



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

Nama : Elvaretta Salsabilla

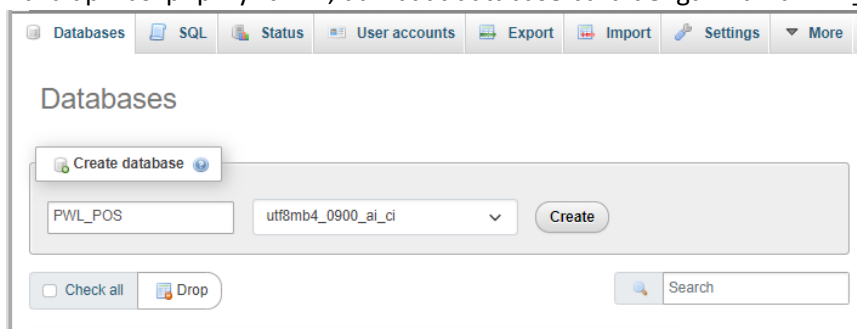
NIM : 2241720116

No/Kelas: 14 / TI-2H

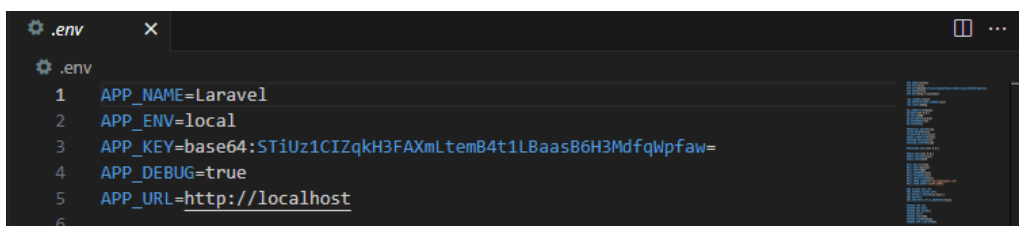
A. PENGATURAN DATABASE

Praktikum 1 - Pengaturan database

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. Edit file .env dan sesuaikan dengan database yang telah dibuat

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=PWL_POS
DB_USERNAME=root
DB_PASSWORD=
```

B. MIGRATION



Jurusan Teknologi Informasi Politeknik Negeri Malang

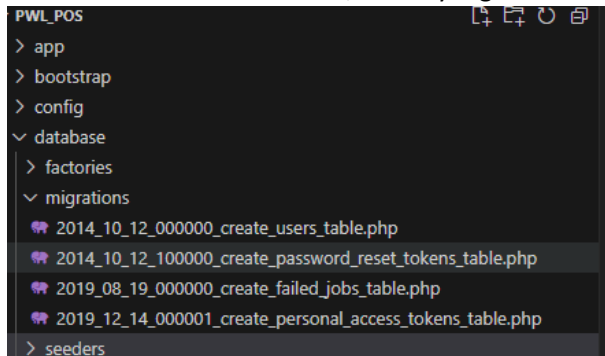
Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

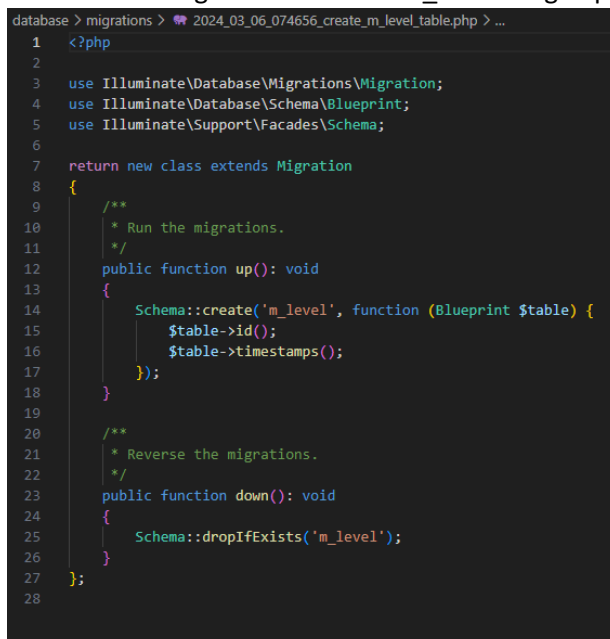
Maret 2024

Praktikum 2.1 - Pembuatan file migrasi tanpa relasi

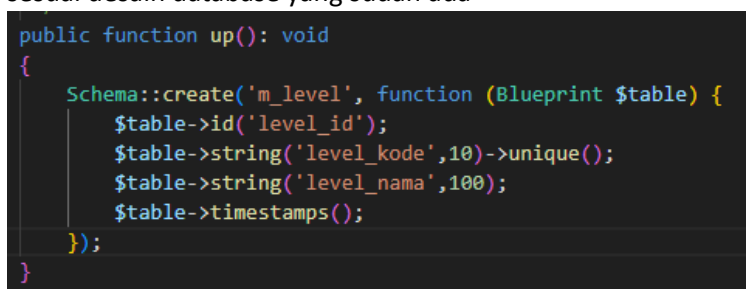
1. Buka terminal VSCode kalian, untuk yang di kotak merah adalah default dari laravel



2. Kita abaikan dulu yang di kotak merah (jangan di hapus)
3. Kita buat file migrasi untuk table m_level dengan perintah



4. Kita perhatikan bagian yang di kotak merah, bagian tersebut yang akan kita modifikasi sesuai desain database yang sudah ada





Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

- Simpan kode pada tahapan 4 tersebut, kemudian jalankan perintah ini pada terminal VSCode untuk melakukan migrasi

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

INFO Preparing database.

Creating migration table ..... 220ms DONE

INFO Running migrations.

2014_10_12_000000_create_users_table ..... 799ms DONE
2014_10_12_100000_create_password_reset_tokens_table ..... 186ms DONE
2019_08_19_000000_create_failed_jobs_table ..... 464ms DONE
2019_12_14_000001_create_personal_access_tokens_table ..... 377ms DONE
2024_03_06_074656_create_m_level_table ..... 270ms DONE
```

- Kemudian kita cek di phpMyAdmin apakah table sudah ter-generate atau belum

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> failed_jobs	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> migrations	★ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> m_level	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> password_reset_tokens	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> personal_access_tokens	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> users	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
6 tables	Sum	5	InnoDB	utf8mb4_0900_ai_ci	96.0 KiB	0 B

- Ok, table sudah dibuat di database
- Buat table database dengan migration untuk table m_kategori yang sama-sama tidak memiliki foreign key

```
public function up(): void
{
    Schema::create('m_kategori', function (Blueprint $table) {
        $table->id("kategori_id");
        $table->string("kategori_kode", 10) ->unique();
        $table->string("kategori_nama",100);
        $table->timestamps();
    });
}
```

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> failed_jobs	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> migrations	★ Browse Structure Search Insert Empty Drop	6	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> m_kategori	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> m_level	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
<input type="checkbox"/> password_reset_tokens	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> personal_access_tokens	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> users	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
7 tables	Sum	6	InnoDB	utf8mb4_0900_ai_ci	128.0 KiB	0 B

- Laporkan hasil Praktikum-2.1 ini dan commit perubahan pada git.



Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Praktikum 2.2 - Pembuatan file migrasi dengan relasi

1. Buka terminal VSCode kalian, dan buat file migrasi untuk table m_user

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

INFO Migration [C:\laragon\www\PWL_POS\database\migrations\2024_03_06_084529_create_m_user_table.php] created successfully.

2. Buka file migrasi untuk table m_user, dan modifikasi seperti berikut

```
*/
public function up(): void
{
    Schema::create('m_user', function (Blueprint $table) {
        $table->id('user_id');
        $table->unsignedBigInteger('level_id')->index(); //indexing untuk foreignkey
        $table->string('username', 20)->unique(); // unique untuk memastikan tidak ada
        username yang sama
        $table->string('nama',100);
        $table->string('password');
        $table->timestamps();

        //Mendefinisikan Foreign key pada kolom level_id mengacu pada kolom level_id di
        tabel m_level
        $table->foreign('level_id')->references('level_id')->on('m_level');
    });
}
```

3. Simpan kode program Langkah 2, dan jalankan perintah php artisan migrate. Amati apa yang terjadi pada database.

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> failed_jobs	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> migrations	Browse Structure Search Insert Empty Drop	7	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> m_kategori	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> m_level	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
<input type="checkbox"/> m_user	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
<input type="checkbox"/> password_reset_tokens	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> personal_access_tokens	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> users	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
8 tables	Sum	7	InnoDB	utf8mb4_0900_ai_ci	160.0 KiB	0 B

4. Buat table database dengan migration untuk table-table yang memiliki foreign key
 - a. m_barang



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

```
return new Class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('m_barang', function (Blueprint $table) {
            $table->id('barang_id');
            $table->unsignedBigInteger('kategori_id')->index();
            $table->string('barang_kode', 10)->unique();
            $table->string('barang_nama', 100);
            $table->integer('harga_beli');
            $table->integer('harga_jual');
            $table->timestamps();

            $table->foreign('kategori_id')->references('kategori_id')->on('m_kategori');
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('m_barang');
    }
};
```

b. t_penjualan

```
return new class extends Migration
{
    /**
     * Run the migrations.*/
    public function up(): void
    {
        Schema::create('t_penjualan', function (Blueprint $table) {
            $table->id('penjualan_id');
            $table->unsignedBigInteger('user_id')->index(); //indexing untuk foreignkey
            $table->string('pembeli', 50)->unique(); // unique untuk memastikan tidak ada username yang sama
            $table->string('penjualan_kode', 20);
            $table->dateTime('penjualan_tanggal');
            $table->timestamps();

            //Mendefinisikan Foreign key
            $table->foreign('user_id')->references('user_id')->on('m_user');
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('t_penjualan');
    }
};
```

c. t_stok



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

```
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('t_stok', function (Blueprint $table) {
            $table->id('stok_id');
            $table->unsignedBigInteger('barang_id')->index(); //indexing untuk foreignkey
            $table->unsignedBigInteger('user_id')->index(); //indexing untuk foreignkey
            $table->dateTime('stok_tanggal');
            $table->integer('stok_jumlah');
            $table->timestamps();

            //Mendefinisikan Foreign key
            $table->foreign('barang_id')->references('barang_id')->on('m_barang');
            $table->foreign('user_id')->references('user_id')->on('m_user');
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('t_stok');
    }
};
```

d. t_penjualan_detail

```
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('t_penjualan_detail', function (Blueprint $table) {
            $table->id('detail_id');
            $table->unsignedBigInteger('penjualan_id')->index(); //indexing untuk foreignkey
            $table->unsignedBigInteger('barang_id')->index(); //indexing untuk foreignkey
            $table->integer('harga');
            $table->integer('jumlah');
            $table->timestamps();

            //Mendefinisikan Foreign key
            $table->foreign('barang_id')->references('barang_id')->on('m_barang');
            $table->foreign('penjualan_id')->references('penjualan_id')->on('t_penjualan');
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('t_penjualan_detail');
    }
};
```

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

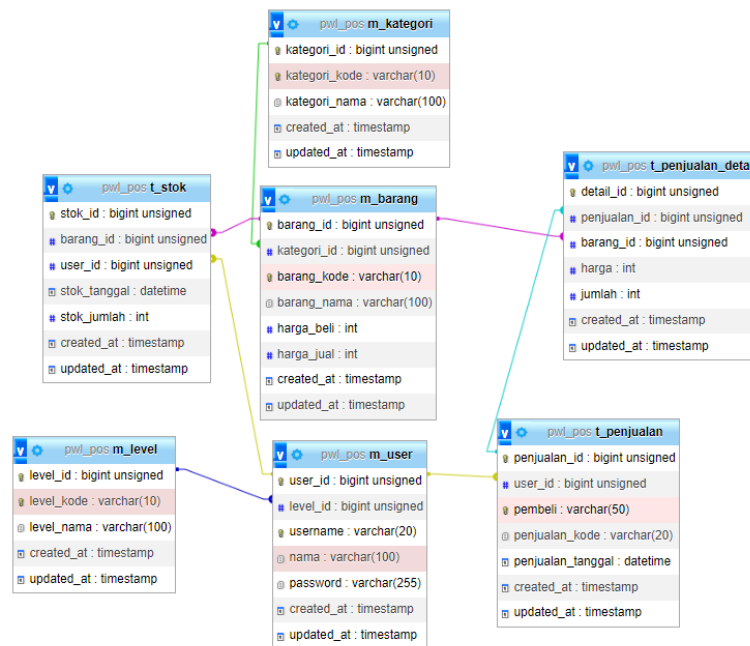


Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024



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

C. SEEDER

Praktikum 3 – Membuat file seeder

1. Kita akan membuat file seeder untuk table m_level dengan mengetikkan perintah

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

```
1 <?php
2
3 namespace Database\Seeders;
4
5 use Illuminate\Database\Console\Seeds\WithoutModelEvents;
6 use Illuminate\Database\Seeder;
7
8 0 references | 0 implementations
9 class LevelSeeder extends Seeder
10 {
11     /**
12      * Run the database seeds.
13      */
14     0 references | 0 overrides
15     public function run(): void
16     {
17         //
18     }
```



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

2. Selanjutnya, untuk memasukkan data awal, kita modifikasi file tersebut di dalam function run()

```
<?php

namespace Database\Seeders;

use Illuminate\Database\Console\Seeds\WithoutModelEvents;
use Illuminate\Database\Seeder;
use Illuminate\Support\Facades\DB;

0 references | 0 implementations
class LevelSeeder extends Seeder
{
    /**
     * Run the database seeds.
     */
    0 references | 0 overrides
    public function run(): void
    {
        $data = [
            ['level_id' => 1, 'level_kode' => 'ADM', 'level_nama' => 'Administrator'],
            ['level_id' => 2, 'level_kode' => 'MNG', 'level_nama' => 'Manager'],
            ['level_id' => 3, 'level_kode' => 'STF', 'level_nama' => 'Staff/Kasir']
        ];
        DB::table('m_level')->insert($data);
    }
}
```

3. Selanjutnya, kita jalankan file seeder untuk table m_level pada terminal

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

INFO Seeding database.
```

4. Ketika seeder berhasil dijalankan maka akan tampil data pada table m_level

		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. Sekarang kita buat file seeder untuk table m_user yang me-refer ke table m_level

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

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

6. Modifikasi file class UserSeeder seperti berikut:



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

```
class UserSeeder extends Seeder
{
    /**
     * Run the database seeds.
     */
    // 0 references | 0 overrides
    public function run(): void
    {
        $data = [
            [
                'user_id' => 1,
                'level_id' => 1,
                'username' => 'admin',
                'nama' => 'Administrator',
                'password' => Hash::make('12345'), // class untuk mengkripsikan/hash password
            ],
            [
                'user_id' => 2,
                'level_id' => 2,
                'username' => 'manager',
                'nama' => 'Manager',
                'password' => Hash::make('12345'),
            ],
            [
                'user_id' => 3,
                'level_id' => 3,
                'username' => 'staff',
                'nama' => 'Staff/Kasir',
                'password' => Hash::make('12345'),
            ],
        ];
        DB::table('m_user')->insert($data);
    }
}
```

7. Jalankan perintah untuk mengeksekusi class UserSeeder

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

INFO Seeding database.

8. Perhatikan hasil seeder pada table m_user

	user_id	level_id	username	nama	password	created_at
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	1	admin	Administrator	\$2y\$12\$FdxpmYAqDLiFapglPe49rOzpjAJEF9ID5OJVeZmAz0J...	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	2	manager	Manager	\$2y\$12\$Sn4OYZarht9BWZ4Q1ht2qEuhi0DfM8QsWAYCTqf7bnkQ...	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	3	staff	Staff/Kasir	\$2y\$12\$uzaB8lIXrcRLAOGRIpW4yumkmKPFMNQKE7YWBtcrCQ1...	NULL

9. Ok, data seeder berhasil di masukkan ke database.
10. Sekarang coba kalian masukkan data seeder untuk table yang lain, dengan ketentuan seperti berikut:

No	Nama Tabel	Jumlah Data	Keterangan
1	m_kategori	5	5 kategori barang
2	m_barang	10	10 barang yang berbeda
3	t_stok	10	Stok untuk 10 barang
4	t_penjualan	10	10 transaksi penjualan
5	t_penjualan_detail	30	3 barang untuk setiap transaksi penjualan

- 1) Tabel m_kategori



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

```
public function run(): void
{
    $data = [
        [
            'kategori_id' => 1,
            'kategori_kode' => 'FSH',
            'kategori_nama' => 'Fashion',
        ],
        [
            'kategori_id' => 2,
            'kategori_kode' => 'MKU',
            'kategori_nama' => 'Make Up',
        ],
        [
            'kategori_id' => 3,
            'kategori_kode' => 'ELK',
            'kategori_nama' => 'Elektronik',
        ],
        [
            'kategori_id' => 4,
            'kategori_kode' => 'BRT',
            'kategori_nama' => 'Barang Rumah Tangga',
        ],
        [
            'kategori_id' => 5,
            'kategori_kode' => 'MKN',
            'kategori_nama' => 'Makanan',
        ],
    ];
    DB::table('m_kategori')->insert($data);
}
```

		kategori_id	kategori_kode	kategori_nama	created_at	updated_at
<input type="checkbox"/>	Edit Copy Delete	1	FSH	Fashion	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	2	MKU	Make Up	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	3	ELK	Elektronik	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	4	BRT	Barang Rumah Tangga	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	5	MKN	Makanan	NULL	NULL

2) Tabel m_barang



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

```
class BarangSeeder extends Seeder
{
    /**
     * Run the database seeds.
     */
    // 0 references | 0 overrides
    public function run(): void
    {
        $data = [
            [
                'barang_id' => 1,
                'kategori_id' => 1,
                'barang_kode' => 'KMJ',
                'barang_nama' => 'Kemeja',
                'harga_beli' => 20000,
                'harga_jual' => 25000,
            ],
            [
                'barang_id' => 2,
                'kategori_id' => 2,
                'barang_kode' => 'MSK',
                'barang_nama' => 'Maskara',
                'harga_beli' => 30000,
                'harga_jual' => 35000,
            ],
            [
                'barang_id' => 3,
                'kategori_id' => 3,
                'barang_kode' => 'KYB',
                'barang_nama' => 'Keyboard',
                'harga_beli' => 110000,
                'harga_jual' => 115000,
            ],
        ],
    }
}
```

	barang_id	kategori_id	barang_kode	barang_nama	harga_beli	harga_jual	created_at	updated_at
<input type="checkbox"/> Edit Copy Delete	1	1 KMJ	Kemeja	20000	25000	NULL	NULL	
<input type="checkbox"/> Edit Copy Delete	2	2 MSK	Maskara	30000	35000	NULL	NULL	
<input type="checkbox"/> Edit Copy Delete	3	3 KYB	Keyboard	110000	115000	NULL	NULL	
<input type="checkbox"/> Edit Copy Delete	4	4 TFL	Teflon	50000	55000	NULL	NULL	
<input type="checkbox"/> Edit Copy Delete	5	5 KBA	Kebab Ayam	20000	25000	NULL	NULL	
<input type="checkbox"/> Edit Copy Delete	6	1 JKT	Jaket	40000	45000	NULL	NULL	
<input type="checkbox"/> Edit Copy Delete	7	2 LPS	Lipstick	15000	20000	NULL	NULL	
<input type="checkbox"/> Edit Copy Delete	8	3 SEP	Speaker	150000	155000	NULL	NULL	
<input type="checkbox"/> Edit Copy Delete	9	4 SPI	Sapu Ijuk	10000	15000	NULL	NULL	
<input type="checkbox"/> Edit Copy Delete	10	5 KPE	Keripik Tempe	17000	22000	NULL	NULL	

3) Tabel t_stok



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

```
public function run(): void
{
    $data = [
        [
            'stok_id' => 1,
            'barang_id' => 1,
            'user_id' => 1,
            'stok_tanggal' => '2024-01-11 09:10:00',
            'stok_jumlah' => 100,
        ],
        [
            'stok_id' => 2,
            'barang_id' => 2,
            'user_id' => 1,
            'stok_tanggal' => '2024-01-12 09:15:00',
            'stok_jumlah' => 50,
        ],
        [
            'stok_id' => 3,
            'barang_id' => 3,
            'user_id' => 1,
            'stok_tanggal' => '2024-01-11 09:20:00',
            'stok_jumlah' => 60,
        ],
        [
            'stok_id' => 4,
            'barang_id' => 4,
            'user_id' => 1,
            'stok_tanggal' => '2024-01-13 10:09:00',
            'stok_jumlah' => 100,
        ],
    ],
```

	stok_id	barang_id	user_id	stok_tanggal	stok_jumlah	created_at	updated_at
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	1	1	2024-01-11 09:10:00	100	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	2	1	2024-01-12 09:15:00	50	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	3	1	2024-01-11 09:20:00	60	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	4	1	2024-01-13 10:09:00	100	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	5	1	2024-01-12 10:10:00	20	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	6	1	2024-01-11 11:30:00	200	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	7	7	1	2024-01-12 11:50:00	70	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	8	8	1	2024-01-12 12:00:00	90	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	9	9	1	2024-01-11 12:10:10	40	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	10	10	1	2024-01-10 13:00:00	80	NULL	NULL

4) Tabel t_penjualan



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

```
public function run(): void
{
    $data = [
        [
            'penjualan_id' => 1,
            'user_id' => 3,
            'Pembeli' => 'Nadia',
            'penjualan_kode' => 'P01',
            'penjualan_tanggal' => '2024-01-01 12:01:00',
        ],
        [
            'penjualan_id' => 2,
            'user_id' => 3,
            'pembeli' => 'Nofan',
            'penjualan_kode' => 'P02',
            'penjualan_tanggal' => '2024-01-02 12:05:00',
        ],
        [
            'penjualan_id' => 3,
            'user_id' => 3,
            'pembeli' => 'Demian',
            'penjualan_kode' => 'P03',
            'penjualan_tanggal' => '2024-01-02 13:04:00',
        ],
        [
            'penjualan_id' => 4,
            'user_id' => 3,
            'pembeli' => 'Aldi',
            'penjualan_kode' => 'P04',
            'penjualan_tanggal' => '2024-01-01 13:07:00',
        ],
    ],
```

	penjualan_id	user_id	pembeli	penjualan_kode	penjualan_tanggal	created_at	updated_at
<input type="checkbox"/> Edit Copy Delete	1	3	Nadia	P01	2024-01-01 12:01:00	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	2	3	Nofan	P02	2024-01-02 12:05:00	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	3	3	Demian	P03	2024-01-02 13:04:00	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	4	3	Aldi	P04	2024-01-01 13:07:00	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	5	3	Udin	P05	2024-01-03 14:00:00	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	6	3	Radit	P06	2024-01-01 14:15:00	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	7	3	Alifian	P07	2024-01-03 14:30:00	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	8	3	Dito	P08	2024-01-02 15:00:00	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	9	3	Zagar	P09	2024-01-01 15:40:00	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	10	3	Dika	P10	2024-01-02 15:50:00	NULL	NULL

5) Tabel t_penjualan_detail



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

```
0 references to overrides
public function run(): void
{
    $data = [
        //transaksi 1
        [
            'detail_id' => 1,
            'penjualan_id' => 1,
            'barang_id' => 1,
            'harga' => 25000,
            'jumlah' => 3,
        ],
        [
            'detail_id' => 2,
            'penjualan_id' => 1,
            'barang_id' => 2,
            'harga' => 35000,
            'jumlah' => 2,
        ],
        [
            'detail_id' => 3,
            'penjualan_id' => 1,
            'barang_id' => 5,
            'harga' => 25000,
            'jumlah' => 5,
        ],
    ],
```

	detail_id	penjualan_id	barang_id	harga	jumlah	created_at	updated_at
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	2	3	115000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	2	1	25000	4	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	2	8	155000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	7	3	6	45000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	8	3	10	22000	2	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	9	3	5	35000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	10	4	1	25000	3	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	11	4	6	45000	2	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	12	4	9	15000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	13	5	1	25000	6	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	14	5	7	20000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	15	5	4	55000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	16	6	2	35000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	17	6	8	150000	2	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	18	6	9	15000	10	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	19	7	6	45000	2	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	20	7	3	115000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	21	7	8	150000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	22	8	1	25000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	23	8	7	20000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	24	8	10	22000	4	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	25	9	4	55000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	26	9	3	115000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	27	9	9	15000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	28	10	5	25000	1	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	29	10	4	55000	2	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	30	10	7	20000	1	NULL	NULL

11. . Jika sudah, laporkan hasil Praktikum-3 ini dan commit perubahan pada git.

D. DB FACADE

Praktikum 4 – Implementasi DB Façade



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

1. Kita buat controller dahulu untuk mengelola data pada table m_level

```
PS C:\laragon\www\PWL_POS> php artisan make:controller LevelController

INFO Controller [C:\laragon\www\PWL_POS\app\Http\Controllers\LevelController.php] created successfully.
```

2. Kita modifikasi dulu untuk routing-nya, ada di PWL_POS/routes/web.php

```
Route::get('/', function () {
    return view('welcome');
});

Route::get('/level', [LevelController::class, 'index']);
```

3. Selanjutnya, kita modifikasi file LevelController untuk menambahkan 1 data ke table m_level

```
2 references | 0 implementations
class LevelController extends Controller
{
    1 reference | 0 overrides
    public function index(){
        DB::insert('insert into m_level(level_kode, level_nama, created_at) values (?, ?, ?)
        ', ['CUS', 'Pelanggan', now()]);
        return 'insert data baru berhasil';
    }
}
```

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

level_id	level_kode	level_nama	created_at	updated_at
1	ADM	Administrator	NULL	NULL
2	MNG	Manager	NULL	NULL
3	STF	Staff/Kasir	NULL	NULL
4	CUS	Pelanggan	2024-03-09 08:45:37	NULL

Yang terjadi pada tabel yaitu penambahan data cus sebagai pelanggan.

5. Selanjutnya, kita modifikasi lagi file LevelController untuk meng-update data di table m_level seperti berikut:



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

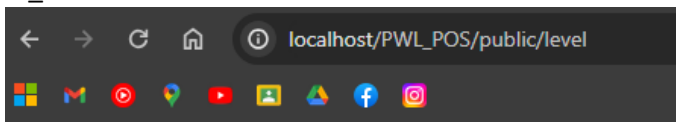
Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

```
class LevelController extends Controller
{
  1 reference | 0 overrides
  public function index(){
    // DB::insert('insert into m_level(level_kode, level_nama, created_at) values
    (?, ?, ?)' ,['CUS', 'Pelanggan', now()]);
    // return 'insert data baru berhasil';

    $row = DB::update('update m_level set level_nama = ? where level_kode = ?',
    ['Customer', 'CUS']);
    return 'Update data berhasil. Jumlah data yang diupdate: '.$row. ' baris';
  }
}
```

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.



Update data berhasil. Jumlah data yang diupdate: 1 baris

Perubahan data pada tabel:

	level_id	level_kode	level_nama	created_at	updated_at
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	ADM	Administrator	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	MNG	Manager	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	STF	Staff/Kasir	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	CUS	Customer	2024-03-09 08:45:37	NULL

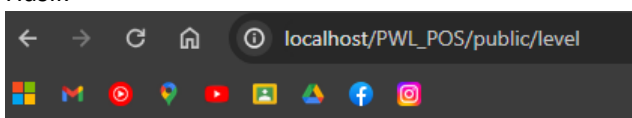
7. Kita coba modifikasi lagi file LevelController untuk melakukan proses hapus data.

```
class LevelController extends Controller
{
  1 reference | 0 overrides
  public function index(){
    // DB::insert('insert into m_level(level_kode, level_nama, created_at) values
    (?, ?, ?)' ,['CUS', 'Pelanggan', now()]);
    // return 'insert data baru berhasil';

    // $row = DB::update('update m_level set level_nama = ? where level_kode = ?',
    // ['Customer', 'CUS']);
    // return 'Update data berhasil. Jumlah data yang diupdate: '.$row. ' baris';

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

Hasil:



Delete data berhasil. Jumlah data yang dihapus: 1baris



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

	level_id	level_kode	level_nama	created_at	updated_at
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	ADM	Administrator	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	MNG	Manager	NULL	NULL
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> 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:

```
class LevelController extends Controller
{
    1 reference | 0 overrides
    public function index(){
        // DB::insert('insert into m_level(level_kode, level_nama,
        // created_at) values (?, ?, ?)', ['CUS', 'Pelanggan', now()]);
        // return 'insert data baru berhasil';

        // $row = DB::update('update m_level set level_nama = ? where
        // level_kode = ?', ['Customer', 'CUS']);
        // return 'Update data berhasil. Jumlah data yang diupdate: ' .
        // $row. ' baris';

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

        $data = DB::select('select * from m_level');
        return view('level', ['data' => $data]);
    }
}
```

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>Kode Nama</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 </html>
```

10. Silahkan dicoba pada browser dan amati apa yang terjadi!
Dalam web menampilkan data level dalam database

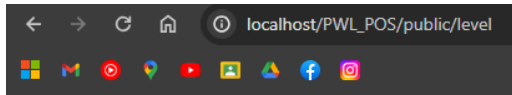


Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024



Data Level Pengguna

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

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

E. QUERY BUILDER

Praktikum 5 – Implementasi Query Builder

1. Kita buat controller dahulu untuk mengelola data pada table m_kategori

```
PS C:\laragon\www\PWL_POS> php artisan make:controller KategoriController
```

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

2. Kita modifikasi dulu untuk routing-nya, ada di PWL_POS/routes/web.php

```
Route::get('/level', [LevelController::class, 'index']);
Route::get('/kategori', [KategoriController::class, 'index']);
```

3. Selanjutnya, kita modifikasi file KategoriController untuk menambahkan 1 data ke table m_kategori

```
class KategoriController extends Controller
{
    1 reference | 0 overrides
    public function index(){
        $data = [
            'kategori_kode' => 'SNK',
            'kategori_nama' => 'Snack/Makanan Ringan',
            'created_at' => now()
        ];
        DB::table('m_kategori')->insert($data);
        return 'Insert data baru berhasil';
    }
}
```

4. 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

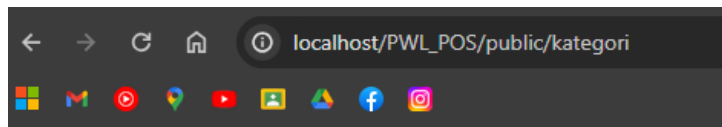


Jurusan Teknologi Informasi Politeknik Negeri Malang


Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024



Insert data baru berhasil

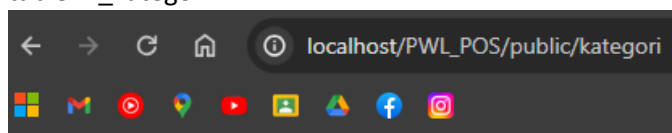
				kategori_id	kategori_kode	kategori_nama	created_at	updated_at
<input type="checkbox"/>	 Edit	 Copy	 Delete	1	FSH	Fashion	NULL	NULL
<input type="checkbox"/>	 Edit	 Copy	 Delete	2	MKU	Make Up	NULL	NULL
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	ELK	Elektronik	NULL	NULL
<input type="checkbox"/>	 Edit	 Copy	 Delete	4	BRT	Barang Rumah Tangga	NULL	NULL
<input type="checkbox"/>	 Edit	 Copy	 Delete	5	MKN	Makanan	NULL	NULL
<input type="checkbox"/>	 Edit	 Copy	 Delete	6	SNK	Snack/Makanan Ringan	2024-03-09 09:43:33	NULL

5. Selanjutnya, kita modifikasi lagi file KategoriController untuk meng-update data di table m_kategori seperti berikut:

```
class KategoriController extends Controller
{
    1 reference|0 overrides
    public function index(){
        // $data = [
        //     'kategori_kode' => 'SNK',
        //     'kategori_nama' => 'Snack/Makanan Ringan',
        //     'created_at' => now()
        // ];
        // DB::table('m_kategori')->insert($data);
        // return 'Insert data baru berhasil';

        $row = DB::table('m_kategori')->where('kategori_kode', 'SNK')
        ->update(['kategori_nama' => 'Camilan']);
        return 'Update data berhasil. Jumlah data yang diupdate: ' . $row->count() . ' baris';
    }
}
```

6. 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



Update data berhasil. Jumlah data yang diupdate: 1 baris

				kategori_id	kategori_kode	kategori_nama	created_at	updated_at
<input type="checkbox"/>	Edit	Copy	Delete	1	FSH	Fashion	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete	2	MKU	Make Up	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete	3	ELK	Elektronik	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete	4	BRT	Barang Rumah Tangga	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete	5	MKN	Makanan	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete	6	SNK	Camilan	2024-03-09 09:43:33	NULL

7. Kita coba modifikasi lagi file KategoriController untuk melakukan proses hapus data



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

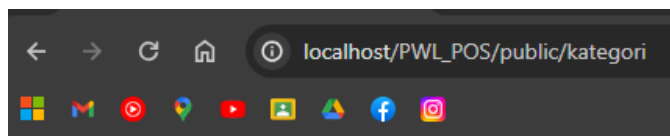
Maret 2024

```
class KategoriController extends Controller
{
    1 reference|0 overrides
    public function index(){
        // $data = [
        //     'kategori_kode' => 'SNK',
        //     'kategori_nama' => 'Snack/Makanan Ringan',
        //     'created_at' => now()
        // ];
        // DB::table('m_kategori')->insert($data);
        // return 'Insert data baru berhasil';

        // $row = DB::table('m_kategori')->where('kategori_kode', 'SNK')
        // ->update(['kategori_nama' => 'Camilan']);
        // return 'Update data berhasil. Jumlah data yang diupdate: ' .
        // $row->count();

        $row = DB::table('m_kategori')->where('kategori_kode', 'SNK')
        ->delete();
        return 'Delete data berhasil. Jumlah data yang dihapus: ' .
        $row->count();
    }
}
```

Hasil:



Delete data berhasil. Jumlah data yang dihapus 1 baris

 			kategori_id	kategori_kode	kategori_nama	created_at	updated_at	
<input type="checkbox"/>	 Edit	 Copy	 Delete	1	FSH	Fashion	NULL	NULL
<input type="checkbox"/>	 Edit	 Copy	 Delete	2	MKU	Make Up	NULL	NULL
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	ELK	Elektronik	NULL	NULL
<input type="checkbox"/>	 Edit	 Copy	 Delete	4	BRT	Barang Rumah Tangga	NULL	NULL
<input type="checkbox"/>	 Edit	 Copy	 Delete	5	MKN	Makanan	NULL	NULL

8. Method terakhir yang kita coba adalah untuk menampilkan data yang ada di table m_kategori. Kita modifikasi file KategoriController seperti berikut



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

```
class KategoriController extends Controller
{
    1 reference | 0 overrides
    public function index(){
        // $data = [
        //     'kategori_kode' => 'SNK',
        //     'kategori_nama' => 'Snack/Makanan Ringan',
        //     'created_at' => now()
        // ];
        // DB::table('m_kategori')->insert($data);
        // return 'Insert data baru berhasil';

        // $row = DB::table('m_kategori')->where('kategori_kode', 'SNK')
        //     ->update(['kategori_nama' => 'Camilan']);
        // return 'Update data berhasil. Jumlah data yang diupdate: ' .
        //     $row->count();

        // $row = DB::table('m_kategori')->where('kategori_kode', 'SNK')
        //     ->delete();
        // return 'Delete data berhasil. Jumlah data yang dihapus: ' .
        //     $row->count();

        $data = DB::table('m_kategori')->get();
        return view('kategori', ['data' => $data]);
    }
}
```

9. 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

```
resources > views > kategori.blade.php > html > body > table > tr > td
```

```
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>Kode 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 </html>
```

10. Silahkan dicoba pada browser dan amati apa yang terjadi.
Yang terjadi adalah halaman web menampilkan tabel kategori barang dari database

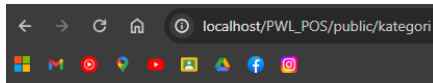


Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024



Data kategori Barang

ID	Kode Kategori	Kode Kategori
1	FSH	Fashion
2	MKU	Make Up
3	ELK	Elektronik
4	BRT	Barang Rumah Tangga
5	MKN	Makanan

11. Laporkan hasil Praktikum-5 ini dan commit perubahan pada git

F. ELOQUENT ORM

Praktikum 6 - Implementasi Eloquent ORM

1. Kita buat file model untuk tabel m_user dengan mengetikkan perintah

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

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

2. 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
3. Kita buka file UserModel.php dan modifikasi seperti berikut

```
class UserModel extends Model
{
    use HasFactory;

    0 references
    protected $table = 'm_user'; // mendefinisikan nama tabel yang
    digunakan oleh model ini
    0 references
    protected $primaryKey = 'user_id'; // mendefinisikan primary key dari
    tabel yang digunakan
}
```

4. Kita modifikasi route web.php untuk mencoba routing ke controller UserController

```
Route::get('/level', [LevelController::class, 'index']);
Route::get('/kategori', [KategoriController::class, 'index']);
Route::get('/user', [UserController::class, 'index']);
```

5. Sekarang, kita buat file controller UserController dan memodifikasinya seperti berikut



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

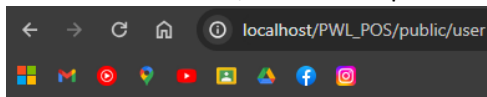
Maret 2024

```
class UserController extends Controller
{
    0 references | 0 overrides
    public function index(){
        //coba akses model UserModel
        $user = UserModel::all(); //ambil semua data dari tabel m_user
        return view('user', ['data'=>$user]);
    }
}
```

6. Kemudian kita buat view user.blade.php

```
<!DOCTYPE html>
<html>
    <head>
        <title>Data User</title>
    </head>
    <body>
        <h1>Data User </h1>
        <table border="1" cellpadding="2" cellspacing="0">
            <tr>
                <th>ID</th>
                <th>Username</th>
                <th>Nama</th>
                <th>ID Level Pengguna</th>
            </tr>
            @foreach ($data as $d )
                <tr>
                    <td>{{ $d->user_id }}</td>
                    <td>{{ $d->username }}</td>
                    <td>{{ $d->nama }}</td>
                    <td>{{ $d->level_id }}</td>
                </tr>
            @endforeach
        </table>
    </body>
</html>
```

7. Jalankan di browser, catat 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

Yang tampil di web adalah tabel data seluruh user

8. Setelah itu, kita modifikasi lagi file UserController



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

```
<?php

namespace App\Http\Controllers;

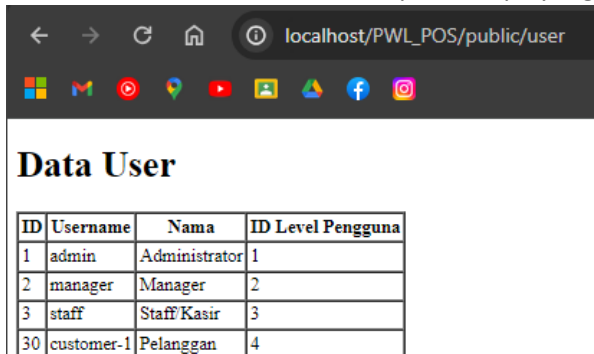
use App\Models\UserModel;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Hash;

2 references | 0 implementations
class UserController extends Controller
{
    1 reference | 0 overrides
    public function index(){

        // tambah data user dengan Eloquent Model
        $data = [
            'username' => 'customer-1',
            'nama' => 'Pelanggan',
            'password' => Hash::make('12345'),
            'level_id' => 4
        ];
        UserModel::insert($data); //tambahkan data ke tabel m_user

        //coba akses model UserModel
        $user = UserModel::all(); //ambil semua data dari tabel m_user
        return view('user', ['data'=>$user]);
    }
}
```

9. 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
30	customer-1	Pelanggan	4

Terjadi penambahan data pada tabel user

10. Kita modifikasi lagi file UserController menjadi seperti berikut



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

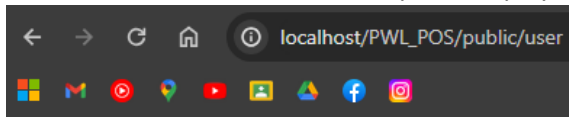
Maret 2024

```
class UserController extends Controller
{
    1 reference | 0 overrides
    public function index(){

        // // tambah data user dengan Eloquent Model
        // $data = [
        //     'level_id' => 4,
        //     'username' => 'customer-1',
        //     'nama' => 'Pelanggan',
        //     'password' => Hash::make('12345')
        // ];
        // UserModel::insert($data); //tambahkan data ke tabel m_user

        // tambah data dengan Eloquent model
        $data = [
            'nama' => 'Pelanggan Pertama',
        ];
        UserModel::where('username', 'customer-1')->update($data); //
        update data user
        //coba akses model UserModel
        $user = UserModel::all(); //ambil semua data dari tabel m_user
        return view('user', ['data'=>$user]);
    }
}
```

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
30	customer-1	Pelanggan Pertama	4

Terjadi perubahan pada nama username customer-1 dari pelanggan menjadi pelanggan pertama

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

G. PENUTUP

Jawablah pertanyaan berikut sesuai pemahaman materi di atas

- Pada Praktikum 1 - Tahap 5, apakah fungsi dari APP_KEY pada file setting .env Laravel?
Jawaban: APP_KEY adalah kunci enkripsi aplikasi, APP_KEY digunakan untuk mengamankan data yang disimpan di dalam cookie dan digunakan untuk enkripsi data dalam aplikasi.
- Pada Praktikum 1, bagaimana kita men-generate nilai untuk APP_KEY?
Jawaban: menggunakan perintah php artisan key:generate



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

3. Pada Praktikum 2.1 - Tahap 1, secara default Laravel memiliki berapa file migrasi? dan untuk apa saja file migrasi tersebut?

Jawaban: secara default laravel memiliki 5 file migrasi default

- 1) Tabel users : untuk menyimpan data pengguna dalam laravel
- 2) Tabel failed_jobs : untuk menyimpan pekerjaan yang gagal di eksekusi
- 3) Tabel migration : untuk melacak migrasi mana yang sudah dijalankan
- 4) Tabel password_reset_token : untuk menyimpan token reset password yang diberikan kepada pengguna saat meminta reset password
- 5) Tabel personal_access_tokens : untuk menyimpan token akses yang dibuat setelah setelah berhasil mengotentikasi

4. Secara default, file migrasi terdapat kode `$table->timestamps();`, apa tujuan/output dari fungsi tersebut?

Jawaban: tujuan dari fungsi tersebut adalah untuk menyederhanakan dan mengotomatiskan pelacakan waktu pembuatan dan pembaruan basis data.

5. Pada File Migrasi, terdapat fungsi `$table->id();` Tipe data apa yang dihasilkan dari fungsi tersebut?

Jawaban: fungsi `$table->id();` adalah untuk membuat kolom autoincrement dengan tipe data unsigned big integer, jadi tipe data yang dihasilkan dari fungsi tersebut adalah unsigned big integer

6. Apa bedanya hasil migrasi pada table `m_level`, antara menggunakan `$table->id();` dengan menggunakan `$table->id('level_id');` ?

Jawaban: Kedua query tersebut sama-sama menghasilkan kolom primary key yang bisa bertambah secara otomatis. Perbedaan keduanya hanya terletak pada penamaannya, jika menggunakan `$table->id();` maka secara otomatis akan membuat primary key dengan nama 'id', sedangkan jika menggunakan `$table->id('level_id');` maka primary key yang dibuat akan memiliki nama `level_id`.

7. Pada migration, Fungsi `->unique()` digunakan untuk apa?

Jawaban: Fungsi `->unique()` digunakan untuk menetapkan constraint unik pada kolom tertentu didalam tabel pada database. Ketika suatu kolom menggunakan konstrain unik berarti setiap nilai dalam kolom yang menggunakannya harus memiliki nama unik dan 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')` ?

Jawaban: kolom `level_id` pada tabel `m_user` menggunakan `$table->unsignedBigInteger('level_id')` dikarenakan kolom `level_id` digunakan sebagai foreign key yang merujuk ke primary key di tabel lain yaitu pada tabel `m_level` yang otomatis memiliki tipe data `unsignedBigInteger` karena menggunakan id untuk mendefinisikan primary key. Sedangkan kolom `level_id` pada tabel `m_level` menggunakan `$table->id('level_id')` karena tidak digunakan untuk foreign key, sehingga tidak menyesuaikan dengan kolom tabel lain.



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3 : MIGRATION, SEEDER, DB FAÇADE, QUERY BUILDER, dan ELOQUENT ORM

Mata Kuliah Praktikum Pemrograman Web Lanjut (PWL)

Maret 2024

9. Pada Praktikum 3 - Tahap 6, apa tujuan dari Class Hash? dan apa maksud dari kode program `Hash::make('1234');`?

Jawaban: class hash digunakan untuk menyediakan fasilitas hashing, hashing ini digunakan untuk mengenkripsi dan memverifikasi kata sandi. `Hash::make('1234')` digunakan untuk membuat hash dari string '12345', Hasilnya adalah string hash yang terenkripsi, yang kemudian dapat disimpan di dalam database sebagai representasi aman dari kata sandi.

10. Pada Praktikum 4 - Tahap 3/5/7, pada query builder terdapat tanda tanya (?), apa kegunaan dari tanda tanya (?) tersebut?

Jawaban: tanda tanya dalam query builder digunakan untuk parameter nilai yang akan dimasukkan ke dalam query saat query dijalankan.

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

Jawaban: tujuan penulisan kode `protected $table = 'm_user';` dan `protected $primaryKey = 'user_id';` adalah untuk menentukan tabel dan primary key yang terkait dalam model Eloquent. Dua kode tersebut digunakan untuk menyesuaikan model dengan tabel dan skema basis data tertentu.

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

Jawaban: Menurut saya lebih mudah menggunakan DB Façade karena querynya hampir sama dengan SQL jadi lebih mudah diingat