

---

**LAB 4 MÔN PHP3: SỬ DỤNG MIGRATION VÀ SEEDER****MỤC TIÊU:**

Kết thúc bài thực hành này bạn có khả năng

- ✓ Sử dụng thành thạo migration để tạo/ sửa cấu trúc database
- ✓ Sử dụng seeder để thêm/sửa/xóa dữ liệu trong các table

**NỘI DUNG****Chuẩn bị:**

- Cài Laravel vào folder **la4**
- Tạo database có tên **la4** và import file `la_news.sql`
- Mở file `.env` cấu hình kết nối đến database vừa tạo

**Bài 1: Migration create table****1. Tạo migration create table dienthoai**

**a. Yêu cầu:** Tạo migration để tạo table **dienthoai** gồm các field

- id (integer , primary key, increament )
- tenDT (string, 30, duy nhất không được trùng)
- moTa (string, 1000, nullable)
- ngayCapNhat (DateTime)
- gia (double 8,2)
- giaKM(double 8,2 , default =0)
- urlHinh (string, 200, nullable)
- soLuongTonKho (integer, default=0)
- hot (boolean, default =0)
- anHien (boolean, default=1)

**b. Hướng dẫn thực hiện**

- Trong command line, chạy lệnh

```
php artisan make:migration taoTableDienThoai --create=dienthoai
```

- Vào folder `database/migrations`, mở file migration mới tạo và code lại trong hàm `up` để được như sau:

```
public function up() {
    Schema::create('dienthoai', function (Blueprint $table) {
        $table->id();
        $table->string('tenDT',30)->unique();
        $table->string('moTa',1000)->nullable();
        $table->dateTime('ngayCapNhat');
        $table->double('gia',8,2);
        $table->double('giaKM',8,2)->nullable();
        $table->double('urlHinh',200)->nullable();
        $table->integer('soLuongTonKho')->default(0);
        $table->boolean('hot')->default(0);
        $table->boolean('anHien')->default(1);
        $table->timestamps();
    });
}
```

- Chạy migration

```
php artisan migrate
```

Xong vào database , phải thấy table dienthoai với cấu trúc như khai báo

## 2. Tạo migration create table thanhvien

**a. Yêu cầu:** Tạo migration để tạo 1 table có tên **thanhvien** gồm các cột sau:

- id(integer, primary key, increament),
- hoTen(string, 100),
- password (string, 100)
- email (string , 200)
- randomKey(string, 100, nullable)
- active(boolean, default = 0)
- idGroup(integer, default =0)

**b. Gợi ý**

```
php artisan make:migration create_thanhvien_table
```

```
public function up() {
    Schema::create('thanhvien', function (Blueprint $table) {
        $table->increments('id');
        $table->string('hoTen',100);
        $table->string('password',100);
        $table->string('email',200);
        $table->string('randomKey',100)->nullable();
        $table->boolean('active')->default(0);
        $table->boolean('idGroup')->default(0);
        $table->timestamps();
    });
}
```

### 3. Tạo migration create table loaisp

a. **Yêu cầu:** Tạo migration để tạo 1 table có tên loiSP gồm các cột sau:

- id (integer , primary key, increament)
- tenLoai (string, 30, unique)
- thuTu (integer, default = 0)
- anHien (boolean, default =1)
- urlHinh (string, 200, nullable)

b. **Gợi ý**

```
php artisan make:migration taotableLoaisp --create=loiSP
```

```
public function up() {
    Schema::create('loiSP', function (Blueprint $table) {
        $table->id();
        $table->string('tenLoai',30)->unique();
        $table->integer('thuTu')->default(0);
        $table->boolean('anHien')->default(1);
        $table->string('urlHinh',200)->nullable();
        $table->timestamps();
    });
}
```

## Bài 2: Migration chỉnh cấu trúc table

### 1. Migration chỉnh cấu trúc bảng dienthoai

a. **Yêu cầu:** Migration sẽ thực hiện các việc như sau:

- Thêm 3 field **baiViet**(text, nullable), **ghiChu**(string, 500, nullable), **idLoai** (unsigned Big Integer) vào bảng dienthoai
- Khai báo khóa ngoại là idLoai để thiết lập quan hệ với bảng loaisp (field id )

b. **Gợi ý thực hiện**

- Tạo migration:

```
php artisan make:migration suaTableDienThoai --table=dienthoai
```

- Mở file mới tạo và code trong hàm up

```
Schema::table('dienthoai', function (Blueprint $table) {
    $table->text('baiViet')->nullable();
    $table->string('ghiChu', 500)->nullable();
    $table->unsignedBigInteger('idLoai');
    $table->foreign('idLoai')->references('id')->on('loaisp');
});
```

- Chạy migration

### Bài 3: Seeder tạo dữ liệu cho các table

#### 1. Seeder chèn loại sản phẩm

- Chạy lệnh:

```
php artisan make:seeder chenLoaisp
```

- Mở file seeder vừa tạo và code trong hàm run

```
DB::table('loaisp')->insert([
    ['tenLoai'=>'Sam sung', 'thuTu'=>1, 'anHien'=>1],
    ['tenLoai'=>'HTC', 'thuTu'=>2, 'anHien'=>1],
    ['tenLoai'=>'Apple', 'thuTu'=>3, 'anHien'=>1],
    ['tenLoai'=>'LG', 'thuTu'=>4, 'anHien'=>1],
    ['tenLoai'=>'Motorola', 'thuTu'=>5, 'anHien'=>1],
    ['tenLoai'=>'Mobel', 'thuTu'=>6, 'anHien'=>0],
]);
```

- Ở đầu seeder, thêm lên **use DB;**
- Chạy seeder:

```
php artisan db:seed --class=chenLoaisp
```

#### 2. Seeder chèn điện thoại

Tạo seeder chèn 300 bộ dữ liệu sản phẩm điện thoại, mỗi bộ lấy 3 tên Oppo XA, Iphone xs Max, Nokia Pro làm cơ sở để phát sinh dữ liệu. Giá của Oppo tạo ngẫu nhiên từ 700000 đến 1000000, Iphone xs max từ 500000 đến 800000, Nokia Pro từ 250000 đến 500000. Các giá trị khác phát sinh tùy ý.

```
php artisan make:seeder chendienthoai
```

Mở file seeder vừa tạo và code trong hàm run

```
public function run() {
    for($i=0;$i<300; $i++ ) {
        DB::table('dienthoai')->insert([
            ['tenDT'=>'Oppo XA ' . $i, 'gia'=>mt_rand(700000, 1000000),'ngayCapNhat'=>Now(),'idLoai'=>2 ],
            ['tenDT'=>'Iphone xs Max ' . $i, 'gia'=>mt_rand(500000, 800000),'ngayCapNhat'=>Now(),'idLoai'=>3],
            ['tenDT'=>'Nokia Pro ' . $i, 'gia'=>mt_rand(250000, 500000),'ngayCapNhat'=>Now(),'idLoai'=>1],
        ]);
    }
}
```

Chạy seeder nhé

### 3. Seeder chèn thành viên

- Chèn 100 user với họ tên ghép ngẫu nhiên các giá trị trong 3 dãy sau, mật khẩu mã hóa từ chữ hehe. Các email phát sinh ngẫu nhiên và có domain gmail.com.

```
php artisan make:seeder chenthanhvien
```

```
public function run() {
    $ho = ['Nguyễn', 'Lê', 'Phan', 'Đỗ', 'Hò', 'Võ', 'Bùi'];
    $lot = ['Thị', 'Văn', 'Đức', 'Ngọc', 'Hoàng', 'Minh', 'Kim', ''];
    $ten = ['Tâm', 'Thảo', 'Hải', 'Hòa', 'Hào', 'Thanh', 'Tú', 'Hậu', 'Phương'];
    for($i=0; $i<100; $i++){
        $ht = Arr::random($ho) . ' ' . Arr::random($lot) . ' ' . Arr::random($ten);
        DB::table('thanhvien')->insert([
            'hoTen'=>$ht,
            'email'=>Str::random(5).'@gmail.com',
            'password' => bcrypt('hehe')
        ]);
    }
}
```

- Chạy seeder: sinh viên tự thực hiện

### \*\*\* Yêu cầu nộp bài:

SV nén file (hoặc share thư mục google drive) bao gồm các yêu cầu đã thực hiện trên, nộp LMS đúng thời gian quy định của giảng viên. KHÔNG NỘP BÀI COI NHƯ KHÔNG CÓ ĐIỂM.

--- Hết ---