

## Bài 11: Tìm Hiểu Library Image Trong Codeigniter

Trong bài này bạn sẽ được học:

- Giới thiệu về Library Image
- Thao tác & làm việc với nó

Trong bài trước, chúng ta đã hoàn thành việc upload hình ảnh trong CI, dựa vào library upload. Chúng ta có thể upload hình ảnh từ máy tính lên web server, cũng như có thể tạo ra những ràng buộc về kích thước hình ảnh, kiểm tra file có hợp lệ hay không, dung lượng tấm hình cho phép là bao nhiêu MB, .... Nhưng như vậy là chưa đủ, khi các bạn bắt tay vào xây dựng hoàn chỉnh một ứng dụng web thì sẽ phát sinh một số vấn đề như làm thế nào để tạo những cái hình nhỏ hơn, thay thế cho hình đại diện trong từng sản phẩm (gọi là các hình **thumbnail**). Để làm được điều này thì chúng ta phải tìm hiểu khái niệm **library image** trong CI.

### 1. Giới thiệu library Image

Với library image thì chúng ta có thể làm được những điều sau:

- **Resize Image**: Từ hình mặc định được upload lên sẽ cắt nhỏ thành hình thumbnail.
- **Image Crop**: Làm việc với thao tác này tức là mình sẽ cắt hình.
- **Image Rotate**: Làm việc với thao tác là xoay hình ảnh.
- **Image Watermark**: Chèn logo hoặc đoạn text bất kì vào tấm hình.

Thực tế thì 2 phương thức mà chúng ta thường sử dụng nhiều nhất chính là **resize & watermark**, đối với việc cắt hoặc xoay hình thì mình nghĩ photoshop sẽ làm tốt việc này hơn so với việc chúng ta dùng PHP để thao tác. Và trong bài này chúng ta sẽ tìm hiểu thao tác resize.

### 2. Cấu hình library image

Đầu tiên thì tất nhiên là sẽ phải gọi library với cú pháp.

```
$this->load->library('image_lib');
```

Và thiết lập một số thông tin cấu hình như sau.

```
$config['image_library'] = 'gd2';
$config['source_image'] = '/path/to/image/mypic.jpg';
$config['create_thumb'] = TRUE;
$config['maintain_ratio'] = TRUE;
$config['width'] = 75;
$config['height'] = 50;
```

Thuộc tính `$config['image_library']` chính là một dạng library image đặc biệt và chúng ta có rất nhiều library giống như thế, ví dụ **GD**, **GD2**, **ImageMagick**, **NetPBM**. Tuy nhiên thông dụng và phổ biến nhất vẫn là GD2, và mặc định thì library sẽ có cấu hình là GD2. Ngoài ra, chúng ta còn phải chỉ ra đường dẫn lưu tấm hình mà chúng ta upload lên.

Sau khi cấu hình đầy đủ các thông tin cần thiết, các bạn phải tiến hành gọi lệnh thực thi thao tác resize hình ảnh gì đó bằng cú pháp sau.

```
$this->image_lib->resize();
```

Và sau đây chúng ta sẽ có một ví dụ nhỏ, giúp các bạn nắm kỹ hơn về thư viện image.

### 3. Thực hành resize với library image

Chúng ta sẽ sử dụng lại controller upload đã làm ở bài trước, vì để có thể resize được thì bắt buộc phải upload được tấm hình lên web server.

```
<?php
class Upload extends CI_Controller{
    public function __construct(){
        parent::__construct();
        $this->load->helper(array("form", "url"));
    }

    public function index(){
        $data['errors'] = '';
        $this->load->view("upload_view", $data);
    }

    public function doupload(){
        if($this->input->post("ok")){
            $config['upload_path'] = './uploads/';
            $config['allowed_types'] = 'gif|jpg|png';
            $config['max_size'] = '900';
            $config['max_width'] = '1024';
            $config['max_height'] = '768';
            $this->load->library("upload", $config);
            if($this->upload->do_upload("img")){
                echo 'Upload Ok';
                $check = $this->upload->data();
                echo "<pre>";
                print_r($check);
                echo "</pre>";
            }else{
                $data['errors'] = $this->upload->display_errors();
                $this->load->view("upload_view", $data);
            }
        }
    }
}
```

Khi đã upload thành công, chúng ta sẽ có một số thao tác như resize hình ảnh. Để làm việc này, đầu tiên chúng ta load library image ra, sau đó tiến hành cấu hình thông tin cho nó. Do chúng ta cần phải có tên gốc của tấm hình mới có thể resize được, nên chúng ta dùng phương thức `$check = ['file_name']` nối chuỗi phía sau đường dẫn tấm hình. Sau đó, chúng ta sẽ cho độ rộng tấm hình khi resize sẽ là 150 và chiều cao của nó là 120, sau khi hoàn tất các thông số trên thì sẽ nạp tất cả thông tin này bằng cú pháp `$this->image_lib->initialize($config)`. Sau đó, chúng ta cần phải gọi lệnh thực thi thao tác resize với cú pháp `$this->image_lib->resize()`. Hàm này sẽ giúp chúng ta cắt hình lớn sang hình nhỏ và thỏa mãn toàn bộ yêu cầu mà chúng ta đã khai báo ở phần thông tin cấu hình.

```
public function doupload(){
    if($this->input->post("ok")){
        $config['upload_path'] = './uploads/';
        $config['allowed_types'] = 'gif|jpg|png';
        $config['max_size'] = '900';
        $config['max_width'] = '1024';
        $config['max_height'] = '768';
        $this->load->library("upload", $config);
        if($this->upload->do_upload("img")){
            echo 'Upload Ok';
            $check = $this->upload->data();
            echo "<pre>";
            print_r($check);
            echo "</pre>";
            $this->load->library("image_lib");
            $config['image_library'] = 'gd2';
            $config['source_image'] =
'./uploads/'.$check['file_name'];
            $config['create_thumb'] = TRUE;
            $config['maintain_ratio'] = TRUE;
            $config['width'] = 150;
            $config['height'] = 120;
            $this->image_lib->initialize($config);
            $this->image_lib->resize();
        }else{
            $data['errors'] = $this->upload->display_errors();
            $this->load->view("upload_view", $data);
        }
    }
}
```

Kiểm tra bằng cách chạy link *localhost/citest/index.php/upload*, sau đó upload bất kì tấm hình nào đó và vào folder uploads xem điều gì sẽ xảy ra nhé. Ví dụ ở đây chúng ta upload tấm hình [lighhou.jpg](#), và lúc kiểm tra tôi thấy ngoài hình gốc là lighthou.jpg thì còn xuất hiện thêm file [lighthou\\_thumb.jpg](#), nó sẽ dựa vào độ rộng hoặc độ cao của tấm hình và xem xét cân đối khi thực hiện thao tác resize.

Lý do tại sao chúng ta phải sử dụng hình thumbnail? Đó là vì nếu một trang web mà chỉ sử dụng tấm hình gốc ban đầu sẽ làm cho website load rất chậm, dung lượng của hình gốc là rất nặng, và chúng ta có hàng chục tấm hình cho sản phẩm, mà sản phẩm nào cũng phải load hình ảnh gốc thì rõ ràng website sẽ load rất chậm. Đó là lý do chúng ta buộc phải sử dụng hình ảnh thumbnail để cải thiện hiệu quả tải website.

### 4. Full code resize image

#### Controller

```
<?php
class Upload extends CI_Controller{
    public function __construct(){
        parent::__construct();
        $this->load->helper(array("form", "url"));
    }

    public function index(){
        $data['errors'] = '';
        $this->load->view("upload_view", $data);
    }

    public function doupload(){
        if($this->input->post("ok")){
            $config['upload_path'] = './uploads/';
            $config['allowed_types'] = 'gif|jpg|png';
            $config['max_size'] = '900';
            $config['max_width'] = '1024';
            $config['max_height'] = '768';
            $this->load->library("upload", $config);
            if($this->upload->do_upload("img")){
                echo 'Upload Ok';
                $check = $this->upload->data();
                echo "<pre>";
                print_r($check);
                echo "</pre>";
                $this->load->library("image_lib");
                $config['image_library'] = 'gd2';
                $config['source_image'] =
                './uploads/'.$check['file_name'];
                $config['create_thumb'] = TRUE;
                $config['maintain_ratio'] = TRUE;
                $config['width'] = 150;
                $config['height'] = 120;
                $this->image_lib->initialize($config);
                $this->image_lib->resize();
            }else{
                $data['errors'] = $this->upload->display_errors();
                $this->load->view("upload_view", $data);
            }
        }
    }
}
```

```
    }  
    }  
}
```

### View

```
<?php  
$upload=array(  
    "name" => "img",  
    "size" => "25",  
);  
?>  
  
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  
<html xmlns="http://www.w3.org/1999/xhtml">  
<head>  
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />  
<title>Freetuts.net</title>  
</head>  
  
<body>  
    <?php  
        if($errors != ""){  
            echo $errors;  
        }  
        echo form_open_multipart(base_url()."index.php/upload/doupload");  
        echo form_label("Avatar: ").form_upload($upload)."<br />";  
        echo form_label(" ").form_submit("ok", "Upload");  
        echo form_close();  
    ?>  
</body>  
</html>
```