

DATABASE MIGRATION

A. Mengatur Setting Database

1. Cari file **.env**, kemudian cari baris berikut:

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

Ubah sesuai nama database, username, dan password yang kalian buat
Mudah bukan?

2. Buka **config/database.php**

```
'mysql' => [
    'driver' => 'mysql',
    'url' => env('DATABASE_URL'),
    'host' => env('DB_HOST', '127.0.0.2'),
    'port' => env('DB_PORT', '3306'),
    'database' => env('DB_DATABASE', 'forge'),
    'username' => env('DB_USERNAME', 'forge'),
    'password' => env('DB_PASSWORD', ''),
    'unix_socket' => env('DB_SOCKET', ''),
    'charset' => 'utf8',
    'collation' => 'utf8_unicode_ci',
    'prefix' => '',
    'prefix_indexes' => true,
    'strict' => true,
    'engine' => 'InnoDB',
    'options' => extension_loaded('pdo_mysql') ? array_filter([
        PDO::MYSQL_ATTR_SSL_CA => env('MYSQL_ATTR_SSL_CA'),
    ]) : [],
],
```

- Ubah charset dan collation dari utf8mb4 menjadi utf8
- Ubah engine menjadi InnoDB. Default engine MYSQL adalah MyISAM. Dengan mengubah default engine menjadi InnoDB dapat mencegah agar tidak error saat menjalankan migration.

B. Mengetahui dan Membuat Migration

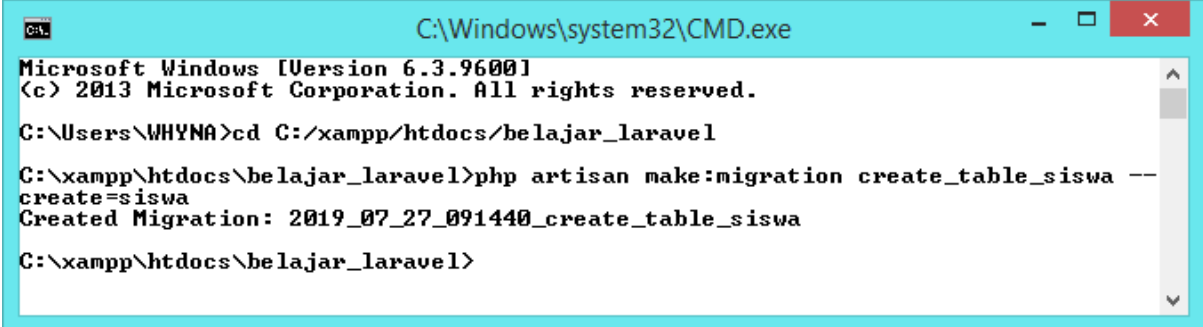
"Migration" adalah cara baru dalam membuat database. Dengan migration, kita bisa membuat class-class dengan method yang tersedia untuk membuat database, tabel, kolom, memberikan index, primary key, foreign key dan sebagainya melalui command prompt yaitu menggunakan perintah **artisan**.

Apa keuntungan menggunakan migration???

- Kita hanya perlu membuat satu class untuk membuat database dari berbagai macam RDBMS. Misalnya untuk mengganti database, kita tidak perlu membuat ulang, cukup dengan menjalankan class yang telah dibuat
- Semua perubahan yang kita lakukan pada database dicatat pada suatu tabel database tersendiri sehingga bisa membatalkan pembuatan atau perubahan
- Dapat menghemat waktu karena kita hanya cukup mengubah class migration dan menjalankannya kembali tanpa perlu membuat semua perintah database dari awal

Untuk dapat memahami, mari buat migration untuk tabel siswa

1. Buka command Prompt, masuk ke dalam folder projek laravel kalian masing-masing
2. Berikan perintah artisan seperti berikut:



```
C:\Windows\system32\CMD.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\WHYNA>cd C:/xampp/htdocs/belajar_laravel

C:\xampp\htdocs\belajar_laravel>php artisan make:migration create_table_siswa --
create=siswa
Created Migration: 2019_07_27_091440_create_table_siswa

C:\xampp\htdocs\belajar_laravel>
```

3. Buka folder **database/migrations**, buka file yang namanya mengandung **create_table_siswa.php** ini merupakan file migration yang baru kita buat
4. Ubah isi class pada schema menjadi seperti berikut:

```
class CreateTableSiswa extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('siswa', function (Blueprint $table) {
            $table->increments('id');
            $table->string('nisan', 4)->unique();
            $table->string('nama_siswa', 30);
            $table->date('tanggal_lahir');
            $table->enum('jenis_kelamin', ['L', 'P']);
            $table->timestamps();

        });
    }
}
```

5. Buat database, agar lebih mudah gunakan phpMyAdmin
6. Menjalankan Migrations

```

C:\Windows\system32\CMD.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\WHYNA>cd C:/xampp/htdocs/belajar_laravel

C:\xampp\htdocs\belajar_laravel>php artisan migrate
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table
Migrating: 2019_07_27_091440_create_table_siswa
Migrated: 2019_07_27_091440_create_table_siswa
  
```

7. Jika dilihat pada phpMyAdmin maka akan muncul tabel sebagai berikut:

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> migrations	★ Browse Structure Search Insert Empty Drop	3	InnoDB	utf8_unicode_ci	16 KiB	-
<input type="checkbox"/> password_resets	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_unicode_ci	32 KiB	-
<input type="checkbox"/> siswa	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_unicode_ci	32 KiB	-
<input type="checkbox"/> users	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_unicode_ci	32 KiB	-
4 tables	Sum	3	InnoDB	latin1_swedish_ci	112 KiB	0 B

Keterangan:

- Tabel **migrations** digunakan untuk mencatat (tracking) migration yang kita lakukan
- Tabel **users** adalah tabel default Laravel untuk menangani proses autentikasi
- Tabel **password_resets** digunakan untuk urusan reset password, misal lupa password
- Tabel **siswa** tabel yang dihasilkan dari pembuatan migration create_tabel_siswa