
```

Warning: Module "gd" is already loaded in Unknown on line 0

INFO Migration [C:\laragon\www\PWL_POS\database\Migrations\2024_09_13_134557_create_m_kategori_table.php] created successfully.

```
database > migrations > 2024_09_13_135034_create_m_kategori_table.php > class > up > Closure
1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  return new class extends Migration
8  {
9      /**
10       * Run the migrations.
11       */
12     public function up(): void
13     {
14         Schema::create(table: 'm_kategori', callback: function (Blueprint $table): void {
15             $table->id(column: 'kategori_id');
16             $table->string(column: 'kategori_kode', length: 10);
17             $table->string(column: 'kategori_nama', length: 100);
18             $table->timestamps();
19         });
20     }
21
22     /**
23      * Reverse the migrations.
24      */
25     public function down(): void
26     {
27         Schema::dropIfExists(table: 'm_kategori', callback: function (Blueprint $table): void {
28             //
29         });
30     }
31 };
```

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

Warning: Module "gd" is already loaded in Unknown on line 0

INFO Running migrations.

2024_09_13_135034_create_m_kategori_table 18ms **DONE**

- Table m_barang

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

Warning: Module "gd" is already loaded in Unknown on line 0

INFO Migration [C:\laragon\www\PWL_POS\database\Migrations\2024_09_13_131430_create_m_barang_table.php] created successfully.



```
database > migrations > 2024_09_13_135034_create_m_kategori_table.php > ...
1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  return new class extends Migration
8  {
9      /**
10       * Run the migrations.
11       */
12     public function up(): void
13     {
14         Schema::create(table: 'm_barang', callback: function (Blueprint $table): void {
15             $table->id(column: 'barang_id');
16             $table->unsignedBigInteger(column: 'kategori_id')->index(); // Foreign key ke tabel m_kategori
17             $table->string(column: 'barang_kode', length: 10);
18             $table->string(column: 'barang_nama', length: 100);
19             $table->integer(column: 'harga_beli');
20             $table->integer(column: 'harga_jual');
21             $table->timestamps();
22
23             // Mendefinisikan Foreign Key
24             $table->foreign(columns: 'kategori_id')->references(columns: 'kategori_id')->on(table: 'm_kategori');
25         });
26     }
27
28     /**
29      * Reverse the migrations.
30      */
31     public function down(): void
32     {
33         Schema::dropIfExists(table: 'm_barang');
34     }
35 };
```

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

Warning: Module "gd" is already loaded in Unknown on line 0

INFO Running migrations.

2024_09_13_140100_create_m_barang_table ..... 2ms DONE
```

- Table t_stok

```
PS C:\laragon\www\PWL_POS> php artisan make:migration create_t_stok_table --table=t_stok
Warning: Module "gd" is already loaded in Unknown on line 0

INFO Migration [C:\laragon\www\PWL_POS\database\Migrations\2024_09_13_141617_create_t_stok_table.php] created successfully.
```




```
database > migrations > 2024_09_13_141617_create_t_stok_table.php > ...
1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  return new class extends Migration
8  {
9      /**
10       * Run the migrations.
11       */
12     public function up(): void
13     {
14         Schema::create(table: 't_stok', callback: function (Blueprint $table): void {
15             $table->id(column: 'stok_id');
16             $table->unsignedBigInteger(column: 'barang_id')->index(); // Foreign key ke tabel m_barang
17             $table->unsignedBigInteger(column: 'user_id')->index(); // Foreign key ke tabel m_user
18             $table->dateTime(column: 'stok_tanggal');
19             $table->integer(column: 'stok_jumlah');
20             $table->timestamps();
21
22             // Mendefinisikan Foreign Key
23             $table->foreign(columns: 'barang_id')->references(columns: 'barang_id')->on(table: 'm_barang');
24             $table->foreign(columns: 'user_id')->references(columns: 'user_id')->on(table: 'm_user');
25         });
26     }
27
28     /**
29      * Reverse the migrations.
30      */
31     public function down(): void
32     {
33         Schema::dropIfExists(table: 't_stok');
34     }
35 };
```

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

Warning: Module "gd" is already loaded in Unknown on line 0

INFO Running migrations.

2024_09_13_141617_create_t_stok_table ..... 206ms DONE
```

- Table t_penjualan

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

Warning: Module "gd" is already loaded in Unknown on line 0

INFO Migration [C:\laragon\www\PWL_POS\database\Migrations\2024_09_13_141845_create_t_penjualan_table.php] created successfully.
```

```
php artisan migrate

n\www\PWL_POS>

Warning: Module "gd" is already loaded in Unknown on line 0

INFO Running migrations.

2024_09_13_141845_create_t_penjualan_table ..... 14ms DONE
```



- Table t_penjualan_detail

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

Warning: Module "gd" is already loaded in Unknown on line 0

INFO Migration [C:\laragon\www\PWL_POS\database\Migrations\2024_09_13_143019_create_t_penjualan_detail_table.php] created successfully.
```

```
database > migrations > 2024_09_13_143019_create_t_penjualan_detail_table.php > class > up
1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  return new class extends Migration
8  {
9      /**
10       * Run the migrations.
11       */
12       public function up(): void
13       {
14           Schema::create(table: 't_penjualan_detail', callback: function (Blueprint $table): void {
15               $table->id(column: 'detail_id');
16               $table->unsignedBigInteger(column: 'penjualan_id')->index();
17               $table->unsignedBigInteger(column: 'barang_id')->index();
18               $table->integer(column: 'harga');
19               $table->integer(column: 'jumlah');
20               $table->timestamps();
21
22               $table->foreign(columns: 'penjualan_id')->references(columns: 'penjualan_id')->on(table: 't_penjualan');
23               $table->foreign(columns: 'barang_id')->references(columns: 'barang_id')->on(table: 'm_barang');
24           });
25       }
26
27       /**
28        * Reverse the migrations.
29        */
30       public function down(): void
31       {
32           Schema::dropIfExists(table: 't_penjualan_detail');
33       }
34     };
```

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

Warning: Module "gd" is already loaded in Unknown on line 0

INFO Running migrations.

2024_09_13_143019_create_t_penjualan_detail_table ..... 143ms DONE
```

- Table m_supplier

```
PS C:\laragon\www\PWL_POS> php artisan make:migration create_m_supplier_table
--table=m_supplier

Warning: Module "gd" is already loaded in Unknown on line 0

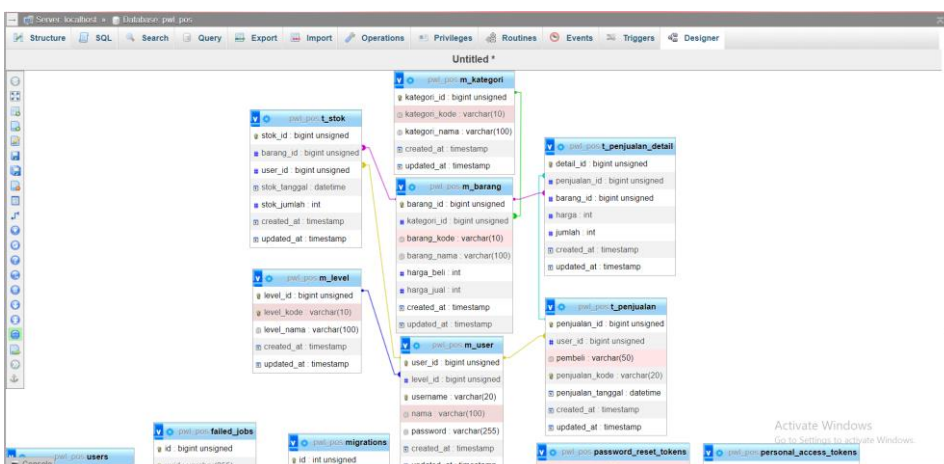
INFO Migration [C:\laragon\www\PWL_POS\database\Migrations\2024_09_13_153909_create_m_supplier_table.php] created successfully.
```



```
database > migrations > 2024_09_13_153909_create_m_supplier_table.php > class
1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  return new class extends Migration
8  {
9      /**
10       * Run the migrations.
11       */
12      public function up(): void
13      {
14          Schema::create(table: 'm_supplier', callback: function (Blueprint $table): void {
15              $table->id(column: 'supplier_id');
16              $table->string(column: 'supplier_kode',length: 10)->unique();
17              $table->string(column: 'supplier_nama',length: 100);
18              $table->string(column: 'supplier_alamat',length: 225);
19              $table->timestamps();
20          });
21      }
22
23      /**
24       * Reverse the migrations.
25       */
26      public function down(): void
27      {
28          Schema::dropIfExists(table: 'm_supplier');
29      }
30  };
```

```
Warning: Module "gd" is already loaded in Unknown on line 0
[INFO] Migration [C:\laragon\www\PWL_POS\database\migrations\2024_09_13_153909_create_m_supplier_table.php] created successfully.
php artisan migrate
Warning: Module "gd" is already loaded in Unknown on line 0
[INFO] Running migrations.
2024_09_13_153909_create_m_supplier_table ..... 162ms DONE
```

- 5 Jika semua file migrasi sudah di buat dan dijalankan maka bisa kita lihat tampilan designer pada phpMyAdmin seperti berikut





6	Laporkan hasil Praktikum-2.2 ini dan commit perubahan pada git.
---	---

Praktikum Ke-3

Langkah	Jawaban/Deskripsi
1	<p>Kita akan membuat file seeder untuk table <code>m_level</code> dengan mengetikkan perintah php artisan make:seeder LevelSeeder</p> 
2	<p>Selanjutnya, untuk memasukkan data awal, kita modifikasi file tersebut di dalam function <code>run()</code></p> 
3	<p>Selanjutnya, kita jalankan file <i>seeder</i> untuk table <code>m_level</code> pada terminal dengan perintah <code>php artisan db:seed --class=LevelSeeder</code></p> 
4	<p>Ketika seeder berhasil dijalankan maka akan tampil data pada table <code>m_level</code></p>



	<div><pre>SELECT * FROM `m_level`</pre><div><input type="checkbox"/> Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]</div><div><input type="checkbox"/> Show all Number of rows: <input type="text" value="25"/> Filter rows: <input type="text" value="Search this table"/> Sort by key:</div><div>Extra options</div><table><tr><th></th><th>level_id</th><th>level_kode</th><th>level_nama</th><th>created_at</th><th>updated_at</th></tr><tr><td><input type="checkbox"/> Edit Copy Delete</td><td>1</td><td>ADM</td><td>Administrator</td><td>NULL</td><td>NULL</td></tr><tr><td><input type="checkbox"/> Edit Copy Delete</td><td>2</td><td>MNG</td><td>Manager</td><td>NULL</td><td>NULL</td></tr><tr><td><input type="checkbox"/> Edit Copy Delete</td><td>3</td><td>STF</td><td>Staff/Kasir</td><td>NULL</td><td>NULL</td></tr></table></div>		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
	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																				
5	<p>Sekarang kita buat file <i>seeder</i> untuk table <code>m_user</code> yang me-refer ke table <code>m_level</code> dengan perintah <code>php artisan make:seeder UserSeeder</code></p> <pre>PS C:\laragon\www\PWL_POS> php artisan make:seeder UserSeeder Warning: Module "gd" is already loaded in Unknown on line 0 INFO Seeder [C:\laragon\www\PWL_POS\database\seeders\UserSeeder.php] created successfully.</pre>																								
6	Modifikasi file class <code>UserSeeder</code> seperti berikut																								



	<pre>database > seeders > UserSeeder.php > PHP Intelephense > UserSeeder > run 1 <?php 2 3 namespace Database\Seeders; 4 5 use Illuminate\Database\Console\Seeds\WithoutModelEvents; 6 use Illuminate\Database\Seeder; 7 use Illuminate\Support\Facades\DB; 8 use Illuminate\Support\Facades\Hash; 9 10 0 references 0 implementations 11 class UserSeeder extends Seeder 12 { 13 0 references 0 overrides 14 public function run(): void 15 { 16 \$data = [17 [18 'user_id'=>1, 19 'level_id'=>1, 20 'username'=>'admin', 21 'nama'=>'Administrator', 22 'password'=> Hash::make(value: '12345'), //class untuk mengenkripsi/hash password 23], 24 [25 'user_id'=>2, 26 'level_id'=>2, 27 'username'=>'manager', 28 'nama'=>'Manager', 29 'password'=> Hash::make(value: '12345'), 30], 31 [32 'user_id'=>3, 33 'level_id'=>3, 34 'username'=>'staff', 35 'nama'=>'Staff/Kasir', 36 'password'=> Hash::make(value: '12345'), 37] 38]; 39 DB::table(table: 'm_user')->insert(values: \$data); 40 } 41 }</pre>
7	<p>Jalankan perintah untuk mengeksekusi class UserSeeder</p> <pre>PS C:\laragon\www\PWL_POS> php artisan db:seed --class=UserSeeder Warning: Module "gd" is already loaded in Unknown on line 0 [INFO] Seeding database.</pre>
8	<p>Perhatikan hasil seeder pada table m_user</p>



	<div><div><div>←T→</div><div>user_idlevel_idusernamenamapassword</div><div><div><input type="checkbox"/></div><div>Edit</div><div>Copy</div><div>Delete</div></div><div>11adminAdministrator\$2y\$12\$KJjuSr.BSon0RkeRR1Ap4.4dfzDk2zVCVud..msxhRG...</div></div><div><div><input type="checkbox"/></div><div>Edit</div><div>Copy</div><div>Delete</div></div><div>22managerManager\$2y\$12\$tLNbjRw9HtnHX5nL5iQe7urgFoJSdZXdgVez9zHZJK6...</div></div> <div><div><input type="checkbox"/></div><div>Edit</div><div>Copy</div><div>Delete</div></div> <div>33staffStaff/Kasir\$2y\$12\$5/fgslWS9EcYYK7eVIFfNeV1RpTXjbr8OE2.4TPySN...</div>
--	---



```
PS C:\laragon\www\PWL_POS> php artisan make:seeder SupplierSeeder
```

```
Warning: Module "gd" is already loaded in Unknown on line 0
```

```
INFO Seeder [C:\laragon\www\PWL_POS\database\seeders\SupplierSeeder.php]  
created successfully.
```

```
database > seeders > SupplierSeeder.php > PHP Intelephense > SupplierSeeder > run
```

```
1  <?php
2
3  namespace Database\Seeders;
4
5  use Illuminate\Database\Console\Seeds\WithoutModelEvents;
6  use Illuminate\Database\Seeder;
7  use Illuminate\Support\Facades\DB;
8
9  0 references | 0 implementations
10 class SupplierSeeder extends Seeder
11 {
12     /**
13      * Run the database seeds.
14      */
15     0 references | 0 overrides
16     public function run(): void
17     {
18         $data = [
19             [
20                 'supplier_id'=>1,
21                 'supplier_kode'=>'yol',
22                 'supplier_nama'=>'Yolanda Supplier',
23                 'supplier_alamat'=>'Jln Hamidrusdi No.102',
24             ],
25             [
26                 'supplier_id'=>2,
27                 'supplier_kode'=>'snd',
28                 'supplier_nama'=>'Sendi Supplier',
29                 'supplier_alamat'=>'Jln Buring No.54',
30             ],
31             [
32                 'supplier_id'=>3,
33                 'supplier_kode'=>'liy',
34                 'supplier_nama'=>'Liya Supplier',
35                 'supplier_alamat'=>'Jln Tuban no 65',
36             ]
37         ];
38         DB::table('m_supplier')->insert(values: $data);
39     }
40 }
```




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

```
Warning: Module "gd" is already loaded in Unknown on line 0
```

```
INFO Seeding database.
```

	supplier_id	supplier_kode	supplier_nama	supplier_alamat	created_at	updated_at
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	yol	Yolanda Supplier	Jln Hamidrusdi No.102	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	snd	Sendi Supplier	Jln Buring No.54	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	liy	Liya Supplier	Jln Tuban no 65	NULL	NULL

- Isi tabel Barang

```
PS C:\laragon\www\PWL_POS> php artisan make:seeder BarangSeeder
```

```
Warning: Module "gd" is already loaded in Unknown on line 0
```

```
INFO Seeder [C:\laragon\www\PWL_POS\database\seeders\BarangSeeder.php] created successfully.
```

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

```
Warning: Module "gd" is already loaded in Unknown on line 0
```

```
INFO Seeding database.
```

	barang_id	kategori_id	barang_kode	barang_nama	harga_beli	harga_jual	created_at	updated_at
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	1	nas	Nasi	10000	12000	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	1	aym	Ayam	8000	10000	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	1	saw	Sawi	5000	7000	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	2	pmg	Promag	8000	10000	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	2	scr	Sunscreen	12000	15000	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	2	dix	Dulcolax	8000	10000	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	7	3	myt	Minyak Telon	15000	19000	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	8	3	pop	Popok Bayi	8000	20000	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	9	3	bdk	Bedak Bayi	8000	10000	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	10	4	krs	Kursi Plastik	20000	22000	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	11	4	pel	Pel	8000	10000	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	12	4	sap	Sapu	8000	10000	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	13	5	lmp	Lampu	8000	10000	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	14	5	stk	Stop Kontak	8000	10000	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	15	5	kbl	Kabel	8000	10000	NULL	NULL

- Isi tabel Stok



```
PS C:\laragon\www\PWL_POS> php artisan make:seeder StokSeeder
```

```
Warning: Module "gd" is already loaded in Unknown on line 0
```

```
INFO Seeder [C:\laragon\www\PWL_POS\database\seeders\StokSeeder.php] created successfully.
```

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

```
Warning: Module "gd" is already loaded in Unknown on line 0
```

```
INFO Seeding database.
```

					stok_id	barang_id	supplier_id	user_id	stok_tanggal	stok_jumlah	created_at	updated_at
<input type="checkbox"/>	Edit	Copy	Delete		1	1	1	3	2024-09-11 00:00:00	4	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete		2	2	1	3	2024-09-11 00:00:00	5	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete		3	1	1	3	2024-09-11 00:00:00	4	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete		4	2	1	3	2024-09-11 00:00:00	4	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete		5	3	1	3	2024-09-11 00:00:00	4	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete		6	4	2	3	2024-09-11 00:00:00	4	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete		7	5	2	3	2024-09-11 00:00:00	4	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete		8	6	2	3	2024-09-11 00:00:00	4	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete		9	7	2	3	2024-09-11 00:00:00	4	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete		10	8	2	3	2024-09-11 00:00:00	4	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete		11	9	3	3	2024-09-11 00:00:00	4	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete		12	10	3	3	2024-09-11 00:00:00	4	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete		13	11	3	3	2024-09-11 00:00:00	4	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete		14	12	3	3	2024-09-11 00:00:00	4	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete		15	13	3	3	2024-09-11 00:00:00	4	NULL	NULL

- Isi tabel Penjualan

```
PS C:\laragon\www\PWL_POS> php artisan make:seeder PenjualanSeeder
```

```
Warning: Module "gd" is already loaded in Unknown on line 0
```

```
INFO Seeder [C:\laragon\www\PWL_POS\database\seeders\PenjualanSeeder.php] created successfully.
```

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

```
Warning: Module "gd" is already loaded in Unknown on line 0
```

```
INFO Seeding database.
```



		penjualan_id	user_id	pembeli	penjualan_kode	penjualan_tanggal	created_at	updated_at
<input type="checkbox"/>	Edit Copy Delete	1	3	Nela	aaa	2024-09-11 00:00:00	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	2	3	Nara	aab	2024-09-11 00:00:00	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	3	3	Lala	aba	2024-09-11 00:00:00	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	4	3	Chia	abc	2024-09-11 00:00:00	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	5	3	Nela	acb	2024-09-11 00:00:00	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	6	3	Nela	acc	2024-09-11 00:00:00	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	7	3	Nara	ana	2024-09-11 00:00:00	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	8	3	Neva	ama	2024-09-11 00:00:00	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	9	3	Nela	omo	2024-09-11 00:00:00	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	10	3	Nela	umu	2024-09-11 00:00:00	NULL	NULL

- Isi tabel Penjualan_detail

```
PS C:\laragon\www\PWL_POS> php artisan make:seeder DetailPenjualanSeeder

Warning: Module "gd" is already loaded in Unknown on line 0

[INFO] Seeder [C:\laragon\www\PWL_POS\database\seeders\DetailPenjualanSeeder.php] created successfully.
```



```
database > seeders > DetailPenjualanSeeder.php > PHP Intelephense > DetailPenjualanSeeder > run
1  <?php
2
3  namespace Database\Seeders;
4
5  use Illuminate\Database\Console\Seeds\WithoutModelEvents;
6  use Illuminate\Database\Seeder;
7  use Illuminate\Support\Facades\DB;
8
9  0 references | 0 implementations
10 class DetailPenjualanSeeder extends Seeder
11 {
12     /**
13      * Run the database seeds.
14      */
15     0 references | 0 overrides
16     public function run(): void
17     {
18         $data = [];
19
20         $detail_id = 1;
21         for ($penjualan_id = 1; $penjualan_id <= 10; $penjualan_id++) {
22             for ($barang_id = 1; $barang_id <= 3; $barang_id++) {
23                 $data[] = [
24                     'detail_id' => $detail_id,
25                     'penjualan_id' => $penjualan_id,
26                     'barang_id' => $barang_id,
27                     'harga' => rand(min: 5000, max: 15000),
28                     'jumlah' => rand(min: 1, max: 10),
29                 ];
30                 $detail_id++;
31             }
32         }
33
34         DB::table('t_penjualan_detail')->insert(values: $data);
35     }
36 }
```

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

Warning: Module "gd" is already loaded in Unknown on line 0

INFO Seeding database.
```

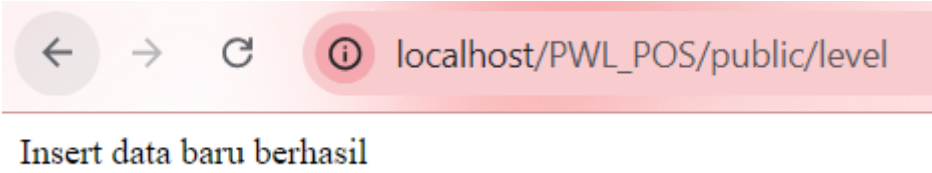


			detail_id	penjualan_id	barang_id	harga	jumlah	created_at	updated_at					
	<input type="checkbox"/>		Edit		Copy		Delete	1	1	1	10588	4	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	2	1	2	8523	3	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	3	1	3	5615	10	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	4	2	1	7633	5	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	5	2	2	7789	10	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	6	2	3	9942	7	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	7	3	1	7921	4	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	8	3	2	7418	4	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	9	3	3	14221	5	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	10	4	1	12405	4	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	11	4	2	13574	1	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	12	4	3	8812	3	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	13	5	1	10465	3	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	14	5	2	13213	2	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	15	5	3	7606	7	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	16	6	1	12951	2	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	17	6	2	11677	3	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	18	6	3	7052	6	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	19	7	1	6308	3	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	20	7	2	8610	5	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	21	7	3	13260	3	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	22	8	1	9177	2	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	23	8	2	12588	9	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	24	8	3	7159	1	NULL	NULL
	<input type="checkbox"/>		Edit		Copy		Delete	25	9	1	14770	8	NULL	NULL
11	Laporkan hasil Praktikum-3 ini dan commit perubahan pada git.													

Praktikum Ke-4

Langkah	Jawaban/Deskripsi
1	Kita buat controller dahulu untuk mengelola data pada table m_level dengan perintah php artisan make:controller LevelController



	<pre>PS C:\laragon\www\PWL_POS> php artisan make:controller LevelController Warning: Module "gd" is already loaded in Unknown on line 0 INFO Controller [C:\laragon\www\PWL_POS\app\Http\Controllers\LevelController.php] created successfully.</pre>
2	<p>Kita modifikasi dulu untuk routing-nya, ada di PWL_POS/routes/web.php</p> <pre>routes > web.php > ... 1 <?php 2 3 use App\Http\Controllers\LevelController; 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: '/level', action: [LevelController::class, 'index']); 11 12</pre>
3	<p>Selanjutnya, kita modifikasi file LevelController untuk menambahkan 1 data ke table m_level</p> <pre>app > Http > Controllers > LevelController.php > PHP Intelephense > LevelController > index 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use Illuminate\Http\Request; 6 use Illuminate\Support\Facades\DB; 7 8 2 references 0 implementations 9 class LevelController extends Controller 10 { 11 1 reference 0 overrides 12 public function index(): string{ 13 DB::insert(query: 'insert into m_level(level_id, level_kode, level_nama, created_at) values(?,?,?,?)',bindings: [4,'CUS','Pelanggan',now()]); 14 return 'Insert data baru berhasil'; 15 } 16 }</pre>
4	<p>Kita coba jalankan di browser dengan url localhost/PWL_POS/public/level dan amati apa yang terjadi pada table m_level di database, <i>screenshot</i> perubahan yang ada pada table m_level</p> 



← T →

	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	2024-09-14 02:27:37	NULL

5

Selanjutnya, kita modifikasi lagi file `LevelController` untuk meng-*update* data di table `m_level` seperti berikut

```
app > Http > Controllers > LevelController.php > PHP Intelephense > LevelController > index
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use Illuminate\Support\Facades\DB;
7
8  2 references | 0 implementations
9  class LevelController extends Controller
10 {
11     1 reference | 0 overrides
12     public function index(): string{
13         // DB::insert('insert into m_level(level_id, level_kode, level_nama, created_at) values(?, ?, ?, ?)', [4, 'CUS', 'Pelanggan', now()]);
14         // return 'Insert data baru berhasil';
15
16         $row = DB::update(query: 'update m_level set level_nama = ? where level_kode = ?', bindings: ['Customer', 'CUS']);
17         return 'Update data berhasil. Jumlah data yang diupdate: ' . $row . ' baris';
18     }
19 }
```

6

Kita coba jalankan di browser dengan url `localhost/PWL_POS/public/level` lagi dan amati apa yang terjadi pada table `m_level` di database, screenshot perubahan yang ada pada table `m_level`

← → ↻ ⓘ localhost/PWL_POS/public/level

Update data berhasil. Jumlah data yang diupdate: 1 baris

← T →

	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	Customer	2024-09-14 02:27:37	NULL

7

Kita coba modifikasi lagi file `LevelController` untuk melakukan proses hapus data



```
app > Http > Controllers > LevelController.php > PHP Intelephense > LevelController > index
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use Illuminate\Support\Facades\DB;
7
8 2 references | 0 implementations
9 class LevelController extends Controller
10 {
11     1 reference | 0 overrides
12     public function index(): string
13     {
14         // DB::insert('insert into m_level(level_id, level_kode, level_nama, created_at) values(?, ?, ?, ?)', [4, 'CUS', 'Pelanggan', now()]);
15         // return 'Insert data baru berhasil';
16
17         // $row = DB::update('update m_level set level_nama = ? where level_kode = ?', ['Customer', 'CUS']);
18         // return 'Update data berhasil. Jumlah data yang diupdate: ' . $row.' baris';
19
20         $row = DB::delete('delete from m_level where level_kode = ?', ['CUS']);
21         return 'Delete data berhasil. Jumlah data yang dihapus: ' . $row.' baris';
22     }
23 }
```

localhost/PWL_POS/public/level

Delete data berhasil. Jumlah data yang dihapus: 1 baris

	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

- 8 Method terakhir yang kita coba adalah untuk menampilkan data yang ada di table m_level. Kita modifikasi file LevelController seperti berikut

```
app > Http > Controllers > LevelController.php > PHP Intelephense > LevelController > index
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use Illuminate\Support\Facades\DB;
7
8 2 references | 0 implementations
9 class LevelController extends Controller
10 {
11     1 reference | 0 overrides
12     public function index(): Factory|View
13     {
14         // DB::insert('insert into m_level(level_id, level_kode, level_nama, created_at) values(?, ?, ?, ?)', [4, 'CUS', 'Pelanggan', now()]);
15         // return 'Insert data baru berhasil';
16
17         // $row = DB::update('update m_level set level_nama = ? where level_kode = ?', ['Customer', 'CUS']);
18         // return 'Update data berhasil. Jumlah data yang diupdate: ' . $row.' baris';
19
20         // $row = DB::delete('delete from m_level where level_kode = ?', ['CUS']);
21         // return 'Delete data berhasil. Jumlah data yang dihapus: ' . $row.' baris';
22
23         $data = DB::select('select * from m_level');
24         return view('level', data: ['data' => $data]);
25     }
26 }
```




9

Coba kita perhatikan kode yang diberi tanda kotak merah, berhubung kode tersebut memanggil `view('level')`, maka kita buat file view pada VSCode di `PWL_POS/resources/view/level.blade.php`

```
resources > views > level.blade.php > html > body > table
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Data Level Pengguna</title>
5      </head>
6      <body>
7          <h1>Data Level Pengguna</h1>
8          <table border="1" cellpadding="2" cellspacing="0">
9              <tr>
10                 <th>ID</th>
11                 <th>Kode Level</th>
12                 <th>Nama Level</th>
13             </tr>
14             @foreach ( $data as $d )
15                 <tr>
16                     <td>{{ $d->level_id }}</td>
17                     <td>{{ $d->level_kode }}</td>
18                     <td>{{ $d->level_nama }}</td>
19                 </tr>
20             @endforeach
21         </table>
22     </body>
23 </html>
```

10

Silahkan dicoba pada browser dan amati apa yang terjadi

← → ↻ ⓘ localhost/PWL_POS/public/level

Data Level Pengguna

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

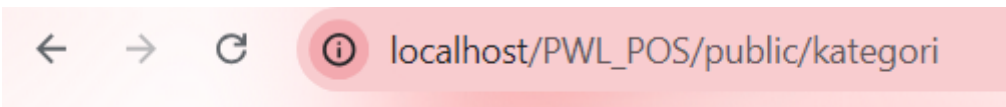


11	Laporkan hasil Praktikum-4 ini dan commit perubahan pada git.
----	---

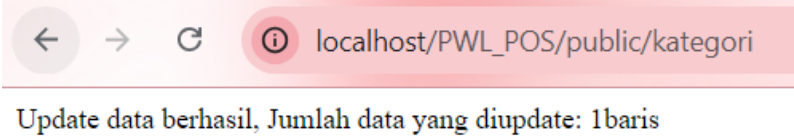
Praktikum Ke-5

Langkah	Jawaban/Deskripsi
1	<p>Kita buat controller dahulu untuk mengelola data pada table m_kategori</p> <pre>PS C:\laragon\www\PWL_POS> php artisan make:controller KategoriController</pre> <p>Warning: Module "gd" is already loaded in Unknown on line 0</p> <p>INFO Controller [C:\laragon\www\PWL_POS\app\Http\Controllers\KategoriController.php] created successfully.</p>
2	<p>Kita modifikasi dulu untuk routing-nya, ada di PWL_POS/routes/web.php</p> <pre>routes > web.php > ... 1 <?php 2 3 use App\Http\Controllers\KategoriController; 4 use App\Http\Controllers\LevelController; 5 use Illuminate\Support\Facades\Route; 6 7 Route::get(uri: '/', action: function (): Factory View{ 8 return view(view: 'welcome'); 9 }); 10 11 Route::get(uri: '/level', action: [LevelController::class, 'index']); 12 Route::get(uri: '/kategori', action: [KategoriController::class, 'index']); 13 14 ?></pre>
3	<p>Selanjutnya, kita modifikasi file KategoriController untuk menambahkan 1 data ke table m_kategori</p>



	<pre>app > Http > Controllers > KategoriController.php > PHP Intelephense > KategoriController > 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use Illuminate\Http\Request; 6 use Illuminate\Support\Facades\DB; 7 8 2 references 0 implementations 9 class KategoriController extends Controller 10 { 11 1 reference 0 overrides 12 public function index(): string 13 { 14 \$data =[15 'kategori_kode'=>'SNK', 16 'kategori_nama'=>'Snack/ Makanan ringan', 17 'created_at'=>now() 18]; 19 DB::table('m_kategori')->insert(values: \$data); 20 return 'Insert data baru berhasil'; 21 } 22 }</pre>
4	<p>Kita coba jalankan di browser dengan url localhost/PWL_POS/public/kategori dan amati apa yang terjadi pada table m_kategori di database, screenshot perubahan yang ada pada table m_kategori</p>  <p>Insert data baru berhasil</p>
5	<p>Selanjutnya, kita modifikasi lagi file KategoriController untuk meng-update data di table m_kategori seperti berikut</p>



	<pre>app > Http > Controllers > KategoriController.php > PHP Intelephense > KategoriController > index 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use Illuminate\Http\Request; 6 use Illuminate\Support\Facades\DB; 7 8 2 references 0 implementations 9 class KategoriController extends Controller 10 { 11 1 reference 0 overrides 12 public function index(): string 13 { 14 // \$data = [15 // 'kategori_kode'=>'SNK', 16 // 'kategori_nama'=>'Snack/ Makanan ringan', 17 // 'created_at'=>now() 18 //]; 19 // DB::table('m_kategori')->insert(\$data); 20 // return 'Insert data baru berhasil'; 21 22 \$row = DB::table('m_kategori')->where(column: 'kategori_kode', operator: 'SNK')->update(values: ['kategori_nama'=>'Camilan']); 23 return 'Update data berhasil, Jumlah data yang diupdate: '.\$row.'baris'; 24 } 25 }</pre>
6	<p>Kita coba jalankan di browser dengan url localhost/PWL_POS/public/kategori lagi dan amati apa yang terjadi pada table m_kategori di database, screenshot perubahan yang ada pada table m_kategori</p> 
7	<p>Kita coba modifikasi lagi file KategoriController untuk melakukan proses hapus data</p> <pre>app > Http > Controllers > KategoriController.php > PHP Intelephense > KategoriController > index 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use Illuminate\Http\Request; 6 use Illuminate\Support\Facades\DB; 7 8 2 references 0 implementations 9 class KategoriController extends Controller 10 { 11 1 reference 0 overrides 12 public function index(): string 13 { 14 // \$data = [15 // 'kategori_kode'=>'SNK', 16 // 'kategori_nama'=>'Snack/ Makanan ringan', 17 // 'created_at'=>now() 18 //]; 19 // DB::table('m_kategori')->insert(\$data); 20 // return 'Insert data baru berhasil'; 21 22 // \$row = DB::table('m_kategori')->where('kategori_kode', 'SNK')->update(['kategori_nama'=>'Camilan']); 23 // return 'Update data berhasil, Jumlah data yang diupdate: '.\$row.'baris'; 24 25 \$row = DB::table('m_kategori')->where(column: 'kategori_kode', operator: 'SNK')->delete(); 26 return 'Delete anda berhasil. Jumlah data yang dihapus '.\$row.' baris'; 27 } 28 }</pre>



	<div><div>← → ↺ ⓘ localhost/PWL_POS/public/kategori</div><div>Delete data berhasil. Jumlah data yang dihapus 0 baris</div></div>
8	<p>Method terakhir yang kita coba adalah untuk menampilkan data yang ada di table m_kategori. Kita modifikasi file KategoriController seperti berikut</p> <pre>app > Http > Controllers > KategoriController.php > PHP Intelephense > KategoriController > index 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use Illuminate\Http\Request; 6 use Illuminate\Support\Facades\DB; 7 8 2 references 0 implementations 9 class KategoriController extends Controller 10 { 11 1 reference 0 overrides 12 public function index(): Factory View 13 { 14 // \$data = [15 // 'kategori_kode'=>'SNK', 16 // 'kategori_nama'=>'Snack/ Makanan ringan', 17 // 'created_at'=>now() 18 //]; 19 // DB::table('m_kategori')->insert(\$data); 20 // return 'Insert data baru berhasil'; 21 22 // \$row = DB::table('m_kategori')->where('kategori_kode', 'SNK')->update(['kategori_nama'=>'Camilan']); 23 // return 'Update data berhasil, Jumlah data yang diupdate: '.\$row.' baris'; 24 25 // \$row = DB::table('m_kategori')->where('kategori_kode', 'SNK')->delete(); 26 // return 'Delete data berhasil. Jumlah data yang dihapus '.\$row.' baris'; 27 28 \$data = DB::table('m_kategori')->get(); 29 return view('kategori', data: ['data'=>\$data]); 30 } 31 }</pre>
9	<p>Coba kita perhatikan kode yang diberi tanda kotak merah, berhubung kode tersebut memanggil view('kategori'), maka kita buat file view pada VSCode di PWL_POS/resources/view/kategori.blade.php</p>



resources > views > kategori.blade.php > html > body > table > tr > th

```

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

```

10

Silahkan dicoba pada browser dan amati apa yang terjadi.

←

→

↺

ⓘ localhost/PWL_POS/public/kategori

Data Kategori Barang

ID	Kode Kategori	Nama Kategori
1	fnb	Food and Baverage
2	bnh	Beauty and Health
3	bnk	Baby and Kid
4	hnc	Home and Care
5	elk	Elektronik

11

Laporkan hasil Praktikum-5 ini dan *commit* perubahan pada *git*

Praktikum Ke-6



Langkah	Jawaban/Deskripsi
1	<p>Kita buat file model untuk tabel m_user dengan mengetikkan perintah php artisan make:model UserModel</p> <pre>PS C:\laragon\www\PWL_POS> php artisan make:model UserModel Warning: Module "gd" is already loaded in Unknown on line 0 INFO Model [C:\laragon\www\PWL_POS\app\Models\UserModel.php] created successfully.</pre>
2	<p>Setelah berhasil generate model, terdapat 2 file pada folder model yaitu file User.php bawaan dari laravel dan file UserModel.php yang telah kita buat. Kali ini kita akan menggunakan file UserModel.php</p>
3	<p>Kita buka file UserModel.php dan modifikasi seperti berikut</p> <pre>app > Models > UserModel.php > 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 0 references 0 implementations 9 class UserModel extends Model 10 11 use HasFactory; 12 13 0 references 14 protected \$table = 'm_user'; //Mendefinisikan nama tabel yang digunakan oleh model ini 15 16 0 references 17 protected \$primaryKey = 'user_id'; //Mendefinisikan primary key dari table yang digunakan</pre>
4	<p>Kita modifikasi route web.php untuk mencoba routing ke controller UserController</p>



	<pre>routes > web.php > ... 1 <?php 2 3 use App\Http\Controllers\KategoriController; 4 use App\Http\Controllers\LevelController; 5 use App\Http\Controllers\UserController; 6 use Illuminate\Support\Facades\Route; 7 8 Route::get(uri: '/', action: function (): Factory View{ 9 return view(view: 'welcome'); 10 }); 11 12 Route::get(uri: '/level', action: [LevelController::class, 'index']); 13 Route::get(uri: '/kategori', action: [KategoriController::class, 'index']); 14 Route::get(uri: '/user', action: [UserController::class, 'index']); 15 16 ?></pre>
5	<p>Sekarang, kita buat file controller UserController dan memodifikasinya seperti berikut</p> <pre>app > Http > Controllers > UserController.php > PHP Intelephense > UserController 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use App\Models\UserModel; 6 use Illuminate\Http\Request; 7 8 class UserController extends Controller{ 9 public function index(): Factory View 10 { 11 //coba akses model UserModel 12 \$user = UserModel::all(); //ambil semua data dari tabel m_user 13 return view(view: 'user', data: ['id' => \$user]); 14 } 15 } 16 ?></pre>
6	<p>Kemudian kita buat view user.blade.php</p>



```
resources > views > user.blade.php > html > body
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Data User</title>
5  </head>
6  <body>
7      <h1>Data User</h1>
8      <table border="1" cellpadding="2" cellspacing="0">
9          <tr>
10             <th>ID</th>
11             <th>Username</th>
12             <th>Nama</th>
13             <th>ID Level pengguna</th>
14          </tr>
15          @foreach ($data as $d)
16              <tr>
17                  <td>{{ $d->user_id }}</td>
18                  <td>{{ $d->username }}</td>
19                  <td>{{ $d->nama }}</td>
20                  <td>{{ $d->level_id }}</td>
21              </tr>
22          @endforeach
23      </table>
24  </body>
25 </html>
```

7 Jalankan di browser, catat dan laporkan apa yang terjadi

← → ↻ ⓘ localhost/PWL_POS/public/user 🔍 ☆ 📁 🖱 ⋮

Data User

ID	Username	Nama	ID Level pengguna
1	admin	Administrator	1
2	manager	Manager	2
3	staff	Staff/Kasir	3

8 Setelah itu, kita modifikasi lagi file UserController

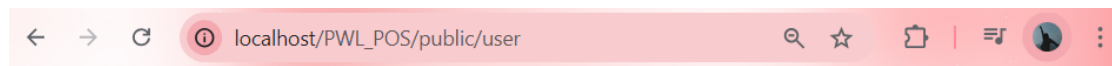


	<pre>app > Http > Controllers > UserController.php > ... 1 <?php 2 3 namespace App\Http\Controllers; 4 5 use App\Models\UserModel; 6 use Illuminate\Http\Request; 7 use Illuminate\Support\Facades\Hash; 8 9 class UserController extends Controller{ 10 public function index(): Factory View 11 { 12 //tambah data user dengan Eloquent Model 13 \$data = [14 'username' => 'customer-1', 15 'nama' => 'Pelanggan', 16 'password' => Hash::make('12345'), 17 'level_id' => 4 18]; 19 UserModel::insert(values: \$data); //tambahkan data ke tabel m_user 20 21 //coba akses model UserModel 22 \$user = UserModel::all(); //ambil semua data dari tabel m_user 23 return view('user', data: ['data' => \$user]); 24 } 25 } 26</pre>																				
9	<p>Jalankan di browser, amati dan laporkan apa yang terjadi</p> <div><div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div></div><div>localhost/PWL_POS/public/user</div><div><div></div><div></div><div></div><div></div><div></div><div></div></div></div> <h3>Data User</h3> <table><tr><th>ID</th><th>Username</th><th>Nama</th><th>ID Level pengguna</th></tr><tr><td>1</td><td>admin</td><td>Administrator</td><td>1</td></tr><tr><td>2</td><td>manager</td><td>Manager</td><td>2</td></tr><tr><td>3</td><td>staff</td><td>Staff/Kasir</td><td>3</td></tr><tr><td>14</td><td>customer-1</td><td>Pelanggan</td><td>4</td></tr></table>	ID	Username	Nama	ID Level pengguna	1	admin	Administrator	1	2	manager	Manager	2	3	staff	Staff/Kasir	3	14	customer-1	Pelanggan	4
ID	Username	Nama	ID Level pengguna																		
1	admin	Administrator	1																		
2	manager	Manager	2																		
3	staff	Staff/Kasir	3																		
14	customer-1	Pelanggan	4																		
10	Kita modifikasi lagi file UserController menjadi seperti berikut																				



```
app > Http > Controllers > UserController.php > PHP > UserController > index()
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\Models\UserModel;
6  use Illuminate\Http\Request;
7  use Illuminate\Support\Facades\Hash;
8
9  class UserController extends Controller{
10     public function index(): Factory|View
11     {
12         //tambah data user dengan Eloquent Model
13         // $data = [
14         //     'username' => 'customer-1',
15         //     'nama' => 'Pelanggan',
16         //     'password' => Hash::make('12345'),
17         //     'level_id' => 4
18         // ];
19         // UserModel::insert($data); //tambahkan data ke tabel m_user
20
21         $data = [
22             'nama'=> 'Pelanggan Pertama',
23         ];
24         UserModel::where(column: 'username',operator: 'customer-1')->update(attributes: $data);//update data user
25
26         $user = UserModel::all(); //mengambil semua data dari table m_user
27         return view(view: 'user',data: ['data'=>$user]);
28     }
29 }
30
```

- 11 Jalankan di browser, amati dan laporkan apa yang terjadi



Data User

ID	Username	Nama	ID Level pengguna
1	admin	Administrator	1
2	manager	Manager	2
3	staff	Staff/Kasir	3
14	customer-1	Pelanggan Pertama	4

- 12 Jika sudah, laporkan hasil Praktikum-6 ini dan *commit* perubahan pada *git*



Soal Praktikum

Jawablah pertanyaan berikut sesuai pemahaman materi di atas

Langkah	Jawaban/Deskripsi
1	<p>Pada Praktikum 1 - Tahap 5, apakah fungsi dari APP_KEY pada file setting .env Laravel?</p> <p>Berdasarkan pemahaman saya, APP_KEY pada file setting .env di Laravel adalah kunci rahasia yang digunakan untuk mengamankan data dengan cara mengenkripsi dan mendekripsi informasi penting, seperti sesi pengguna dan cookie. Ini memastikan data aman dan hanya dapat diakses oleh aplikasi yang benar-benar berwenang.</p>
2	<p>Pada Praktikum 1, bagaimana kita men-<i>generate</i> nilai untuk APP_KEY?</p> <p>Berdasarkan pemahaman saya, dari penjelasan bapak Dimas kemarin saat jam kuliah, perintah yang digunakan untuk men-<i>generate</i> nilai untuk APP_KEY adalah php artisan key:generate</p>
3	<p>Pada Praktikum 2.1 - Tahap 1, secara <i>default</i> Laravel memiliki berapa file migrasi? dan untuk apa saja file migrasi tersebut?</p> <p>Berdasarkan ingatan saya, secara default Laravel memiliki beberapa file migrasi diantaranya file-file berikut yang berfungsi untuk membantu dalam mengatur struktur database dasar yang diperlukan untuk aplikasi Laravel:</p> <ul style="list-style-type: none">- create_users_table.php: untuk membuat tabel `users` dan menyimpan data pengguna.- create_password_resets_table.php: untuk membuat tabel `password_resets` menyimpan data reset password.- create_failed_jobs_table.php: untuk membuat tabel `failed_jobs` untuk menyimpan informasi pekerjaan yang gagal.
4	<p>Secara <i>default</i>, file migrasi terdapat kode \$table->timestamps();, apa tujuan/output dari fungsi tersebut?</p> <p>Berdasarkan pemahaman saya dari hasil praktikum, fungsi dari kode \$table->timestamps();, dalam migrasi Laravel menambahkan dua kolom pada tabel database yaitu:</p>



	<ul style="list-style-type: none">- <code>created_at</code>: untuk mencatat waktu saat baris data dibuat.- <code>updated_at</code>: Mencatat waktu saat baris data terakhir diperbarui. <p>Jadi, kedua kolom tersebut membantu melacak kapan data ditambahkan atau diubah.</p>
5	<p>Pada File Migrasi, terdapat fungsi <code>\$table->id()</code> ; Tipe data apa yang dihasilkan dari fungsi tersebut?</p> <p>Berdasarkan pemahaman saya, fungsi dari kode <code>\$table->id()</code> ; pada file migrasi Laravel menghasilkan kolom dengan tipe data <code>BIGINT</code> yang auto-increment dan menjadi <code>primary key</code> pada tabel. Ini secara otomatis mengatur kolom tersebut sebagai identifikasi unik untuk setiap baris data.</p>
6	<p>Apa bedanya hasil migrasi pada table <code>m_level</code>, antara menggunakan <code>\$table->id()</code> ; dengan menggunakan <code>\$table->id('level_id')</code> ; ?</p> <p>Berdasarkan pemahaman saya dari hasil praktikum, fungsi dari kode <code>\$table->id()</code> ; pada file migrasi Laravel menghasilkan kolom dengan tipe data <code>BIGINT</code> yang auto-increment dan menjadi <code>primary key</code> pada tabel. Ini secara otomatis mengatur kolom tersebut sebagai identifikasi unik untuk setiap baris data. Sedangkan, perbedaannya adalah <code>table->id()</code> ; untuk membuat kolom bernama <code>`id`</code> dengan tipe data <code>BIGINT</code> yang auto-increment dan menjadi <code>primary key</code> tabel secara otomatis. Sementara, <code>\$table->id('level_id')</code> ; untuk membuat kolom bernama <code>'level_id'</code> dengan tipe data <code>`BIGINT`</code> yang auto-increment dan menjadi <code>primary key</code> tabel, tetapi dengan nama kolom yang spesifik yaitu <code>level_id</code>. Jadi, perbedaannya terletak pada nama kolom yang dihasilkan <code>`id`</code> vs. <code>`level_id`</code></p>
7	<p>Pada migration, Fungsi <code>->unique()</code> digunakan untuk apa?</p> <p>Berdasarkan pemahaman saya, fungsi dari kode <code>unique()</code> digunakan untuk menetapkan kolom pada tabel agar hanya menerima nilai yang unik. Artinya, tidak boleh ada dua baris atau lebih yang memiliki nilai yang sama di kolom tersebut.</p>
8	<p>Pada Praktikum 2.2 - Tahap 2, kenapa kolom <code>level_id</code> pada tabel <code>m_user</code> menggunakan <code>\$tabel->unsignedBigInteger('level_id')</code>, sedangkan</p>



	<p>kolom <code>level_id</code> pada tabel <code>m_level</code> menggunakan <code>\$tabel->id('level_id')</code> ?</p> <p>Berdasarkan pemahaman saya dari hasil praktikum, kolom <code>level_id</code> di <code>m_level</code> didefinisikan sebagai <code>BIGINT</code> dengan auto-increment untuk primary key. Sementara, kolom <code>level_id</code> di <code>m_user</code> didefinisikan sebagai <code>unsignedBigInteger</code> untuk menyimpan referensi ke primary key <code>level_id</code> di <code>m_level</code> dan diindeks untuk mempercepat pencarian. Jadi, perbedaan ini memastikan bahwa <code>level_id</code> di <code>m_user</code> cocok dengan tipe data dan atribut dari <code>level_id</code> di <code>m_level</code>, mendukung integritas referensial dalam database</p>
9	<p>Pada Praktikum 3 - Tahap 6, apa tujuan dari Class Hash? dan apa maksud dari kode program <code>Hash::make('1234')</code> ; ?</p> <p>Berdasarkan pemahaman saya dari hasil praktikum, tujuan dari Class <code>`Hash`</code> di Laravel digunakan untuk mengenkripsi password. <code>`Hash::make('12345')`</code> menghasilkan versi terenkripsi dari password <code>`12345`</code>, yang disimpan di database untuk keamanan.</p>
10	<p>Pada Praktikum 4 - Tahap 3/5/7, pada <i>query builder</i> terdapat tanda tanya (?), apa kegunaan dari tanda tanya (?) tersebut?</p> <p>Berdasarkan pemahaman saya, dalam <i>query builder</i> Laravel tanda tanya (?) adalah placeholder untuk parameter yang akan diisi saat eksekusi query. Fungsinya adalah untuk menggantikan nilai-nilai yang akan diinsert, update, atau digunakan dalam kondisi query.</p>
11	<p>Pada Praktikum 6 - Tahap 3, apa tujuan penulisan kode <code>protected \$table = 'm_user'</code>; dan <code>protected \$primaryKey = 'user_id'</code>; ?</p> <p>Berdasarkan pemahaman saya, tujuan penulisan kode <code>protected \$table = 'm_user'</code>; untuk menetapkan nama tabel di database yang digunakan oleh model ini. Sementara, tujuan penulisan kode <code>protected \$primaryKey = 'user_id'</code>; untuk menetapkan nama kolom yang menjadi primary key di tabel tersebut.</p>
12	<p>Menurut kalian, lebih mudah menggunakan mana dalam melakukan operasi CRUD ke database (<i>DB Façade / Query Builder / Eloquent ORM</i>) ? jelaskan</p>



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<https://jti.polinema.ac.id>

	Berdasarkan pemahaman saya, setiap operasi pastinya memiliki kelebihan dan kekurangan masing-masing. Tetapi Menurut saya yang lebih mudah yaitu Eloquent ORM karena lebih mudah dan cepat untuk pengembangan, sementara DB Façade dan Query Builder menawarkan lebih banyak kontrol dan fleksibilitas.
--	--