

■Carrierwave 單張圖片上傳 桃園Ruby on Rails讀書會 蘇介吾 104/11/04

■目標

做出一個有單張圖片上傳功能的部落格，圖片可以增刪修。

■環境

假設你已經有 Ubuntu Desktop 或 Mac 的Ruby on Rails 開發環境，如果沒有可參考：
<http://afgnsu.logdown.com>

■指定Ruby on Rails版本

```
cd
mkdir projects
cd ~/projects
rvm install 2.2.0
rvm use 2.2.0
rvm gemset create rails420
rvm use 2.2.0@rails420
```

■建立專案

rails _4.2.0_ new lab1 -d mysql <= 指定用 4.2.0版的Rails，並指定用MySQL資料庫

```
cd lab1
```

rvm --create --ruby-version 2.2.0@rails420 <= 會產生.ruby-version和.ruby-gemset

vi Gemfile => 加入

```
=====
gem 'carrierwave'
gem 'mini_magick'
gem 'mysql2', '0.3.20' <= 只加入版本號
gem 'pry'
gem 'pry-rails'
=====
```

bundle => 安裝Gem

vi config/database => 修改 mysql 帳密

rake db:create db:migrate => 新增資料庫（以本例為 lab1_development 和 lab1_test）

■用Scaffold快速建立 blog 資料表（註：Scaffold可快速產生MVC相關程式，後面名稱為單數）

rails generate (簡寫g) scaffold Blog title:string content:text

sudo apt-get install imagemagick <= 安裝相關圖片壓縮套件

rails g uploader Photo => 產生 app/uploaders/photo_uploader.rb

rails g migration add_photo_to_blogs photo:string <= 在blogs資料表中新增photo欄位

rake db:migrate <= 將新增的資料庫規則套用到所有資料表中

■資料庫檢查

```
mysql -u root -p <= 輸入密碼
use lab1_development
show tables; <= 記得要加分號
desc blogs;
quit
```

```
mysql> desc blogs;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
title	varchar(255)	YES		NULL	

content	text	YES		NULL	
created_at	datetime	NO		NULL	
updated_at	datetime	NO		NULL	
photo	varchar(255)	YES		NULL	

6 rows in set (0.01 sec)

■修改模型(Models)

```
vi app/models/blog.rb => 在中間加入
=====
mount_uploader :photo, PhotoUploader
validates :title, :content, presence: true #欄位必填
=====
```

■修改視圖 (Views)

```
vi app/views/blogs/_form.html.erb => 在 submit 之前加入
=====
<div class="field">
  <%= f.label :photo %><br>
  <%= f.file_field :photo %>
  <%= f.hidden_field :photo_cache %>
  <%= image_tag @blog.photo if @blog.photo? %>
  <%= f.check_box :remove_photo %>刪除
</div>
=====

vi app/views/blogs/index.html.erb 表單裡上下各加入
=====
上 <th>Photo</th>
下 <%= image_tag @blog.photo.thumb, style: 'width: 60px; height: 60px;' %>
=====

vi app/views/blogs/show.html.erb 第二行加上 <%= image_tag @blog.photo %>
```

■修改控制器 (Controllers)

```
vi app/controllers/blogs_controller.rb => Strong Parameter 加入三個Hash
=====
def blog_params
  params.require(:blog).permit(:title, :content, :photo, :photo_cache, :remove_photo)
end
=====
```

■修改 Carrierwave設定 (圖片存放路徑、亂數檔名、預設圖片、縮圖、允許圖片副檔名)

```
vi app/uploads/photo_uploaders.rb
=====
# encoding: utf-8
class PhotoUploader < CarrierWave::Uploader::Base
  #指定圖片壓縮程式
  include CarrierWave::MiniMagick

  #儲存方式
  storage :file
  # storage :fog (存到Amazon S3雲端)

  #圖片存放路徑 (範例: ~/projects/lab1/public/uploads/2015/11/xxx.jpg)
  def store_dir
    "uploads/#{model.created_at.year}/#{model.created_at.month}"
  end

  #將檔名更改為雜湊值 (避免上傳檔名重複, 範例: f4445e4a4eca1f46284216dfa954da98.jpg)
  def filename
    if original_filename
      @name ||= Digest::MD5.hexdigest(File.dirname(current_path))
      "#{@name}.#{file.extension}"
    end
  end
end
```

```
#預設圖片（當找不到圖片時就會用這張，檔案路徑： app/assets/images/default.jpg）
def default_url
  'default.jpg'
end

#產生各種縮圖（大、中、小、縮圖）
version :large do
  process :resize_to_fit => [800, 800]
end
version :medium do
  process :resize_to_fit => [400, 400]
end
version :small do
  process :resize_to_fit => [200, 200]
end
version :thumb do
  process :resize_to_fit => [60, 60]
end

#允許上傳圖片副檔名類型
def extension_white_list
  %w(jpg jpeg gif png)
end
end
=====
```

改好後記得在 app/assets/images/ 目錄下新增一張預設圖 default.jpg（最好是方的）

■修改預設路由

```
vi config/routes.rb <= 在resources :blogs 下方加入一行
=====
root "blogs#index"
=====
```

■驗收

```
rails s
打開瀏覽器 http://localhost:3000 => 測試成功！
```