

Laravel migrations

php artisan make:migration create_members

```
class CreateMembers extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('members', function (Blueprint $table) {
            $table->bigIncrements('member_id');
            $table->string('first_name');
            $table->string('last_name');
            $table->string('nick_name');
            $table->date('birth_day');
            $table->string('gender');
            $table->string('phone');
            $table->string('email');
            $table->string('image_url')->nullable();
            $table->year('year_study');
            $table->integer('gen_id');
            $table->integer('school_id');
            $table->integer('region_id');
            $table->integer('edu_status');
            $table->integer('graduate_id')->nullable();
            $table->text('work_exp')->nullable();
            $table->text('work_now')->nullable();
            $table->text('education_high')->nullable();
            $table->timestamps();
        });
    }

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

php artisan make:migration create_generation

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateGeneration extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('generation', function (Blueprint $table) {
            $table->bigIncrements('gen_id');
            $table->integer('year');
            $table->integer('gen_count');
            $table->timestamps();
        });
    }

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

Colum type, Data type : <https://laravel.com/docs/5.6/migrations#creating-columns>

php artisan make:migration create_advisor

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateGraduate extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('graduate', function (Blueprint $table) {
            $table->bigIncrements('graduate_id');
            $table->year('year');
            $table->string('project_name');
            $table->string('project_type');
            $table->string('expert_special');
            $table->timestamps();
        });
    }

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

php artisan make:migration create_graduate

```
class CreateMembers extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('members', function (Blueprint $table) {
            $table->bigIncrements('member_id');
            $table->string('first_name');
            $table->string('last_name');
            $table->string('nick_name');
            $table->date('birth_day');
            $table->string('gender');
            $table->string('phone');
            $table->string('email');
            $table->string('image_url')->nullable();
            $table->year('year_study');
            $table->integer('gen_id');
            $table->integer('school_id');
            $table->integer('region_id');
            $table->integer('edu_status');
            $table->integer('graduate_id')->nullable();
            $table->text('work_exp')->nullable();
            $table->text('work_now')->nullable();
            $table->text('education_high')->nullable();
            $table->timestamps();
        });
    }

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

Colum type, Data type : <https://laravel.com/docs/5.6/migrations#creating-columns>

หลักการทำงาน

Route

- App/route/web.php
- กำหนดเส้นทางหรือ URI ทุกครั้งที่ Client ส่ง Request อะไรมา Route ก็จะตรวจสอบว่าเราได้ทำการผูก url กับ ฟังก์ชัน หรือยัง ก่อนจะส่ง Request ไปทำงานยังฟังก์ชันนั้นๆ

```
> Route::get('foo', function () {  
  
    return 'Hello World';  
  
});
```

```
> Route::get('/user', 'UserController@index');    method get (เรียกแสดง)
```

```
> Route::post('/user', 'UserController@register'); method post (ส่งข้อมูล หรือฟอร์ม)
```

```
> Route::any('/user', 'UserController@register'); ได้ทั้ง get และ post
```

```
> Route::redirect('/here', '/there', 301);        การกำหนดให้เปลี่ยนเส้นทาง
```

```
> Route::view('/welcome', 'welcome');            การแสดงหน้า view โดยตรง
```

Model Controller View (MVC)

Model — เป็นส่วนของข้อมูลดิบ หรืออาจจะมองว่าเป็นส่วนเชื่อมกับฐานข้อมูลที่เป็น Object (Database)

Controller — สื่อกลาง ของ Model และ View เป็นผู้เตรียมข้อมูล, จัดการข้อมูลที่ไหลเข้าออก, เรียกใช้ Method ของ Model (PHP)

View — ส่วนตอบสนองกับผู้ใช้ อาจจะพูดว่ามันก็คือหน้าเว็บของเราบนเว็บเบราว์เซอร์ใช้นั่นเอง (HTML+CSS)

การสร้าง Model

- php artisan make:model Members การสร้าง model member
- หรือ php artisan make:model Members -m การสร้าง model member พร้อมกับการสร้าง migration table member

```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;
use DB;

class Members extends Model
{
    //
}
```

ต้องเพิ่ม Use DB;

การสร้าง Controller

- php artisan make:controller MembersController
ไฟล์จะอยู่ใน App/http/controller

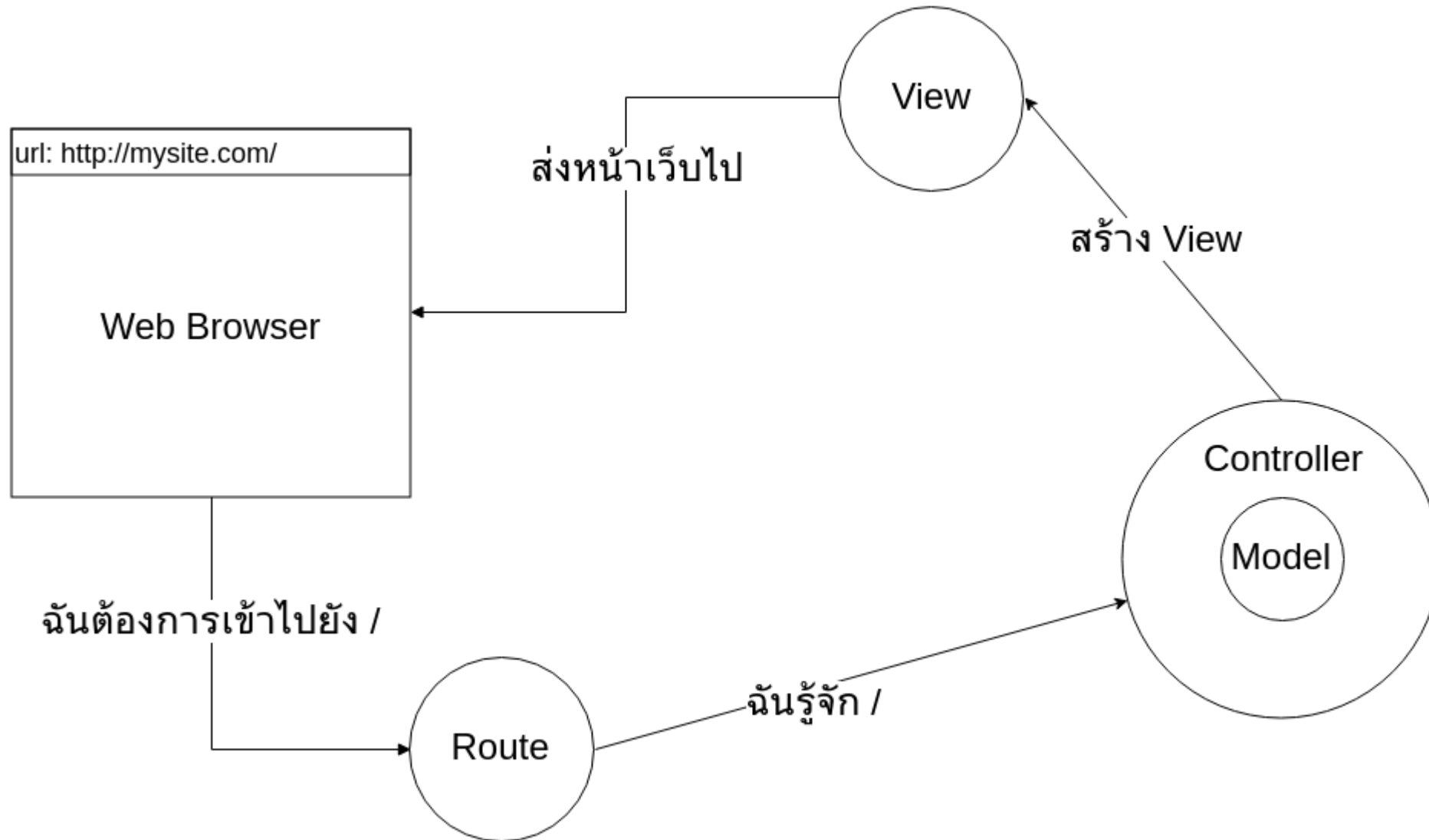
```
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use App\Members as member;
7  class membersController extends Controller
8  {
9      //
10 }
11
```

การประกาศใช้ model

การสร้าง View

- สร้างไฟล์ .blade.php เช่น member.blade.php ที่ ../resource/view
- เพื่อใช้ในการสร้างหน้าแสดงผล และติดต่อกับผู้ใช้งาน
- ในส่วนนี้มักจะมี html เป็นภาษาหลักและมีการแสดงผลข้อมูล object ที่ได้จากการ query โดยใช้ภาษา php

สรุปหลักการทำงาน



Lab

- ดาวน์โหลดไฟล์ member.blade.php ที่ <https://bit.ly/3593mYu>
- ดาวน์โหลดไฟล์ cis_directory ที่ <https://bit.ly/31P0Zry>
จากนั้น ทำการ import database ใหม่
- สร้างหน้าแสดงรายการสมาชิกทั้งหมด พร้อมสามารถคลิกเพื่อเข้าไปดูรายละเอียดได้
- ทำหน้าแสดงรายละเอียดข้อมูลของแต่ละ member ในหน้า welcome.blade.php ให้เปลี่ยนเป็นหน้าแสดง
รุ่นทั้งหมดพร้อมระบุจำนวน นักศึกษาในรุ่น และจำนวนข้อมูลจริงที่มี โดยออกแบบอย่างไรก็ได้ เมื่อคลิกที่แต่ละรุ่น จะแสดงข้อมูลรายชื่อของ นศ ในรุ่นนั้นๆ พร้อมก็สามารถดูรายละเอียดได้ โดยใช้ url เก่าที่เคยทำไว้
- สร้างแบบฟอร์มเพิ่มข้อมูล generation
- สร้างแบบฟอร์มเพิ่มข้อมูล members (การบ้าน)