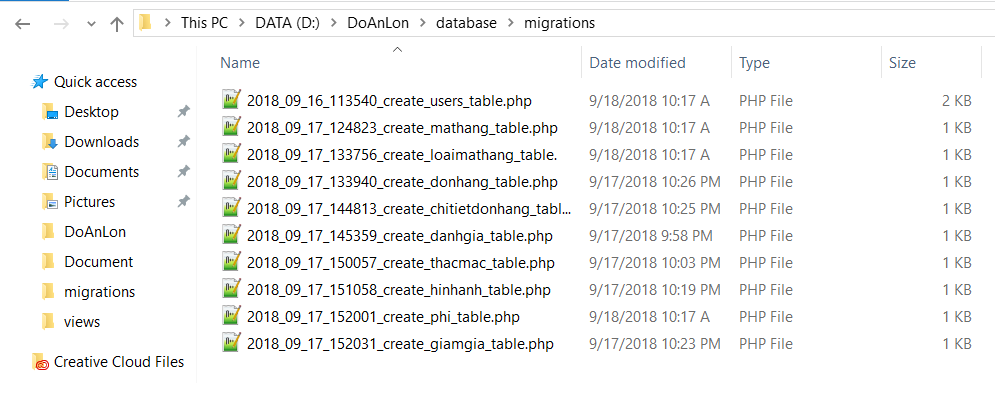
**Migration:Các tạo và thực thi**

**1. Migration là gì ?**

- Migration trong Lavarel giống như một control database có tác dụng quản lý cũng như lưu trữ lại cấu trúc của database giúp cho việc sửa đổi database trở lên dễ dàng hơn.

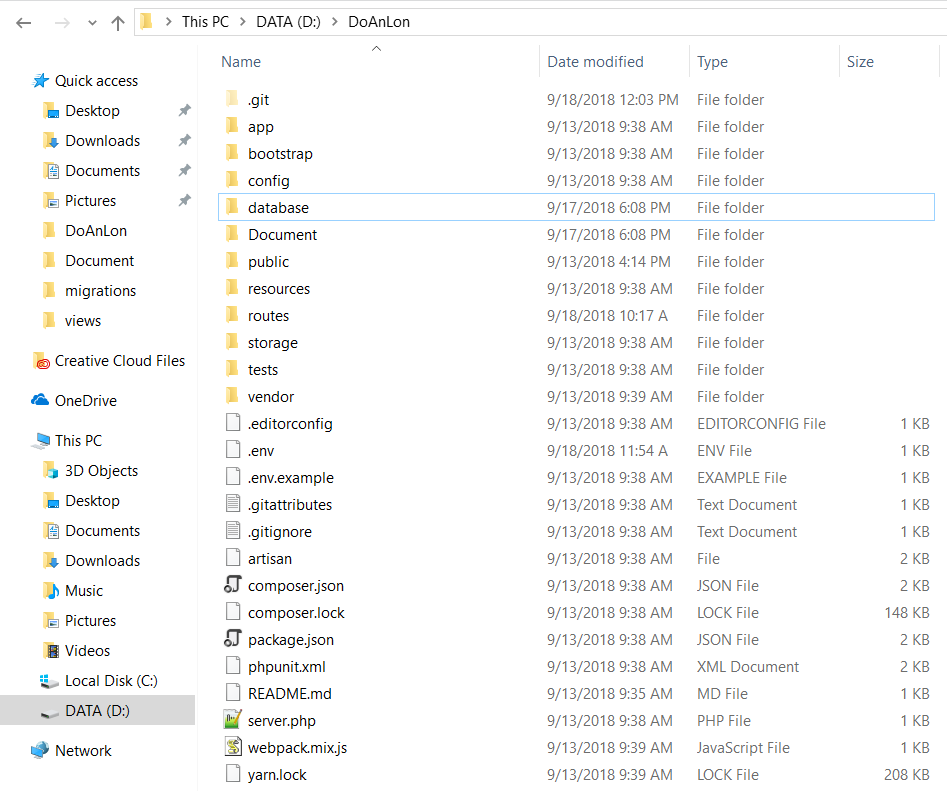
- Migration sẽ được lưu trữ tại **App\database\migrations.**



**2. Cách tạo Migration.**

- Ta có thể Migration theo các bước sau:

+ B1: Mở Folder Project:

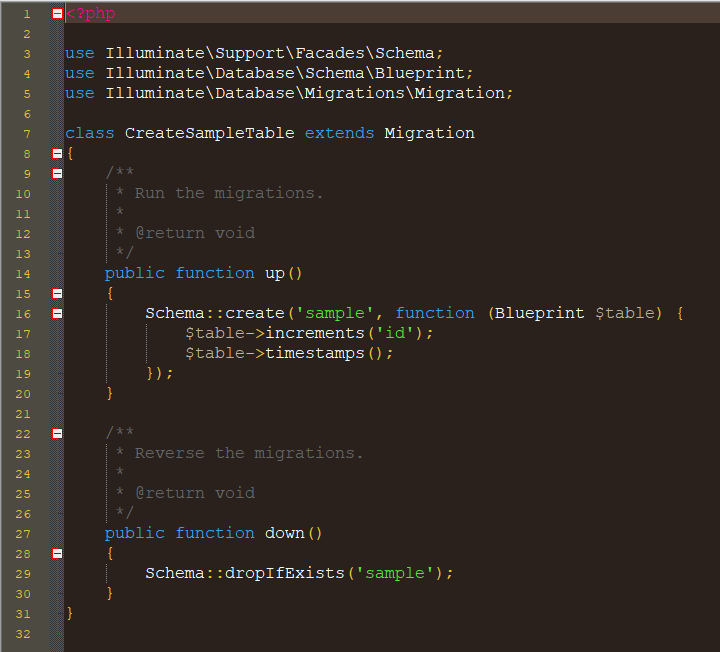


- B2: Mở cmd và gõ lệnh sau :

**php artisan make:migration TenMigrate --create=TableName**

-Sau khi tạo thành công thì sẽ có thông báo như sau:

****

****

Cấu trúc bên trong của Migration

**3.Hàm up và down**

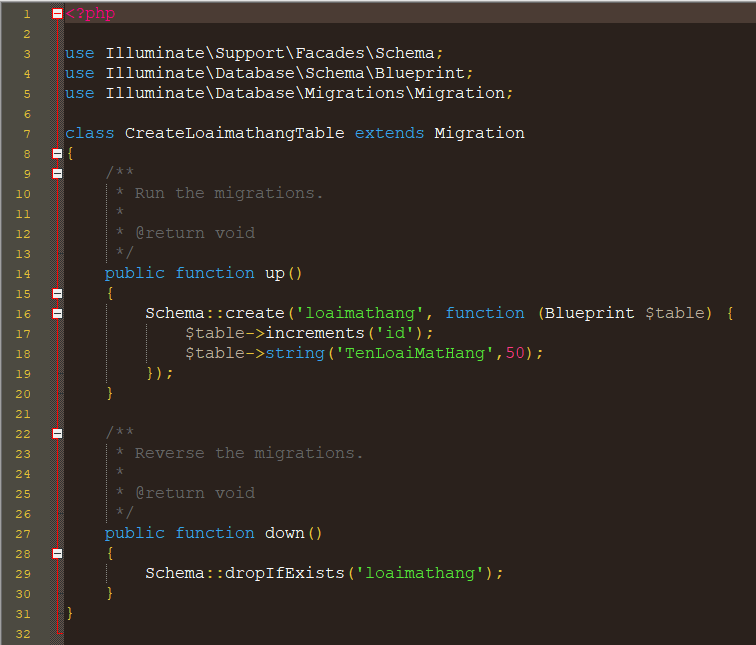
- Hàm up trong Migrations có tác dụng thực thi migration

- Hàm down trong Migrations có tác dụng thực thi đoạn lệnh rollback(trở về trước đó).

**4.Cú pháp trong Migration**

| **Command** | **Description** |
| --- | --- |
| $table->bigIncrements('id'); | Incrementing ID (primary key) using a "UNSIGNED BIG INTEGER" equivalent. |
| $table->bigInteger('votes'); | BIGINT equivalent for the database. |
| $table->binary('data'); | BLOB equivalent for the database. |
| $table->boolean('confirmed'); | BOOLEAN equivalent for the database. |
| $table->char('name', 4); | CHAR equivalent with a length. |
| $table->date('created\_at'); | DATE equivalent for the database. |
| $table->dateTime('created\_at'); | DATETIME equivalent for the database. |
| $table->dateTimeTz('created\_at'); | DATETIME (with timezone) equivalent for the database. |
| $table->decimal('amount', 5, 2); | DECIMAL equivalent with a precision and scale. |
| $table->double('column', 15, 8); | DOUBLE equivalent with precision, 15 digits in total and 8 after the decimal point. |
| $table->enum('choices', ['foo', 'bar']); | ENUM equivalent for the database. |
| $table->float('amount', 8, 2); | FLOAT equivalent for the database, 8 digits in total and 2 after the decimal point. |
| $table->increments('id'); | Incrementing ID (primary key) using a "UNSIGNED INTEGER" equivalent. |
| $table->integer('votes'); | INTEGER equivalent for the database. |
| $table->ipAddress('visitor'); | IP address equivalent for the database. |
| $table->json('options'); | JSON equivalent for the database. |
| $table->jsonb('options'); | JSONB equivalent for the database. |
| $table->longText('description'); | LONGTEXT equivalent for the database. |
| $table->macAddress('device'); | MAC address equivalent for the database. |
| $table->mediumIncrements('id'); | Incrementing ID (primary key) using a "UNSIGNED MEDIUM INTEGER" equivalent. |
| $table->mediumInteger('numbers'); | MEDIUMINT equivalent for the database. |
| $table->mediumText('description'); | MEDIUMTEXT equivalent for the database. |
| $table->morphs('taggable'); | Adds unsigned INTEGER taggable\_id and STRING taggable\_type. |
| $table->nullableMorphs('taggable'); | Nullable versions of the morphs() columns. |
| $table->nullableTimestamps(); | Nullable versions of the timestamps()columns. |
| $table->rememberToken(); | Adds remember\_token as VARCHAR(100) NULL. |
| $table->smallIncrements('id'); | Incrementing ID (primary key) using a "UNSIGNED SMALL INTEGER" equivalent. |
| $table->smallInteger('votes'); | SMALLINT equivalent for the database. |
| $table->softDeletes(); | Adds nullable deleted\_at column for soft deletes. |
| $table->string('email'); | VARCHAR equivalent column. |
| $table->string('name', 100); | VARCHAR equivalent with a length. |
| $table->text('description'); | TEXT equivalent for the database. |
| $table->time('sunrise'); | TIME equivalent for the database. |
| $table->timeTz('sunrise'); | TIME (with timezone) equivalent for the database. |
| $table->tinyInteger('numbers'); | TINYINT equivalent for the database. |
| $table->timestamp('added\_on'); | TIMESTAMP equivalent for the database. |
| $table->timestampTz('added\_on'); | TIMESTAMP (with timezone) equivalent for the database. |
| $table->timestamps(); | Adds nullable created\_at and updated\_atcolumns. |
| $table->timestampsTz(); | Adds nullable created\_at and updated\_at(with timezone) columns. |
| $table->unsignedBigInteger('votes'); | Unsigned BIGINT equivalent for the database. |
| $table->unsignedInteger('votes'); | Unsigned INT equivalent for the database. |
| $table->unsignedMediumInteger('votes'); | Unsigned MEDIUMINT equivalent for the database. |
| $table->unsignedSmallInteger('votes'); | Unsigned SMALLINT equivalent for the database. |
| $table->unsignedTinyInteger('votes'); | Unsigned TINYINT equivalent for the database. |
| $table->uuid('id'); | UUID equivalent for the database. |

Ví dụ :



**5.Thực thi Migration**

-Sau khi đã tạo xong thì chúng ta cần thực thi Migration:

+Mở cmd và gõ lệnh sau: php artisan migrate

-Nếu thành công thì tại …/phpmyadmin sẽ hiện các migration đã được tạo tại Folder: migrations

