

Laporan Praktikum Web Lanjut

Jobsheet 3



NIM : 2241720107



Nama : Denny Malik Ibrahim

Kelas : TI-2H

**JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
TAHUN AJARAN 2023/2024**

Praktikum 1 - pengaturan database:

Databases

 Create database 

Create

```
.env x .env.example
.env
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:F8Xovb6XbtQr5WwlpKp2oFDVjP9kjOs1CBHKJJU5ZWM=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_DEPRECATIONS_CHANNEL=null
9 LOG_LEVEL=debug
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3307
14 DB_DATABASE=PWL_POS
15 DB_USERNAME=root
16 DB_PASSWORD=
17
```

Praktikum 2.1 - Pembuatan file migrasi tanpa relasi

```
PS C:\xampp\htdocs\PWL_2024> php artisan make:migration create_m_level_table --create=m_level
```

```
INFO Migration [C:\xampp\htdocs\PWL_2024\database\Migrations\2024_03_06_072501_create_m_level_table.php] created successfully.
```

```
PS C:\xampp\htdocs\PWL_2024>
```

```
Schema::create('m_level', function (Blueprint $table){
    $table->id('level_id');
    $table->string('level_kode', 10)->unique();
    $table->string('level_nama', 100);
    $table->timestamps();
});
```

```
PS C:\xampp\htdocs\PWL_2024> php artisan migrate

INFO Preparing database.

Creating migration table ..... 74ms DONE

INFO Running migrations.

2014_10_12_000000_create_users_table ..... 51ms DONE
2014_10_12_100000_create_password_reset_tokens_table ..... 5ms DONE
2019_08_19_000000_create_failed_jobs_table ..... 39ms DONE
2019_12_14_000001_create_personal_access_tokens_table ..... 23ms DONE
2024_03_06_072501_create_m_level_table ..... 17ms DONE
```

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> failed_jobs	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 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	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	48.0 KiB	-
<input type="checkbox"/> users	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
6 tables	Sum	5	InnoDB	utf8mb4_general_ci	176.0 KiB	0 B

```
INFO Migration [C:\xampp\htdocs\PWL_2024\database\migrations\2024_03_06_073706_create_m_kategori_table.php] created successfully.
```

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

```
PS C:\xampp\htdocs\PWL_2024> php artisan migrate

INFO Running migrations.

2024_03_06_073706_create_m_kategori_table ..... 24ms DONE
```

<input type="checkbox"/> m_kategori	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
<input type="checkbox"/> m_level	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-

Praktikum 2.2 - Pembuatan file migrasi dengan relasi

```
PS C:\xampp\htdocs\PWL_2024> php artisan make:migration create_m_user_table --create=m_user
```

```
INFO Migration [C:\xampp\htdocs\PWL_2024\database\Migrations\2024_03_06_074531_create_m_user_table.php] created successfully.
```

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

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

```
PS C:\xampp\htdocs\PWL_2024> php artisan migrate
```

```
INFO Running migrations.
```

```
2024_03_06_074531_create_m_user_table ..... 63ms DONE
```

```
PS C:\xampp\htdocs\PWL_2024> php artisan make:migration create_m_barang_table --create=m_barang
```

```
INFO Migration [C:\xampp\htdocs\PWL_2024\database\Migrations\2024_03_06_075320_create_m_barang_table.php] created successfully.
```

```
PS C:\xampp\htdocs\PWL_2024> php artisan make:migration create_t_penjualan_table --create=t_penjualan
```

```
INFO Migration [C:\xampp\htdocs\PWL_2024\database\Migrations\2024_03_06_075413_create_t_penjualan_table.php] created successfully.
```

```
PS C:\xampp\htdocs\PWL_2024> php artisan make:migration create_t_stok_table --create=t_stok
```

```
INFO Migration [C:\xampp\htdocs\PWL_2024\database\Migrations\2024_03_06_075425_create_t_stok_table.php] created successfully.
```

```
PS C:\xampp\htdocs\PWL_2024> php artisan make:migration create_t_penjualan_detail_table --create=t_penjualan_detail
```

```
INFO Migration [C:\xampp\htdocs\PWL_2024\database\Migrations\2024_03_06_075446_create_t_penjualan_detail_table.php] created successfully.
```

Barang

```
Schema::create('m_barang', function (Blueprint $table) {
    $table->id('barang_id');
    $table->unsignedBigInteger('kategori_id')->index(); // indexing for foreign key
    $table->string('barang_kode', 10)->unique(); // unique untuk memastikan tidak ada kode barang yang sama
    $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');
});
```

Penjualan

```
Schema::create('t_penjualan', function (Blueprint $table) {
    $table->id('penjualan_id');
    $table->unsignedBigInteger('user_id');
    $table->string('pembeli');
    $table->string('penjualan_kode', 20)->unique();
    $table->dateTime('penjualan_tanggal');
    $table->timestamps();

    $table->foreign('user_id')->references('user_id')->on('m_user');
});
```

Stok

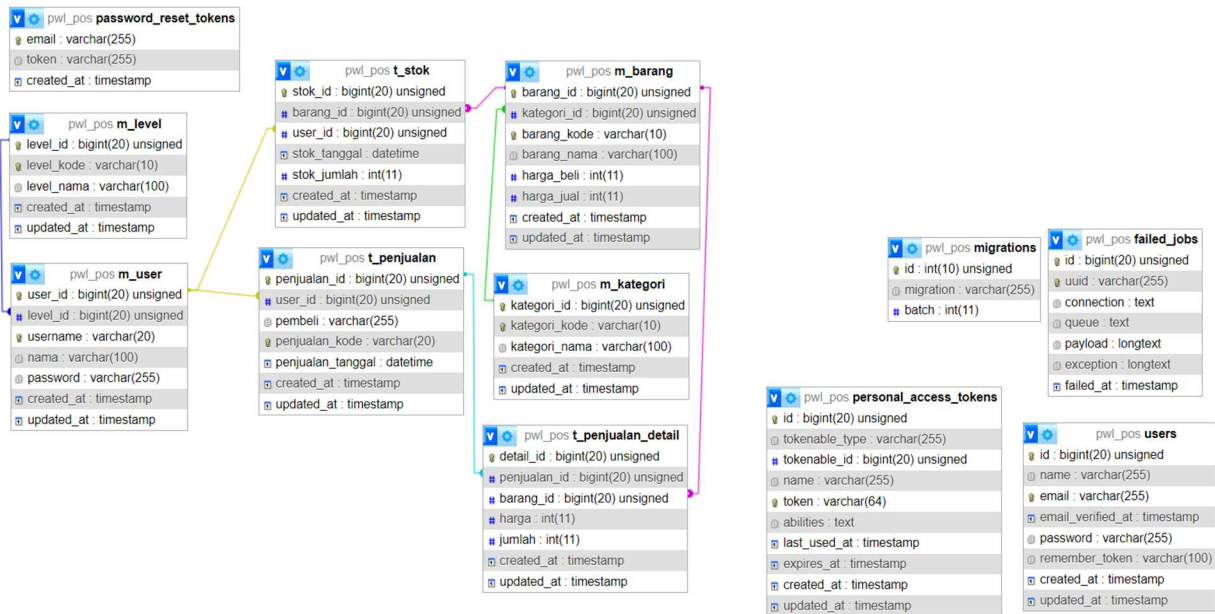
```
Schema::create('t_stok', function (Blueprint $table) {
    $table->id('stok_id');
    $table->unsignedBigInteger('barang_id');
    $table->unsignedBigInteger('user_id');
    $table->dateTime('stok_tanggal');
    $table->integer('stok_jumlah');
    $table->timestamps();

    $table->foreign('barang_id')->references('barang_id')->on('m_barang');
    $table->foreign('user_id')->references('user_id')->on('m_user');
});
```

Penjualan detail

```
Schema::create('t_penjualan_detail', function (Blueprint $table) {
    $table->id('detail_id');
    $table->unsignedBigInteger('penjualan_id');
    $table->unsignedBigInteger('barang_id');
    $table->integer('harga');
    $table->integer('jumlah');
    $table->timestamps();

    $table->foreign('penjualan_id')->references('penjualan_id')->on('t_penjualan');
    $table->foreign('barang_id')->references('barang_id')->on('m_barang');
});
```



Praktikum 3 : membuat file seeder

level

```
PS C:\xampp\htdocs\PHIL_2024> php artisan make:seeder levelSeeder
```

```
INFO Seeder [C:\xampp\htdocs\PHIL_2024\database\seeders\levelSeeder.php] created successfully.
```

```
PS C:\xampp\htdocs\PHIL_2024>
```

```
References: 0 Overrides: 0
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);
}
```

```
PS C:\xampp\htdocs\POS> php artisan db:seed --class=levelSeeder
```

```
INFO Seeding database.
```

				level_id	level_kode	level_nama	created_at	updated_at
<input type="checkbox"/>				1	ADM	Administrator	NULL	NULL
<input type="checkbox"/>				2	MNG	Manager	NULL	NULL
<input type="checkbox"/>				3	STF	Staff/Kasir	NULL	NULL

user

```
PS C:\xampp\htdocs\POS> php artisan make:seeder UserSeeder
```

```
INFO Seeder [C:\xampp\htdocs\POS\database\seeders\UserSeeder.php] created successfully.
```



```

public function run(): void
{
    $data = [
        [
            'user_id' => 1,
            'level_id' => 1,
            'username' => 'admin',
            'nama' => 'Administrator',
            'password' => Hash::make('12345'), //class untuk mengenkripsi/has
        ],
        [
            'user_id' => 2,
            'level_id' => 2,
            'username' => 'manager',
            'nama' => 'Manager',
            'password' => Hash::make('12345'),
        ],
        [
            'user_id' => 3,
            'level_id' => 3,
            'username' => 'staff',
            'nama' => 'Staff/Kasir',
            'password' => Hash::make('12345'),
        ],
    ];
    DB::table('m_user')->insert($data);
}

```

```
PS C:\xampp\htdocs\POS> php artisan db:seed --class=UserSeeder
```

```
INFO Seeding database.
```

<div><div><div>←</div><div>→</div></div><div></div></div>										user_id	level_id	username	nama	password	created_at	updated_at
<div><div><div><div></div></div></div><div><div>Edit</div><div>Copy</div><div>Delete</div></div></div>	1	1	admin	Administrator	\$2y\$12\$I5H39A5oLirYR9hCwDz7DOozKATY0NgUd/9kLnwKqD...	NULL	NULL									
<div><div><div><div></div></div></div><div><div>Edit</div><div>Copy</div><div>Delete</div></div></div>	2	2	manager	Manager	\$2y\$12\$iiYmVG4IEfDC8xllUyA9uadueqgapw7celaiyFh/Lo...	NULL	NULL									
<div><div><div><div></div></div></div><div><div>Edit</div><div>Copy</div><div>Delete</div></div></div>	3	3	staff	Staff/Kasir	\$2y\$12\$XudCZ.3/uUollTyoaR67/OAG.X.NC6Qgjac7D3FBNVE...	NULL	NULL									

Seeder tabel lain

```
PS C:\xampp\htdocs\POS> php artisan make:seeder KategoriSeeder
```

```
INFO Seeder [C:\xampp\htdocs\POS\database\seeders\KategoriSeeder.php] created successfully.
```

```
PS C:\xampp\htdocs\POS> php artisan make:seeder BarangSeeder
```

```
INFO Seeder [C:\xampp\htdocs\POS\database\seeders\BarangSeeder.php] created successfully.
```

```
PS C:\xampp\htdocs\POS> php artisan make:seeder StokSeeder
```

```
INFO Seeder [C:\xampp\htdocs\POS\database\seeders\StokSeeder.php] created successfully.
```

```
PS C:\xampp\htdocs\POS> php artisan make:seeder PenjualanSeeder
```

```
INFO Seeder [C:\xampp\htdocs\POS\database\seeders\PenjualanSeeder.php] created successfully.
```

```
PS C:\xampp\htdocs\POS> php artisan make:seeder PenjualanDetailSeeder
```

```
INFO Seeder [C:\xampp\htdocs\POS\database\seeders\PenjualanDetailSeeder.php] created successfully.
```

Kategori

```
PS C:\xampp\htdocs\POS> php artisan db:seed --class=KategoriSeeder
```

```
INFO Seeding database.
```

				kategori_id	kategori_kode	kategori_nama	created_at	updated_at
<input type="checkbox"/>				1	KTG001	Peralatan Mandi	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>				2	KTG002	Elektronik	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>				3	KTG003	Sembako	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>				4	KTG004	Makanan	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>				5	KTG005	Minuman	2024-03-07 07:56:30	2024-03-07 07:56:30

Barang

```
PS C:\xampp\htdocs\POS> php artisan db:seed --class=BarangSeeder
INFO Seeding database.
```

				barang_id	kategori_id	barang_kode	barang_nama	harga_beli	harga_jual	created_at	updated_at
<input type="checkbox"/>				1	1	BRG001	Sampo	35000	38000	2024-03-07 08:18:32	2024-03-07 08:18:32
<input type="checkbox"/>				2	1	BRG002	Sabun	10000	12000	2024-03-07 08:18:32	2024-03-07 08:18:32
<input type="checkbox"/>				3	2	BRG003	Kulkas	4000000	4400000	2024-03-07 08:18:32	2024-03-07 08:18:32
<input type="checkbox"/>				4	2	BRG004	TV	5000000	5500000	2024-03-07 08:18:32	2024-03-07 08:18:32
<input type="checkbox"/>				5	3	BRG005	Beras	50000	57000	2024-03-07 08:18:32	2024-03-07 08:18:32
<input type="checkbox"/>				6	3	BRG006	Gula	11000	13000	2024-03-07 08:18:32	2024-03-07 08:18:32
<input type="checkbox"/>				7	4	BRG007	Donat	4500	6000	2024-03-07 08:18:32	2024-03-07 08:18:32
<input type="checkbox"/>				8	4	BRG008	Roti	18000	20000	2024-03-07 08:18:32	2024-03-07 08:18:32
<input type="checkbox"/>				9	5	BRG009	Fanta	6000	8000	2024-03-07 08:18:32	2024-03-07 08:18:32
<input type="checkbox"/>				10	5	BRG010	Coca-cola	6000	7500	2024-03-07 08:18:32	2024-03-07 08:18:32

Stok

```
PS C:\xampp\htdocs\POS> php artisan db:seed --class=StokSeeder
INFO Seeding database.
```

				stok_id	barang_id	user_id	stok_tanggal	stok_jumlah	created_at	updated_at
<input type="checkbox"/>				1	1	1	2024-03-07 08:31:08	134	2024-03-07 08:31:08	2024-03-07 08:31:08
<input type="checkbox"/>				2	2	1	2024-03-07 08:31:08	157	2024-03-07 08:31:08	2024-03-07 08:31:08
<input type="checkbox"/>				3	3	1	2024-03-07 08:31:08	20	2024-03-07 08:31:08	2024-03-07 08:31:08
<input type="checkbox"/>				4	4	1	2024-03-07 08:31:08	32	2024-03-07 08:31:08	2024-03-07 08:31:08
<input type="checkbox"/>				5	5	1	2024-03-07 08:31:08	160	2024-03-07 08:31:08	2024-03-07 08:31:08
<input type="checkbox"/>				6	6	1	2024-03-07 08:31:08	235	2024-03-07 08:31:08	2024-03-07 08:31:08
<input type="checkbox"/>				7	7	1	2024-03-07 08:31:08	30	2024-03-07 08:31:08	2024-03-07 08:31:08
<input type="checkbox"/>				8	8	1	2024-03-07 08:31:08	15	2024-03-07 08:31:08	2024-03-07 08:31:08
<input type="checkbox"/>				9	9	1	2024-03-07 08:31:08	48	2024-03-07 08:31:08	2024-03-07 08:31:08
<input type="checkbox"/>				10	10	1	2024-03-07 08:31:08	79	2024-03-07 08:31:08	2024-03-07 08:31:08

Penjualan

```
PS C:\xampp\htdocs\POS> php artisan db:seed --class=PenjualanSeeder
INFO Seeding database.
```


← T →		▼ penjualan_id	user_id	pembeli	penjualan_kode	penjualan_tanggal	created_at	updated_at
<input type="checkbox"/>	 Edit	 Copy	 Delete	1	3 Dina	PJL001	2024-03-07 08:34:03	2024-03-07 08:34:03
<input type="checkbox"/>	 Edit	 Copy	 Delete	2	3 Dina	PJL002	2024-03-07 08:34:03	2024-03-07 08:34:03
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	3 Dina	PJL003	2024-03-07 08:34:03	2024-03-07 08:34:03
<input type="checkbox"/>	 Edit	 Copy	 Delete	4	3 Krisna	PJL004	2024-03-07 08:34:03	2024-03-07 08:34:03
<input type="checkbox"/>	 Edit	 Copy	 Delete	5	3 Krisna	PJL005	2024-03-07 08:34:03	2024-03-07 08:34:03
<input type="checkbox"/>	 Edit	 Copy	 Delete	6	3 Raffy	PJL006	2024-03-07 08:34:03	2024-03-07 08:34:03
<input type="checkbox"/>	 Edit	 Copy	 Delete	7	3 Raffy	PJL007	2024-03-07 08:34:03	2024-03-07 08:34:03
<input type="checkbox"/>	 Edit	 Copy	 Delete	8	3 Raffy	PJL008	2024-03-07 08:34:03	2024-03-07 08:34:03
<input type="checkbox"/>	 Edit	 Copy	 Delete	9	3 Putri	PJL009	2024-03-07 08:34:03	2024-03-07 08:34:03
<input type="checkbox"/>	 Edit	 Copy	 Delete	10	3 Putri	PJL010	2024-03-07 08:34:03	2024-03-07 08:34:03

Penjualan_detail

```
PS C:\xampp\htdocs\POS> php artisan db:seed --class=PenjualanDetailSeeder
```

INFO Seeding database.

			detail_id	penjualan_id	barang_id	harga	jumlah	created_at	updated_at
<input type="checkbox"/>				1	1	1	38000	1	2024-03-07 08:52:33
<input type="checkbox"/>				2	1	2	12000	2	2024-03-07 08:52:33
<input type="checkbox"/>				3	1	8	20000	2	2024-03-07 08:52:33
<input type="checkbox"/>				4	2	3	4400000	1	2024-03-07 08:52:33
<input type="checkbox"/>				5	2	4	5500000	1	2024-03-07 08:52:33
<input type="checkbox"/>				6	2	10	7500	2	2024-03-07 08:52:33
<input type="checkbox"/>				7	3	10	7500	3	2024-03-07 08:52:33
<input type="checkbox"/>				8	3	7	6000	10	2024-03-07 08:52:33
<input type="checkbox"/>				9	3	6	13000	2	2024-03-07 08:52:33
<input type="checkbox"/>				10	4	1	38000	5	2024-03-07 08:52:33
<input type="checkbox"/>				11	4	2	12000	2	2024-03-07 08:52:33
<input type="checkbox"/>				12	4	5	57000	1	2024-03-07 08:52:33
<input type="checkbox"/>				13	5	5	57000	1	2024-03-07 08:52:33
<input type="checkbox"/>				14	5	2	12000	2	2024-03-07 08:52:33
<input type="checkbox"/>				15	5	1	38000	5	2024-03-07 08:52:33
<input type="checkbox"/>				16	6	3	4400000	1	2024-03-07 08:52:33
<input type="checkbox"/>				17	6	7	6000	2	2024-03-07 08:52:33
<input type="checkbox"/>				18	6	6	13000	3	2024-03-07 08:52:33
<input type="checkbox"/>				19	7	6	13000	3	2024-03-07 08:52:33
<input type="checkbox"/>				20	7	4	5500000	1	2024-03-07 08:52:33
<input type="checkbox"/>				21	7	10	7500	2	2024-03-07 08:52:33
<input type="checkbox"/>				22	8	5	57000	1	2024-03-07 08:52:33
<input type="checkbox"/>				23	8	1	38000	1	2024-03-07 08:52:33
<input type="checkbox"/>				24	8	7	6000	1	2024-03-07 08:52:33
<input type="checkbox"/>				25	9	6	13000	1	2024-03-07 08:52:33
<input type="checkbox"/>				26	9	8	20000	1	2024-03-07 08:52:33
<input type="checkbox"/>				27	9	1	38000	1	2024-03-07 08:52:33
<input type="checkbox"/>				28	10	4	5500000	4	2024-03-07 08:52:33
<input type="checkbox"/>				29	10	3	4400000	6	2024-03-07 08:52:33
<input type="checkbox"/>				30	10	2	12000	2	2024-03-07 08:52:33

Praktikum 4 – Implementasi DB Façade

```
PS C:\xampp\htdocs\POS> php artisan make:controller levelController
```

```
INFO Controller [C:\xampp\htdocs\POS\app\Http\Controllers\levelController.php] created successfully.
```

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

INSERT

```
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';
}
```

←

→

↺

🏠

📘

127.0.0.1:8000/level

Insert data baru berhasil

<div>↳ T ↵</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
<input type="checkbox"/>		Edit		Copy		Delete	4 CUS	Pelanggan	2024-03-08 06:27:41	NULL

UPDATE

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

←

→

↺

🏠

📘

127.0.0.1:8000/level

Update data berhasil. Jumlah data yang diupdate: 1 baris

<div><div><div>←</div><div>→</div></div><div></div></div>						level_id	level_kode	level_nama	created_at	updated_at
<input type="checkbox"/>	<div><div></div><div>Edit</div></div>	<div><div></div><div>Copy</div></div>	<div><div></div><div>Delete</div></div>	1	ADM	Administrator	NULL	NULL		
<input type="checkbox"/>	<div><div></div><div>Edit</div></div>	<div><div></div><div>Copy</div></div>	<div><div></div><div>Delete</div></div>	2	MNG	Manager	NULL	NULL		
<input type="checkbox"/>	<div><div></div><div>Edit</div></div>	<div><div></div><div>Copy</div></div>	<div><div></div><div>Delete</div></div>	3	STF	Staff/Kasir	NULL	NULL		
<input type="checkbox"/>	<div><div></div><div>Edit</div></div>	<div><div></div><div>Copy</div></div>	<div><div></div><div>Delete</div></div>	4	CUS	Customer	2024-03-08 06:27:41	NULL		

Delete

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

←

→

↺

🏠

📘

127.0.0.1:8000/level

Delete data berhasil. Jumlah data yang dihapus: 1 baris

<div><div><div>←</div><div>→</div></div><div></div></div>						level_id	level_code	level_nama	created_at	updated_at
<div><div><input type="checkbox"/></div><div><div><div></div><div></div><div></div></div></div><div>Edit</div><div><div><div></div><div></div><div></div></div></div><div>Copy</div><div><div><div></div><div></div><div></div></div></div><div>Delete</div></div>	1	ADM	Administrator	NULL	NULL					
<div><div><input type="checkbox"/></div><div><div><div></div><div></div><div></div></div></div><div>Edit</div><div><div><div></div><div></div><div></div></div></div><div>Copy</div><div><div><div></div><div></div><div></div></div></div><div>Delete</div></div>	2	MNG	Manager	NULL	NULL					
<div><div><input type="checkbox"/></div><div><div><div></div><div></div><div></div></div></div><div>Edit</div><div><div><div></div><div></div><div></div></div></div><div>Copy</div><div><div><div></div><div></div><div></div></div></div><div>Delete</div></div>	3	STF	Staff/Kasir	NULL	NULL					

Menampilkan data

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

```
resources > views > level.blade.php > html
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Data Level Pengguna</title>
8 </head>
9
10 <body>
11   <h1>Data Level Pengguna</h1>
12   <table border="1" cellpadding="2" cellspacing="0">
13     <tr>
14       <th>ID</th>
15       <th>Kode Level</th>
16       <th>Nama Level</th>
17     </tr>
18     @foreach ($data as $d)
19     <tr>
20       <td>{{ $d -> level_id }} </td>
21       <td>{{ $d -> level_kode }} </td>
22       <td>{{ $d -> level_nama }} </td>
23     </tr>
24     @endforeach
25   </table>
26
27 </body>
28
29 </html>
```

← → ↻ 🏠 ⓘ 127.0.0.1:8000/level

Data Level Pengguna

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

Praktikum 5 – Implementasi Query Builder

```
PS C:\xampp\htdocs\POS> php artisan make:controller KategoriController
```

```
INFO Controller [C:\xampp\htdocs\POS\app\Http\Controllers\KategoriController.php] created successfully.
```

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

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

← → ↺ 🏠 ⓘ 127.0.0.1:8000/kategori

Insert data baru berhasil

← T →		kategori_id	kategori_kode	kategori_nama	created_at	updated_at
<input type="checkbox"/>	 Edit  Copy  Delete	1	KTG001	Peralatan Mandi	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>	 Edit  Copy  Delete	2	KTG002	Elektronik	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>	 Edit  Copy  Delete	3	KTG003	Sembako	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>	 Edit  Copy  Delete	4	KTG004	Makanan	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>	 Edit  Copy  Delete	5	KTG005	Minuman	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>	 Edit  Copy  Delete	6	SMK	Snack/Makanan Ringan	2024-03-08 07:11:35	NULL

Hapus

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

← → ↺ 🏠 ⓘ 127.0.0.1:8000/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	KTG001	Peralatan Mandi	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>	Edit Copy Delete	2	KTG002	Elektronik	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>	Edit Copy Delete	3	KTG003	Sembako	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>	Edit Copy Delete	4	KTG004	Makanan	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>	Edit Copy Delete	5	KTG005	Minuman	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>	Edit Copy Delete	6	SMK	Camilan	2024-03-08 07:11:35	NULL

Delete

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



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	KTG001	Peralatan Mandi	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>	Edit Copy Delete	2	KTG002	Elektronik	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>	Edit Copy Delete	3	KTG003	Sembako	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>	Edit Copy Delete	4	KTG004	Makanan	2024-03-07 07:56:30	2024-03-07 07:56:30
<input type="checkbox"/>	Edit Copy Delete	5	KTG005	Minuman	2024-03-07 07:56:30	2024-03-07 07:56:30

Menampilkan data

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

```

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Data kategori Barang</title>
</head>

<body>
  <h1>Data Kategori Barang</h1>
  <table border="1" cellpadding="2" cellspacing="0">
    <tr>
      <th>ID</th>
      <th>Kode Kategori</th>
      <th>Nama Kategori</th>
    </tr>
    @foreach ($data as $d)
    <tr>
      <td>{{ $d->kategori_id }}</td>
      <td>{{ $d->kategori_kode }}</td>
      <td>{{ $d->kategori_nama }}</td>
    </tr>
    @endforeach
  </table>
</body>
</html>

```

Data Kategori Barang

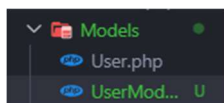
ID	Kode Kategori	Nama Kategori
1	KTG001	Peralatan Mandi
2	KTG002	Elektronik
3	KTG003	Sembako
4	KTG004	Makanan
5	KTG005	Minuman

Praktikum 6 – Implementasi Eloquent ORM

```

PS C:\xampp\htdocs\POS> php artisan make:model UserModel
INFO Model [C:\xampp\htdocs\POS\app\Models\UserModel.php] created successfully.

```



```

0 references | 0 implementations
class UserModel extends Model
{
    use HasFactory;

    0 references
    protected $table = 'm_user';
    0 references
    protected $primaryKey = 'user_id';
}

```

```

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

```

```

use App\Models\UserModel;

```

```

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

```

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <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>

```

Data User

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

```

// reference | u 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
}

```

Nambahi sendiri

<input type="checkbox"/>	Edit	Copy	Delete	3 STT	Staff/Kasir	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete	4 CUS	Customer	NULL	NULL

Data User

ID	Username	Nama	ID Level Pengguna
1	admin	Administrator	1
2	manager	Manager	2
3	staff	Staff/Kasir	3
10	customer-1	Pelanggan	4

			user_id	level_id	username	nama	password	created_at	updated_at
<input type="checkbox"/>	Edit	Copy	Delete	1	1 admin	Administrator	\$2y\$12\$15H39A5oLlrYR9hCwDz7DOozKATY0NgUd/9kLnwKqD...	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete	2	2 manager	Manager	\$2y\$12\$1iYmVGe4IEfDC8xllUyA9uadueqgapw7celaiyFh/Lo...	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete	3	3 staff	Staff/Kasir	\$2y\$12\$XudCZ.3/uUoIItyoaR67/OAG.X.NC6Qgjac7D3FBNVE...	NULL	NULL
<input type="checkbox"/>	Edit	Copy	Delete	10	4 customer-1	Pelanggan	\$2y\$12\$TFmxc46oOcZh7ty08iM2qe7q4iGoXt1D4H3Gcc.reb3...	NULL	NULL

```
//update You, 1 second ago • Uncommitted changes
// tambah data user 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]);
```

← → ↺ 🏠 ⓘ 127.0.0.1:8000/user

Data User

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

			user_id	level_id	username	nama	password	created_at	updated_at				
<input type="checkbox"/>		Edit		Copy		Delete	1	1	admin	Administrator	\$2y\$12\$I5H39A5oLlrYR9hCwDz7DOozKATfY0NgUd/9kLnwKqD...	NULL	NULL
<input type="checkbox"/>		Edit		Copy		Delete	2	2	manager	Manager	\$2y\$12\$iiYmVG4IEfDC8xllUyA9uadueqgapw7celaiyFh/Lo...	NULL	NULL
<input type="checkbox"/>		Edit		Copy		Delete	3	3	staff	Staff/Kasir	\$2y\$12\$XudCZ.3/uUollTyoaR67/OAG.X.NC6Qgjac7D3FBNVE...	NULL	NULL
<input type="checkbox"/>		Edit		Copy		Delete	10	4	customer-1	Pelanggan Pertama	\$2y\$12\$TFmxc46oOcZh7ty08iM2qe7q4iGoXt1D4H3Gcc.reb3...	NULL	2024-03-08 08:28:23

Penutup

1. Pada Praktikum 1 - Tahap 5, apakah fungsi dari APP_KEY pada file setting .env Laravel?

Jawab:

Terdapat 2 fungsi utama,

- Enkripsi Data: APP_KEY digunakan untuk mengenkripsi data dalam aplikasi Anda. Setiap data yang dienkripsi dalam aplikasi menggunakan APP_KEY di belakang layar. Misalnya, data sesi, token CSRF, dan cookie menggunakan APP_KEY untuk keamanan.
- Bukan untuk Hashing: APP_KEY tidak digunakan untuk mengenkripsi kata sandi. Laravel menggunakan teknik hashing untuk mengamankan kata sandi.

2. Pada Praktikum 1, bagaimana kita men-generate nilai untuk APP_KEY?

Jawab:

- Buka terminal atau command prompt.
- Jalankan perintah berikut untuk menghasilkan kunci aplikasi:
php artisan key:generate
- Setelah menjalankan perintah ini, kunci baru akan dihasilkan dan disimpan dalam file .env.

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

Jawab:

Secara default terdapat 4 file migrasi,

- create_users_table.php: digunakan untuk membuat tabel pengguna (users) di basis data. Ini termasuk kolom seperti nama pengguna (username), alamat surel (email), dan kata sandi (password).
- create_password_resets_table.php: untuk membuat tabel untuk menyimpan token pengaturan ulang kata sandi (password reset) yang biasanya digunakan dalam proses pemulihan kata sandi pengguna.
- create_failed_jobs_table.php: membuat tabel untuk menyimpan pekerjaan yang gagal dijalankan oleh sistem antrian (queue). Ini berguna untuk pelacakan dan pemantauan ketika pekerjaan gagal dieksekusi.
- create_sessions_table.php: membuat tabel untuk menyimpan informasi sesi pengguna. Ini digunakan oleh Laravel untuk menyimpan status dan data sesi pengguna.

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

Jawab:

Fungsi \$table->timestamps() untuk secara otomatis menambahkan dua kolom ke tabel yang sedang dibuat, yaitu created_at dan updated_at,

- created_at: Kolom ini akan menyimpan timestamp (waktu) ketika sebuah record baru pertama kali dimasukkan ke dalam tabel.
- updated_at: Kolom ini akan menyimpan timestamp ketika record yang sudah ada di dalam tabel diperbarui (diubah).

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

Jawab:

Fungsi \$table->id() dalam sebuah file migrasi Laravel menghasilkan sebuah kolom auto-increment bertipe data integer yang digunakan sebagai primary key (kunci utama) untuk tabel

tersebut. Sehingga secara otomatis nilainya meningkat (auto-increment) setiap kali sebuah record baru ditambahkan ke dalam tabel

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

Jawab:

Keduanya memiliki tujuan yang sama yaitu untuk membuat kolom auto-increment sebagai primary key untuk tabel `m_level`

Perbedaannya terletak pada nama kolom primary key yang dihasilkan. Laravel secara otomatis akan membuat kolom primary key dengan nama `'id'`, sedangkan kalau `$table->id('level_id')`, secara eksplisit memberikan nama kolom primary key dengan nama `'level_id'`.

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

Jawab:

Digunakan untuk menetapkan konstrain unik (unique constraint) pada sebuah kolom di dalam tabel basis data, yaitu memastikan bahwa setiap nilai di dalam kolom tersebut harus unik, atau dalam kata lain, tidak boleh ada duplikat.

8. Pada Praktikum 2.2 - Tahap 2, kenapa kolom `level_id` pada tabel `m_user` menggunakan `$tabel->unsignedBigInteger('level_id')`, sedangkan kolom `level_id` pada tabel `m_level` menggunakan `$tabel->id('level_id')` ?

Jawab:

- `$table->unsignedBigInteger('level_id')` pada tabel `m_user`:
menunjukkan bahwa kolom `level_id` dalam tabel `m_user` adalah sebuah kolom yang berisi nilai yang berhubungan dengan kolom primary key (biasanya bertipe big integer) di tabel lain, yaitu `m_level`.
`UnsignedBigInteger` digunakan untuk mengindikasikan bahwa kolom tersebut harus berisi nilai bilangan bulat yang tidak negatif.
- `$table->id('level_id')` pada tabel `m_level`:
menunjukkan bahwa kolom `level_id` di sini adalah primary key (biasanya bertipe big integer) dan memiliki nama kolom yang khusus yaitu `level_id`.

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

Jawab:

Kelas Hash dalam Laravel bertujuan untuk menyediakan fasilitas untuk hashing atau mengacak data, khususnya kata sandi (password). Hashing adalah proses mengubah data menjadi nilai yang acak dan tidak dapat diubah kembali, yang disebut hash value.

Pada kodenya, `Hash::make('1234')` akan menghasilkan hash dari kata sandi `'1234'`. Dengan demikian, Laravel akan mengacak kata sandi tersebut dan mengembalikan hasilnya dalam bentuk hash value. Seperti berikut ini:

		user_id	level_id	username	nama	password	created_at	updated_at
<input type="checkbox"/>	Edit		1	1	admin	Administrator	\$2y\$12\$5H39A5oLlrYR9hCwDz7DOozKATfY0NgUd/9kLnwKqD...	NULL
<input type="checkbox"/>	Edit		2	2	manager	Manager	\$2y\$12\$iiYmVGe4IEIDC8xliUyA9uadueqgapw7celaiyFh/Lo...	NULL
<input type="checkbox"/>	Edit		3	3	staff	Staff/Kasir	\$2y\$12\$XudCZ.3/uUolITyOaR67/OAG X.NC6Qgjac7D3FBNVE...	NULL

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

Jawab:

Tanda tanya (?) dalam query builder Laravel merupakan bagian dari parameter binding atau parameter binding. Ini adalah teknik yang digunakan untuk mengamankan query SQL dari serangan injeksi SQL dan juga membuat query lebih dinamis.

Ketika menggunakan tanda tanya (?) dalam query builder, dapat mengganti nilai sebenarnya dari parameter tersebut saat mengeksekusi query, dan Laravel akan secara otomatis melakukan sanitasi dan escaping terhadap nilai tersebut, sehingga mencegah serangan injeksi SQL.

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

Jawab:

Memiliki tujuan untuk menentukan nama tabel dan primary key yang terkait dengan model tersebut.

- Properti `$table` digunakan untuk menentukan nama tabel yang terkait dengan model. Dengan menetapkan `$table` ke nilai `'m_user'`, berarti memberitahu Laravel bahwa model tersebut berkaitan dengan tabel basis data yang disebut `'m_user'`. Ini memungkinkan Laravel untuk secara otomatis menggunakan nama tabel yang benar ketika melakukan operasi database menggunakan model tersebut, seperti mengambil data, menyimpan data, atau membuat query lainnya.
- Properti `$primaryKey` digunakan untuk menentukan nama primary key dari tabel yang terkait dengan model. Dengan menetapkan `$primaryKey` ke nilai `'user_id'`, berarti memberitahu Laravel bahwa primary key untuk tabel `'m_user'` adalah kolom `'user_id'`. Ini memungkinkan Laravel untuk memahami dan menggunakan primary key yang benar ketika melakukan operasi-operasi yang memerlukan primary key, seperti menyimpan atau mengambil data.

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

Jawab:

Db façade

```
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';
}
```

Query builder

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

Eloquent ORM

```
0 references | 0 implementations
class UserModel extends Model
{
    use HasFactory;

    0 references
    protected $table = 'm_user';
    0 references
    protected $primaryKey = 'user_id';
}
```

Menurutku untuk saat ini lebih mudah query builder, karena membangun query dengan lebih mudah menggunakan sintaks yang lebih deklaratif. Meskipun masih menulis query dalam bentuk kode PHP, Query Builder memberikan lebih banyak abstraksi dari SQL murni, sehingga lebih mudah dipahami dan dipelajari. Dan juga menulis query lebih sedikit kode daripada menggunakan DB Facade, tetapi masih mempertahankan kontrol yang tinggi atas query yang dihasilkan