



LARAVEL FRAMEWORK

BÀI 2: SỬ DỤNG QUEY BUILDER PHẦN 1

- ☐ Tìm hiểu về thư viện Query Builder
- ☐ Sử dụng Query Builder để tương tác với DB



❑ Phần 1

- ❖ Cấu hình kết nối db trong Laravel
- ❖ Giới thiệu thư viện Query Builder
- ❖ Các hàm thường dùng trong query
- ❖ Lấy record đầu tiên
- ❖ Lấy (select) danh sách các record
- ❖ Lấy giá trị trong một field
- ❖ Các hàm tổng kết
- ❖ Query có where, orderby, limit

❑ Phần 2

- ❖ Sử dụng Query có join bảng
- ❖ Kiểm tra record tồn tại
- ❖ Chèn record và lấy id
- ❖ Chèn nhiều record cùng lúc
- ❖ Cập nhật record
- ❖ Xóa record với query
- ❖ Tổng hợp 1: routing, lấy data, nạp view
- ❖ Tổng hợp 2: hiện data theo tham số



- ❑ Laravel hỗ trợ kết nối đến nhiều loại database qua PDO như sqlite, mysql , sql server, postgresql. Khai báo kết nối mặc định trong **config/database.php**
- ❑ Thông số kết nối đến db trong file **.env** , gồm:

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1      ← Địa chỉ server database
DB_PORT=3306           ← port của database server
DB_DATABASE=la_news    ← Tên database kết nối đến
DB_USERNAME=root       ← username kết nối đến database
DB_PASSWORD=           ← password kết nối đến database
```

- ❑ Bạn có thể tạo database và import database từ file **la_new.sql** trong tài nguyên môn học.

- ❑ Query builder là 1 công cụ có sẵn trong Laravel. Dùng để select, insert, update, delete dữ liệu trong các table rất dễ dàng.
- ❑ Trong Laravel, sử dụng query builder thông qua class **DB**.
- ❑ Sử dụng lệnh sau trước khi dùng DB (trong controller):
use DB;

DATABASE:Query Builder



- ❑ Tạo 1 query để lấy dữ liệu trong 1 table

```
$query = DB::table('tblName')  
->select('tenField1', 'tênField2')  
->where('tenField', 'op', giá trị)  
->orderBy(tênFied, 'asc|desc')  
->offset(from)  
->limit(numrerord)
```

- **tblName** là tên table trong database
- **op** là phép toán : > =, < , <=, > =
- **from**: lấy từ record thứ mấy
- **numrecord**: lấy bao nhiêu record

- ❑ Khi đã tạo query, có thể thực thi các hàm trong query để lấy mọi record, một record, cột cột, các hàm tổng kết...(xem slide sau)

❑ Các hàm thường dùng trong query :

- ❖ \$query-> **get()** : Lấy danh sách các record thỏa mãn điều kiện trong query.
- ❖ \$query-> **first()** : lấy record đầu tiên
- ❖ \$query-> **value('tenField')** : lấy giá trị của cột tenField
- ❖ \$query-> **pluck('tenField')** : lấy các giá trị của field tenField
- ❖ \$query-> **max('tenField')** : lấy giá trị lớn nhất của tenField
- ❖ \$query-> **min('tenField')** : lấy giá trị nhỏ nhất của tenField
- ❖ \$query-> **sum('tenField')** : lấy tổng các giá trị của tenField
- ❖ \$query-> **avg('tenField')** : lấy giá trị trung bình của tenField
- ❖ \$query-> **count('tenField')** : đếm số record trong query

- ❑ Để lấy record đầu tiên trong query, bạn dùng hàm first. Hàm này trả về một đối tượng
- ❑ Ví dụ: routes/web.php

```
Route::get('/db1/', function(){  
    //tạo query  
    $query = DB::table('loaiTin')->select('id', 'ten');  
  
    $kq = $query->first(); //record đầu tiên trong query  
    print_r($kq);  
    echo "<h4> {$kq->ten} </h4>";  
});
```

← → ↻ ⓘ localhost:8000/db1

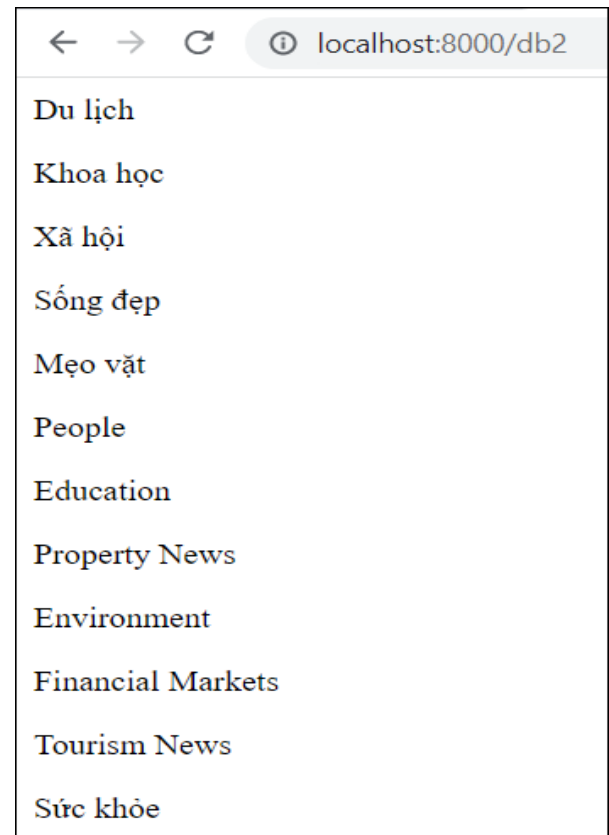
stdClass Object ([id] => 3 [ten] => Du lịch)

Du lịch

- ❑ Để lấy danh sách các record, dùng hàm **get()** của query.
Hàm này trả về 1 collection các record từ database

```
Route::get('/db2/', function(){
    $query = DB::table('loaiTin')
        ->select('id', 'ten');

    //lấy ds record, mỗi record là obj
    $kq = $query->get();
    foreach ($kq as $row) {
        echo "<p> {$row->ten} </p>";
    }
});
```

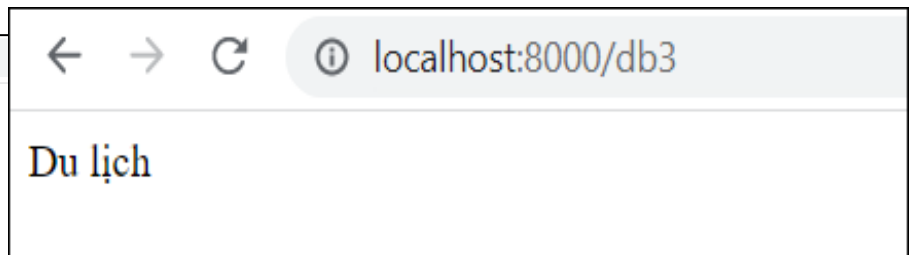
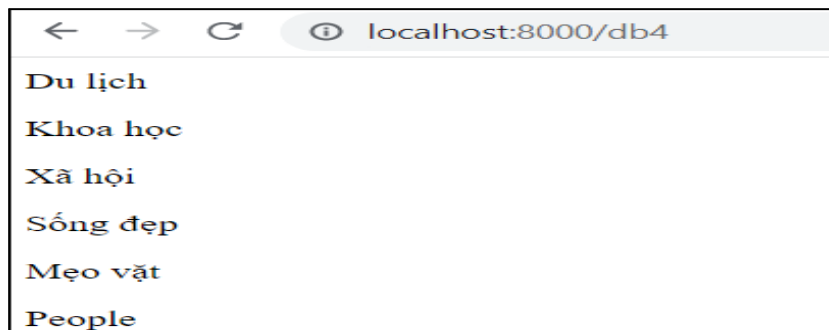


- ❑ Để lấy giá trị trong 1 cột nào đó, dùng hàm **value()**

```
Route::get('/db3/', function(){  
    $t = DB::table('loaiTin')->where('id','=',3)->value('ten');  
    echo $t;  
});
```

- ❑ Lấy mảng các giá trị của 1 field với hàm pluck

```
Route::get('/db4/', function(){  
    $arr= DB::table('loaiTin')->pluck('ten');  
    foreach( $arr as $ten) echo "<p> {$ten} </p>";  
});
```



❑ Các hàm tổng kết max, min, sum, avg

```
Route::get('/db5/', function(){
    $sotin = DB::table('tin')->WHERE('noiBat',1)
    ->count();
    echo "Số tin nổi bật: $sotin <br/>";

    $mn = DB::table('tin')->max("ngayDang");
    echo "Tin mới nhất đăng ngày: $mn <br/>";

    $sn = DB::table('tin')->min("ngayDang");
    echo "Tin sớm nhất đăng ngày: $sn <br/>";

    $tb = DB::table('tin')->avg("xem");
    echo "Xem trung bình $tb <br/>";

    $tong = DB::table('tin')->sum("xem");
    echo "Tổng xem $tong <br/>";
});
```

← → ↻ ⓘ localhost:8000/db5

Số tin nổi bật: 56
Tin mới nhất đăng ngày: 2018-12-19
Tin sớm nhất đăng ngày: 2008-03-30
Xem trung bình 157.8294
Tổng xem 33302

QUERY CÓ WHERE, ORDER BY, LIMIT

Tạo query có dùng where, orderby, limit, offset... Sau đó lấy list record

```
Route::get('/db7/', function(){
    $query = DB::table('loaiTin')
        ->where('anHien', '=', 1)
        ->orderBy('ten', 'ASC')
        ->offset(5)
        ->limit(10);
    $kq = $query->get();
    foreach ($kq as $row) {
        echo "<p>{$row->ten} </p>";
    }
});
```

localhost:8000/db7	
Mẹo vặt	
People	
Property News	
Sống đẹp	
Sức khỏe	
Tourism News	
Xã hội	



DEMO

Thực hiện select, insert,
update, delete





FPT POLYTECHNIC



LARAVEL FRAMEWORK

BÀI 2: SỬ DỤNG QUEY BUILDER PHẦN 2

www.poly.edu.vn

- ❑ Tạo query có join bảng, sau đó lấy danh sách record.

```
Route::get('/db8/', function(){
    $query = DB::table('tin')
        ->Join('loaiTin', 'tin.idLT', '=', 'loaitin.id')
        ->select('tin.id', 'tin.tieuDe', 'loaitin.ten');
    $kq = $query->get();
    foreach ($kq as $row) {
        echo "<p>{$row->tieuDe} ({$row->ten}) </p>";
    }
});
```

← → ↻ ⓘ localhost:8000/db8
Hoàng hôn trên sông Mê Kông (Du lịch)
Tận cùng nỗi đau (Xã hội)
Thăm trang trại bò lớn nhất Việt Nam (Xã hội)
Ngày sẽ dài 25 tiếng (Khoa học)
Đo tiếng cười thật hay già (Khoa học)
Mạo hiểm"với đảo (Du lịch)
Vườn quốc gia Pù Mát (Du lịch)
Cổ tích cho những hy vọng không thành... (Sống đẹp)

- ❑ Để kiểm tra một record tồn tại không, dùng hàm exist của query.

```
Route::get('/db6/', function(){  
    $kq = DB::table('loaiTin')  
        ->where('id', 1111)->exists();  
    if (!$kq) echo "Không tồn tại loại tin này";  
});
```



❑ Tạo query và chèn 1 record

```
Route::get('/db9/', function(){
    DB::table('tin')
        ->insert([ 'tieude'=>'Việt nam vô địch', 'idLT'=>1 ]);
});
```

❑ Chèn 1 record và lấy id của dòng mới chèn

```
Route::get('/db10/', function(){
    $id = DB::table('tin')
        ->insertGetId(['tieude'=>'VN vô địch AFF']);
    echo "id của record mới chèn: $id";
});
```

← → ↻ ⓘ localhost:8000/db10

id của record mới chèn: 789

- ❑ Để chèn nhiều record, truyền cho hàm insert một array dữ liệu

```
Route::get('/db11/', function(){
    DB::table('tin')->insert( [
        [ 'TieuDe'=>'VN mến yêu', 'idLT'=>1 ],
        [ 'TieuDe'=>'VN yêu mến', 'idLT'=>1 ]
    ]);
});
```

- ❑ Để cập nhật record , dùng hàm update của query. Truyền tham số cho hàm update là array các field và giá trị mới.

```
Route::get('/db12/', function(){  
    DB::table('tin')->where('id','=',800)  
    ->update( [ 'TieuDe'=>'Việt nam tuyệt vời', 'idLT'=>3 ] );  
});
```

- ❑ Để xóa record trong table, dùng hàm delete của query

```
Route::get('/db13/', function(){  
    DB::table('tin')->where('id', '=', 802)->delete();  
});
```

Bài tổng hợp này để hiển thị các tin được xem nhiều nhất từ database ra trang web. Các việc thực hiện gồm :

1. Tạo controller
2. Tạo route dẫn vào action
3. Tạo action trong controller
4. Tạo view hiển thị dữ liệu
5. Xem kết quả

1. Tạo controller

```
php artisan make:controller TintucController
```

2. Tạo route dẫn vào action của controller: trong web.php

```
use App\Http\Controllers\TintucController;  
Route::get('/txn/', [TintucController::class, 'tinXN']);
```

3. Tạo action trong controller :

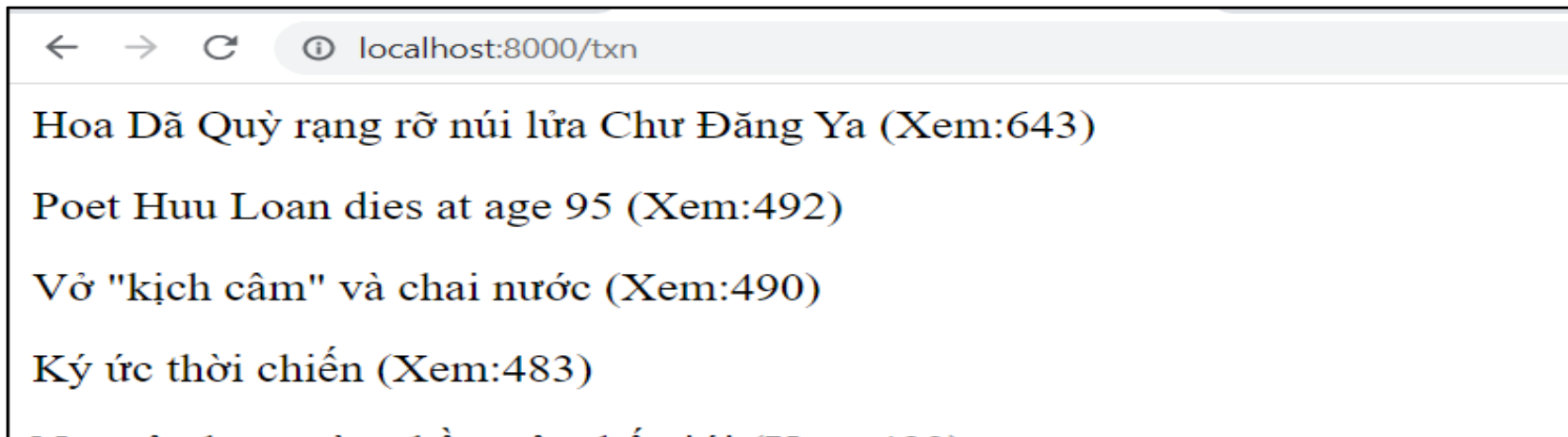
- ❖ Lấy dữ liệu trong database
- ❖ Nạp view và truyền dữ liệu cho view

```
class TintucController extends Controller {  
  function tinXN(){  
    $kq = DB::table("tin")  
    ->select('id','tieuDe', 'xem')  
    ->orderBy('xem','desc')->limit(10)  
    ->get();  
    return view('txn', ['data'=> $kq]);  
  }  
}
```

4. Tạo resources/views/txn.php

```
<?php
foreach ($data as $row){
    echo "<p>{$row->tieuDe} (Xem:{$row->xem})</p>";
}
?>
```

5. Xem kết quả : <http://localhost:8000/txn>



Có thể định dạng css nhanh bằng tag <style> ngay trong view

Bài tổng hợp này dùng để tạo chức năng hiển thị chi tiết một tin từ database theo tham số id của tin. Các việc thực hiện gồm :

1. Tạo route dẫn vào action
 2. Tạo action trong controller
 3. Tạo view hiển thị dữ liệu
 4. Xem kết quả
-

1. Tạo route vào action của controller : trong web.php

```
Route::get('/tin/{id}',[TintucController::class,'detail']);
```


2. Tạo action trong controller :

- ❖ Lấy dữ liệu trong database
- ❖ Nạp view và truyền dữ liệu cho view

```
function detail($id){  
    $kq = DB::table("tin")->WHERE('id', $id)->first();  
    return view("detail", ['tin'=> $kq]);  
}
```

3. Tạo view: resources/views/detail.php

```
<h1> <?=$tin->tieuDe?></h1>  
<em>  
Ngày đăng <?=date('d/m/Y',strtotime($tin->ngayDang)) ?>  
</em>  
<h2> <?=$tin->tomTat?></h2> <hr>  
<div id="content"> <?=$tin->noiDung?></div>
```

4. Test kết quả: <http://localhost:8000/tin/1>

    localhost:8000/tin/1

Hoàng hôn trên sông Mê Kông

Ngày đăng 01/04/2008

Bên ghé đá thị xã Savăn, thủ phủ miền Trung và Hạ Lào của đất nước Hoa Chăm Pa, ngược Chùa Vàng Thái Lan, chúng tôi ngồi hàng giờ để ngắm hoàng hôn đang lịm dần trên mặt s

Bắt chợt, câu thơ của chàng thi sĩ “còm” Võ Văn Luyến ở quê nhà hiện về *Tôi tin những dòng sông chảy ngược/Là những dòng sông tự chảy...*

1. Ở thời khắc giao hòa giữa ngày và đêm, lòng Mê Kông lenh lảnh một ráng hồng rực rỡ. Ánh hào quang còn sót lại dường như cũng ướt đầm m chèo quá vắng.

Giữa dòng sông nhìn từ bến phà Savăn sang Mục Đa Hán (Thái Lan), loi thôi một cồn hoang cô độc. Cái cồn hoang này chắc chắn không có ai c loài chim đến tá túc. Chiều về, có khi cả cồn hoang đầy ắp, lạnh lói tiếng chim...

Ở Savăn muốn ngắm hoàng hôn, không đâu có thể đem lại thật nhiều cảm xúc như bên bờ sông Mê Kông này. Sông dài rộng và điệu vợi. Chút n



DEMO

Tạo web MVC Laravel



❑ Phần 1

- ❖ Cấu hình kết nối db trong Laravel
- ❖ Giới thiệu thư viện Query Builder
- ❖ Các hàm thường dùng trong query
- ❖ Lấy record đầu tiên
- ❖ Lấy (select) danh sách các record
- ❖ Lấy giá trị trong một field
- ❖ Các hàm tổng kết
- ❖ Query có where, orderby, limit

❑ Phần 2

- ❖ Sử dụng Query có join bảng
- ❖ Kiểm tra record tồn tại
- ❖ Chèn record và lấy id
- ❖ Chèn nhiều record cùng lúc
- ❖ Cập nhật record
- ❖ Xóa record với query
- ❖ Tổng hợp 1: routing, lấy data, nạp view
- ❖ Tổng hợp 2: hiện data theo tham số



- ❑ Kết nối database:
<https://laravel.com/docs/database>
- ❑ Sử dụng Query Builder:
<https://laravel.com/docs/queries>





Cảm ơn