**Bài 14: Collections trong Laravel**

-Bài trước mình đã giới thiệu với mọi người về [Relationship trong Laravel](http://toidicode.com/cac-moi-quan-he-relationships-trong-eloquent-14.html)(Ràng buộc quan hệ) Và mình đã nhận xét là nó khá là tiết kiệm thời gian cho lập trình viên, và tiếp đến ở phần này mình lại tiếp tục giới thiệu đến mọi người một bột thư viện(Collections) trong Laravel hỗ trợ chúng ta Lọc,xử lý dữ liệu rất là nhanh.

**1,Collection Trong Laravel là gì?**

-Collection trong Laravel là một class có tích hợp sẵn các phương thức xử lý dữ liệu thường xuyên nhằm làm giảm thiểu tối đa thời gian cho các lập trình viên. Đặc biệt là làm API kết nối tới database vì dữ liệu trả về từ databasecó sẵn kiểu là Collection.

**2, Các Khai báo, sử dụng Collections.**

-Đầu tiên để khai báo cũng như sử dụng được Collections các bạn cần phải gọi namespace của nó.

use Illuminate\Support\collection;

-Sau khi gọi namespace thành công. các bạn có thể khởi tạo collections bằng 2 cách sau đây:

$collection= collect([]);

// hoac

$collection=conlection::make([]);

-Trong đó: [] là mảng giá trị các bạn truyền vào.

-Lúc này các bạn **prinf\_r($collection)**thì PHP sẽ trả về cho chúng ta một object có chứa giá trị tương ứng với mảng các bạn truyền vào.

**3,Các Collection hay dùng.**

**all**

-Hàm này có tác dụng lấy ra tất cả các giá trị trong collections và trả về dưới dạng mảng.

$data=collection::make([1,2,3,4,5,6])->all();

//output [1,2,3,4,5,6];

**filter**

-Hàm này có tác dụng lọc dữ liệu và trả về các dữ liệu đã được lọc.

$collection = collect(['toidicode.com', 'Vu Thanh Tai', 'PHP'])->filter(function($key,$value)

{

return $value=='t';

})

Output

Array

(

[0] => toidicode.com

)

**expect**

-Hàm này loại bỏ đi các giá trị không cần thiết.

$collection = collect(['product\_id' => 1, 'price' => 100, 'discount' => false]);

$filtered = $collection->except(['product\_id', 'discount']);

$filtered->all();

// output ['price'=>100]

**Count**

-Hàm có tác dụng đếm các phần tử trong collections.

$collection = collect([1, 2, 3, 4]);

$collection->count();

//output 4

**avg**

-Hàm tính giá trị trung bình của các phần tử trong mảng.

$data=collection::make([1,2,3,4,5,6])->avg();

//output 3.5

**sum.**

-hàm tính tổng giá trị của các phần tử.

$collection = collect([1, 2, 3, 4, 5, 6, 7]);

$chunks = $collection->sum();

//output 28

**chumk.**

-Hàm các tác dụng tách mảng ra thành các mảng con.

$collection = collect([1, 2, 3, 4, 5, 6, 7]);

$chunks = $collection->chunk(4);

$chunks->toArray();

//output [[1, 2, 3, 4], [5, 6, 7]]

**first.**

-Hàm có tác dụng trả về giá trị đầu tiên.

collect([1, 2, 3, 4])->first(function ($value, $key) {

return $value > 2;

});

// 3

**get**

-Hàm lấy giá trị của phần tử trong mảng.

$collection = collect(['website' => 'toidicode.com', 'author' => 'Vu Thanh Tai']);

$value = $collection->get('author');

// Vu Thanh Tai

**SortBy.**

-Hàm sắp xếp lại theo thứ tự tăng dần.

$collection = collect([

['name' => 'Desk', 'price' => 200],

['name' => 'Chair', 'price' => 100],

['name' => 'Bookcase', 'price' => 150],

]);

$sorted = $collection->sortBy('price');

$sorted->values()->all();

/\*

[

['name' => 'Chair', 'price' => 100],

['name' => 'Bookcase', 'price' => 150],

['name' => 'Desk', 'price' => 200],

]

\*/

**SortByDesc.**

-Sắp xếp lại giá trị theo thứ tự giảm dần.

$collection = collect([

['name' => 'Desk', 'price' => 200],

['name' => 'Chair', 'price' => 100],

['name' => 'Bookcase', 'price' => 150],

]);

$sorted = $collection->sortByDesc('price');

$sorted->values()->all();

/\*

[

['name' => 'Chair', 'price' => 100],

['name' => 'Bookcase', 'price' => 150],

['name' => 'Desk', 'price' => 200],

]

\*/

**Take.**

-Hàm giới hạn số lượng trả về.

$collection = collect([0, 1, 2, 3, 4, 5]);

$chunk = $collection->take(3);

$chunk->all();

//output [0, 1, 2]

-Hoặc lấy từ cuối lên trên.

$collection = collect([0, 1, 2, 3, 4, 5]);

$chunk = $collection->take(-2);

$chunk->all();

//output [4, 5]

**Các hàm khác.**

-Các hàm ít sử dụng khác các bạn có thể xem thêm tại [đây](https://laravel.com/docs/5.3/collections#available-methods).