

# LẬP TRÌNH WEB PHP NÂNG CAO

GV: Trần Thanh Tuấn



Laravel

# Nội dung

- Collection
- Accessors & Mutators

# Giới thiệu

```
$dsCauHoi = CauHoi::where('linh_vuc_id', 1)->get();  
foreach ($dsCauHoi as $cauHoi) {  
    //  
}
```

Illuminate \ Database \ Eloquent \ Collection

# Illuminate \ Database \ Eloquent \ Collection

- `contains()`
- Kiểm tra một thể hiện model có tồn tại trong collection không?

```
$dsCauHoi->contains(1);
```

```
$dsCauHoi->contains(CauHoi::find(1));
```

# Illuminate \ Database \ Eloquent \ Collection

- `diff()`
- Trả về tất cả các model không có trong collection cho trước

```
$dsCauHoi = $dsCauHoi->diff(  
    CauHoi::whereIn('id', [1, 2, 3])->get());
```

# Illuminate \ Database \ Eloquent \ Collection

- `except()`
- Trả về tất cả các model không có trong tập giá trị của khoá chính cho trước

```
$dsCauHoi = $dsCauHoi->except([1, 2, 3]);
```

# Illuminate \ Database \ Eloquent \ Collection

- `find()`
- Tìm 01 model với giá trị khoá chính cho trước

```
$cauHoi = $dsCauHoi->find(1);
```

# Illuminate \ Database \ Eloquent \ Collection

- `intersect()`
- Trả về tất cả các model có trong collection cho trước

```
$dsCauHoi = $dsCauHoi->intersect(  
    CauHoi::whereIn('id', [1, 2, 3])->get());
```



# Illuminate \ Database \ Eloquent \ Collection

- `only()`
- Trả về tất cả các model có giá trị khoá chính trong tập giá trị cho trước

```
$dsCauHoi = $dsCauHoi->only([1, 2, 3]);
```

# Illuminate \ Database \ Eloquent \ Collection

- `unique()`
- Trả về tất cả các model duy nhất trong collection

```
$dsCauHoi = $dsCauHoi->unique();
```

# Illuminate \ Database \ Eloquent \ Collection

- `modelKeys()`
- Trả về tập giá trị khoá chính của các model trong collection cho trước

```
$dsCauHoi->modelKeys();
```

# Illuminate \ Database \ Eloquent \ Collection

- `makeHidden()`
- Ẩn các thuộc tính của các model

```
$dsCauHoi->makeHidden(['dap_an', 'deleted_at']);
```

# Illuminate \ Database \ Eloquent \ Collection

- `makeVisible()`
- Hiện các thuộc tính “ẩn” của các model

```
$dsCauHoi->makeVisible(['dap_an', 'deleted_at']);
```

# Illuminate \ Database \ Eloquent \ Collection

- Tham khảo thêm về các phương thức của Collection

<https://laravel.com/docs/5.8/collections>

# Accessors & Mutators

- Accessor (Getter)
- Mutators (Setter)
- Cho phép định dạng giá trị của các thuộc tính Eloquent khi truy xuất hoặc gán giá trị trên 01 thể hiện model

# Accessors & Mutators

- Quy tắc định nghĩa Accessor & Mutator

```
public function getTenThuocTinhAttribute($value)
{
    // $value là giá trị của thuộc tính ten_thuoc_tinh
}
```

```
public function getTenLinhVucAttribute($value)
{
    // In hoa ký tự đầu tiên của tên lĩnh vực
    return ucfirst($value);
}
```

```
echo LinhVuc::find(1)->ten_linh_vuc;
```



# Accessors & Mutators

- Quy tắc định nghĩa Accessor & Mutator

```
public function getTenThuocTinhMoiAttribute()  
{  
    // code here  
}
```

```
public function getLinhVucAttribute()  
{  
    // In hoa ký tự đầu tiên của tên lĩnh vực  
    return ucfirst($this->ten_linh_vuc);  
}
```

```
echo LinhVuc::find(1)->linh_vuc;
```

# Accessors & Mutators

- Quy tắc định nghĩa Accessor & Mutator

```
public function setTenThuocTinhAttribute($value)
{
    // $value là giá trị cần gán vào thuộc tính ten_thuoc_tinh
}
```

```
public function setTenLinhVucAttribute($value)
{
    // chuyển toàn bộ ký tự trong $value thành chữ thường
    $this->ten_linh_vuc = strtolower($value);
}
```

```
LinhVuc::find(1)->ten_linh_vuc = "Âm Nhạc";
```

# Bài tập

- Hoàn thiện chức năng Quản lý một Quan hệ (bảng) trong CSDL:
  - Lấy danh sách
  - Thêm mới
  - Cập nhật
  - Xóa
  - Thùng rác + khôi phục + xóa vĩnh viễn
- Lưu ý:
  - Hỏi trước khi xóa
  - Thông báo khi thêm, cập nhật, xóa thành công