**Bài 4: Route trong Laravel (phần 2)**

Ở bài trước chúng ta đã tìm hiểu về Route::get(),post(),any(),match(),.. rồi, ở phần này chúng ta sẽ tìm hiểu tiếp về Route::resource() và Route::group().

**1,Route::resource().**

-Route::resource() là một chức năng giúp chúng ta Xây dựng RESTful(websevice) một cách nhanh chóng. Đối với những người mới tiếp cận với PHP,Laravel thì chưa cảm thấy được ý nghĩa tuyệt vời của RESTful controller.

**-Cú Pháp:**

Route::resource('url', 'tencontroller', 'tuybien');

Trong đó:

* Url: thì cũng như các route khác là đường dẫn của route.
* tencontroller: là tên của controller (không đi kèm đuôi .php).
* tuybien: là các tùy biến phương thức được sử dụng trong Route, Tham số này có thể bỏ qua nếu không cần thiết.

**-Chú ý:**Để sử dụng được Route::resource() thì các bạn cần phải tạo ra một RESTful Controller. -Để tạo Một RESTful Controller chúng ta có hai các tạo:

* Cách 1: Là tạo bằng tay và viết code cũng bằng tay.
* Cách 2: Dùng lệnh mà laravel cung cấp sẵn cho chúng ta. Các bạn mở commanline lên và trỏ đến thư mục chứa project Laravel (VD của tôi là lar-53) và gõ lệnh.

php artisan make:controller tencontroller --resource

Trong đó tencontroller là tên của các bạn muốn tạo.

-Một RESTful controller sẽ có dạng như sau(trong code demo của tôi tên controller là homeindex):

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

class homeindex extends Controller

{

/\*\*

\* Display a listing of the resource.

\*

\* @return \Illuminate\Http\Response

\*/

public function index()

{

//

}

/\*\*

\* Show the form for creating a new resource.

\*

\* @return \Illuminate\Http\Response

\*/

public function create()

{

//

}

/\*\*

\* Store a newly created resource in storage.

\*

\* @param \Illuminate\Http\Request $request

\* @return \Illuminate\Http\Response

\*/

public function store(Request $request)

{

//

}

/\*\*

\* Display the specified resource.

\*

\* @param int $id

\* @return \Illuminate\Http\Response

\*/

public function show($id)

{

//

}

/\*\*

\* Show the form for editing the specified resource.

\*

\* @param int $id

\* @return \Illuminate\Http\Response

\*/

public function edit($id)

{

//

}

/\*\*

\* Update the specified resource in storage.

\*

\* @param \Illuminate\Http\Request $request

\* @param int $id

\* @return \Illuminate\Http\Response

\*/

public function update(Request $request, $id)

{

//

}

/\*\*

\* Remove the specified resource from storage.

\*

\* @param int $id

\* @return \Illuminate\Http\Response

\*/

public function destroy($id)

{

//

}

}

**-Chú thích các function:**

* index(): Hiển thị một danh sách.
* create(): Thêm mới.
* store(): Lưu mới.
* show($id): Hiển thị một dữ liệu theo tham số truyền vào.
* edit($id): Sửa một dữ liệu theo tham số truyền vào.
* update($id):Cập nhật một dữ liệu theo tham số truyền vào.
* destroy($id):Xóa một dữ liệu theo tham số truyền vào.

-Chú Ý: Điểm mạnh của RESTful Controller là sinh ra cho chúng ta các phương thức luôn có trong một ứng dụng **VD:**Tôi có một route.

Route::resource('photo', 'PhotoController');

-Thì phương thức cũng như kiểu request của nó sẽ như sau.

| **Verb** | **Path** | **Action** | **Route Name** |
| --- | --- | --- | --- |
| GET | /photo | index | photo.index |
| GET | /photo/create | create | photo.create |
| POST | /photo | store | photo.store |
| GET | /photo/{photo} | show | photo.show |
| GET | /photo/{photo}/edit | edit | photo.edit |
| PUT/PATCH | /photo/{photo} | update | photo.update |
| DELETE | /photo/{photo} | destroy | photo.destroy |

-**Trong đó:**{photo} là các tham số id của photo (xem truyền tham số trong route).

**-Tùy biến:**

-Qua phần trên có bạn nào thắc mắc là : "chẵng lẽ RESTful Controller nào cũng phải đáp ứng đủ các action như trên Nếu không đủ thì không dùng được à" ->Vấn đề này các bạn khỏi lo nhé, vì Laravel  đang đứng top 1 về số lượng developer sử dụng nó, nên trong Laravel đã cung cấp cho chúng ta lọc action trong Route::resource(). +Lọc các phương thức được sử dụng trong RESTful :

Route::resource('urlroute', 'homeindex',['only' => ['index', 'create', 'show', 'edit']]);

Chú thích: Ở trên là đoạn code chỉ cho sử dụng các action được khai báo còn lại sẽ không được sử dụng +Lọc các phương thức không được sử dụng trong RESTful:

Route::resource('duongdan', 'homeindex', ['except'=>['index']]);

Chú thích: Cấm không cho sử dụng action index còn lại được sử dụng hết. +Truyền nhiều tham số trong action.

Route::resource('anh.author', 'homeindex');

Khi đó bên action show của controller sẽ có dạng như sau:

public function show($id,$author)

{

// code

}

Để sử dụng các bạn chạy đường dẫn:

anh/{id}/author/{author}

// vd anh/1/author/tai

**2,Route::gorup()**

-Chúng ta muốn xây dựng trang quản trị cho website mà đường dẫn bắt đầu lúc nào cũng phải là admin và phải đăng nhập mới được sử dụng. VD:admin/product/addproduct,admin/category/addcategory,... thì cứ phải gõ từng route rồi check authencation cho từng route sao. -Đối với vấn đề trên Laraver đã cung cấp cho chúng ta một loại Route::group() để giải quyết.

**-Cú pháp:**

Route::group($attr, $handle);

Trong đó:

* $attr: là các mảng thành phần điều kiện. VD: middleware,prefix,namespace,domain.
* $handle: là các đoạn sử lý thường là hàm.

Ví dụ: -Check đăng nhập mới được sử dụng Route bằng middeware( những phần sau mình sẽ nói về middeware).

Route::group(['middleware' => 'auth'], function () {

Route::get('/', function () {

// Uses Auth Middleware

});

Route::get('user/profile', function () {

// Uses Auth Middleware

});

});

-Thêm tiền tố trước các thành phàn trong group.

Route::group(['prefix' => 'admin'], function () {

Route::get('users', function () {

// route nay se co duong dan là admin/users

});

});

-Các thành phần điều kiện khác các bạn có thể xem tại :  https://laravel.com/docs/5.3/routing#route-groups

**3,Lời kết.**

-Qua bài này mình đã trình bày cho các bạn về 2 lại route resource và group . Bài sau mình sẽ nói thêm về cách truyền tham số trong route.