■Carrierwave + Cocoon 多張圖片上傳
==
桃園Ruby on Rails讀書會　蘇介吾 104/11/06

■目標

做出一個可以多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 lab2 -d mysql  <= 指定用 4.2.0版的Rails，並指定用MySQL資料庫

  cd lab2

  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 'cocoon'
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 lab2_development 和 lab2_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
```

■新增 photo 資料表，並且跟 blog 資料表建立關聯 (會產生 blog_id 欄位)

```
  rails g model Photo photo_name photo_file photo_file_cache blog:references
```

■在 blog 資料表中新增 counter_cache (自動統計圖片張數)

```
  rails g migration add_photos_count_to_blogs photos_count:integer

  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 lab2_development
  show tables;  <= 記得要加分號
  desc blogs;
  desc photos;
  quit

mysql> show tables;
+--------------------+
| Tables_in_lab2_d  |
+--------------------+
| blogs              |
| photos             |
| schema_migrations |
+--------------------+
3 rows in set (0.00 sec)

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    |                |
| photos_count | int(11)      | YES  |     | NULL    |                |
+--------------+--------------+------+-----+---------+----------------+
6 rows in set (0.00 sec)

mysql> desc photos;
+-------------------+--------------+------+-----+---------+----------------+
| Field             | Type         | Null | Key | Default | Extra          |
+-------------------+--------------+------+-----+---------+----------------+
| id                | int(11)      | NO   | PRI | NULL    | auto_increment |
| photo_name        | varchar(255) | YES  |     | NULL    |                |
| photo_file        | varchar(255) | YES  |     | NULL    |                |
| photo_file_cache | varchar(255) | YES  |     | NULL    |                |
| blog_id           | int(11)      | YES  | MUL | NULL    |                |
| created_at        | datetime     | NO   |     | NULL    |                |
| updated_at        | datetime     | NO   |     | NULL    |                |
+-------------------+--------------+------+-----+---------+----------------+
7 rows in set (0.00 sec)
```

■修改模型(Models) 一對多關係 (一篇部落格文章，裡面可以有多張圖片)

```
  vi app/models/blog.rb => 在中間加入
================================================
  has_many :photos, dependent: :destroy #如果刪除某篇文章，相關的圖片也會被刪除
  validates :title, :content, presence: true #欄位必填
  accepts_nested_attributes_for :photos, reject_if: :all_blank, allow_destroy:
true
================================================

  vi app/models/photo.rb => 在中間加入
================================================
  belongs_to :blog, counter_cache: true #文章內的圖片總數會自動統計
  validates :filename, 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">
    <table class="blog-photo-form">
      <tbody>
        <%= f.fields_for :photos do |photo| %>
```

```
          <%= render 'photo_fields', f: photo %>
        <% end %>
      </tbody>
    </table>
    <div class="links">
        <%= link_to_add_association "新增圖片", f, :photos %>
    </div>
  </div>
================================================

新增 vi app/views/blogs/_photo_fields.html.erb
================================================
<tr class="nested-fields">
  <td>
    <%= f.label '檔案名稱' %>
    <%= f.text_field :photo_name %><br><br>
    <%= f.file_field :photo_file %>
    <%= f.hidden_field :photo_file_cache, value: f.object.photo_file_cache %>
  </td>
  <td class="thumb">
    <% unless f.object.new_record? %>
    <%# image_tag(f.object.photo_file.url(:small)) %>
    <%= link_to(image_tag(f.object.photo_file.url(:small)),
f.object.photo_file.url) %>
    <% end %>
  </td>
  <td>
    <% unless f.object.new_record? %>
      <%= link_to_remove_association "刪除", f %>
    <% end %>
  </td>
</tr>
================================================

vi app/views/blogs/index.html.erb
================================================
<p id="notice"><%= notice %></p>

<h1>Listing Blogs</h1>

<table border=1 cellpadding=3 cellspacing=0>
  <thead>
    <tr>
      <th>Title</th>
      <th>Content</th>
      <th>Photo</th>
      <th>圖片張數</th>
      <th>操作</th>
    </tr>
  </thead>

  <tbody>
    <% @blogs.each do |blog| %>
      <tr>
        <td><%= blog.title %></td>
        <td><%= truncate(blog.content, length: 30, ommision: '...') %></td>
        <td align=center><%= blog.photos.present? ? image_tag
(blog.photos.first.photo_file.thumb.url) : image_tag('/default.jpg', style:
'height: 60px; width: 60xp;') %></td>
        <td align=center><%= blog.photos_count.present? ? "#{blog.photos_count}
張" : "無" %></td>
        <td><%= link_to 'Show', blog %>
        <%= link_to 'Edit', edit_blog_path(blog) %>
        <%= link_to 'Destroy', blog, method: :delete, data: { confirm: 'Are you
sure?' } %></td>
      </tr>
    <% end %>
  </tbody>
</table>

<br>
```

```
<%= link_to 'New Blog', new_blog_path %>
================================================
```

vi app/views/blogs/show.html.erb 加入

```
================================================
<p>
  <strong>Photos:</strong>

  <% unless @blog.photos.blank? %>
  <ul>
    <% @blog.photos.each do |p| %>
    <li><%= p.photo_name %><br>
      <%# image_tag p.photo_file.medium.url %></li>
      <%= link_to(image_tag(p.photo_file.url(:small)), p.photo_file.url) %>
    <% end %>
  </ul>
  <% else %>
    <img src="/default.jpg">
  <% end %>
</p>
================================================
```

■修改控制器（Controllers)

vi app/controllers/blogs_controller.rb => Strong Parameter 加入四個Hash

```
================================================
def blog_params
  params.require(:blog).permit(:title, :content, photos_attributes:
[:id, :photo_name, :photo_file, :photo_file_cache, :_destroy])
end
================================================
```

■修改資源（Assets）

修改 app/assets/javascripts/blogs.coffee 加入

```
==============================================
jQuery ($) ->
$(document).on('page:change', ()->

  onAddFile = (event) ->
    file = event.target.files[0]
    url = URL.createObjectURL(file)

    photo_fileContainer = $(this).parent().siblings('div.photo_file')

    if photo_fileContainer.find('img').length == 0
      photo_fileContainer.append('<img src="' + url + '" />')
    else
      photo_fileContainer.find('img').attr('src', url)

  # for redisplayed file inputs and file inputs in edit page
  $('input[type=file]').each(()->
      $(this).change(onAddFile)
  )

  $('#photos').bind('cocoon:before-insert', (e,task_to_be_added) ->
      console.log(task_to_be_added)
      task_to_be_added.fadeIn('slow')
  )

  $('#photos').bind('cocoon:before-remove', (e, task) ->
      $(this).data('remove-timeout', 1000)
      task.fadeOut('slow')
  )

  # register event handler when new cocoon partial is inserted from
link_to_add_association link
```

```
  $('body').on('cocoon:after-insert', (e, addedPartial) ->
      $('input[type=file]', addedPartial).change(onAddFile)
  )
)
==============================================

vi app/assets/javascripts/application.js
==============================================
//= require cocoon
//= require blogs
//= require_tree .  #在這行之上加入2行
==============================================
```

■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/lab2/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

  #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/ 和 public/ 目錄下各新增一張預設圖 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！

■參考資料

https://github.com/nathanvda/cocoon
https://github.com/wstevo4891/rails_cocoon_demo/

■範例下載

https://github.com/afgnsu/cocoon