

# WEB+DBシステム(応用編)



第8回(2016年11月17日)

ショッピング・サイトの制作(2/3)

# Deviseを用いたテストを組み込む

---

ログイン認証をdeviseに任せている。

Deviseを使ったテストとして、別に記述が必要になる。

Macroとして、login\_userメソッドを用意する。

という部分が、うまく動作していません。

(先送りします。)

# 画像のアップロード

---

画像のアップロードに関する設計変更

(1) 1対1で、画像1枚を直接アップロードする。

merchandiseに、フィールド`photo:binary`を追加

(2) 2枚目以後の画像は、当面考えない。

## この設計にした理由(1/2)

---

AttachmentからMerchandiseへリレーションを張ると、最初の1枚の画像を登録する時Merchandise(野菜)のレコードが生成されていないと、リレーションを張るためのIDが取得できない。

(JavaScriptなどを、この講義で扱っていない。)

先に画像をアップロードし「仮」のIDを発効して、一次的なIDに画像を寄せて、本体のアップロード後に画像を張るようにすると、途中キャンセルでの制御の論理が複雑になって、バグの原因になりやすいと思われた。

## この設計にした理由(2/2)

---

WEB上のfile\_fieldからcontrollerへ上がって来るパラメータのクラスを見ると、attachmentのような、直接的にレコードのパラメータになっていないフィールドのfileデータは"String"型になっていて、「ファイル名」だけしか拾えない！

実は、昨年度この部分で何回か設計変更をしました。

# Photoフィールドの追加

---

rails generate migration

AddPhotoToMerchandise photo:binary  
file\_name:string file\_type:string

で画像用のフィールドを追加する。

rake db:migrate

で、データベースに反映する。

```
[root@cisnote vegetable-market]# rails g migration AddPhotoToMerchandise photo:binary file_name:string file_type:string
  invoke  active_record
  create  db/migrate/20151110082657_add_photo_to_merchandise.rb
[root@cisnote vegetable-market]# rake db:migrate
== 20151110082657 AddPhotoToMerchandise: migrating =====
-- add_column(:merchandises, :photo, :binary)
   -> 0.0021s
-- add_column(:merchandises, :file_name, :string)
   -> 0.0003s
-- add_column(:merchandises, :file_type, :string)
   -> 0.0003s
== 20151110082657 AddPhotoToMerchandise: migrated (0.0029s) =====
```

# 画像追加用のmigration

---

```
class AddPhotoToMerchandise < ActiveRecord::Migration
  def change
    add_column :merchandises, :photo, :binary
    add_column :merchandises, :file_name, :string
    add_column :merchandises, :file_type, :string
  end
end
```

```
1 class AddPhotoToMerchandise < ActiveRecord::Migration
2   def change
3     add_column :merchandises, :photo, :binary
4     add_column :merchandises, :file_name, :string
5     add_column :merchandises, :file_type, :string
6   end
7 end
8
```

# 画像データの転送メソッド

---

画像を表示するため、app/controllers/merchandise\_controllers.に photo メソッドを追加する。

Showでの表示(rendering)で、このメソッドが呼ばれる。

```
def photo
  @merchandise = Merchandise.find(params[:id])
  send_data @merchandise.photo,
    :filename =>
      @merchandise.file_name, :type=>@merchandise.file_type
end
```

# ブラウザからのパラメータ

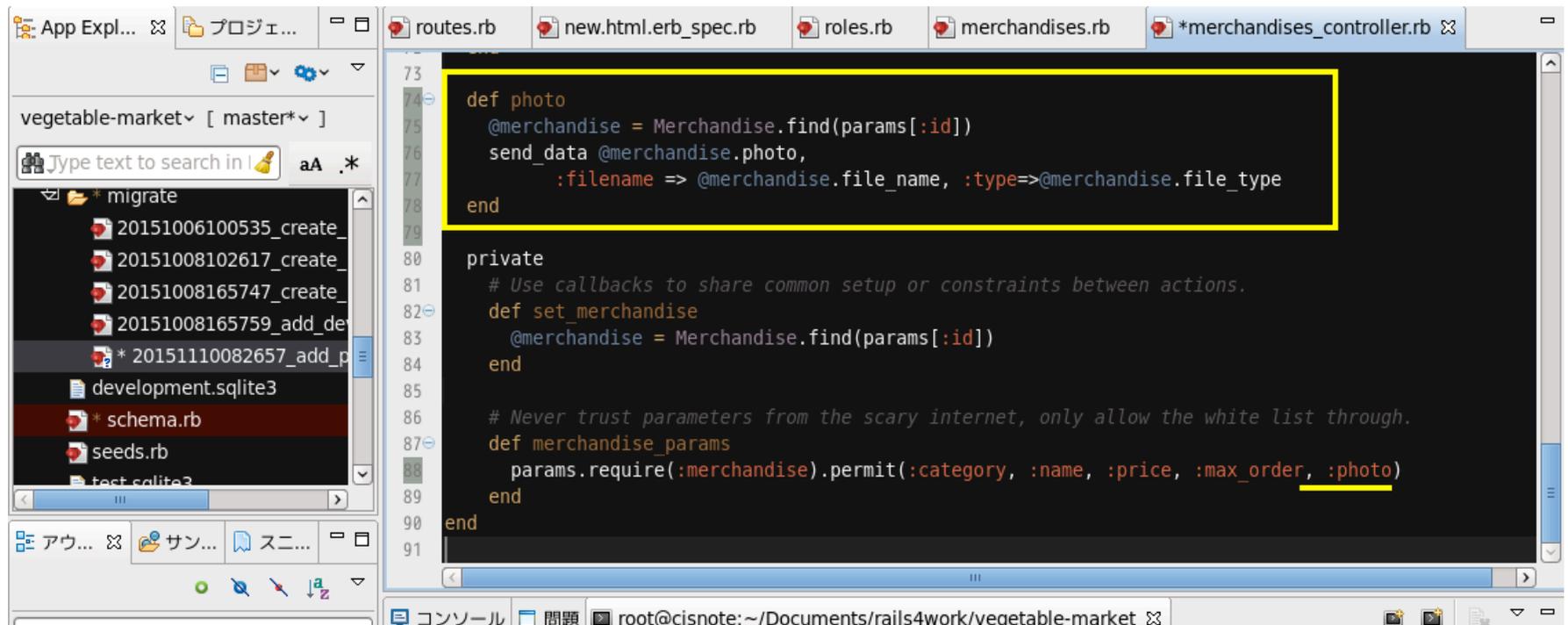
---

ブラウザからデータを受け取るホワイトリストに、  
`:photo`を追加する。

Rails 4からホワイトリストにパラメータを明示すること  
となった。

```
# Never trust parameters from the scary internet, only allow the  
white list through.  
def merchandise_params  
  params.require(:merchandise).permit(:category, :name,  
    :price, :max_order, :photo)  
end
```

# Merchandise\_controller.rb



```
73  
74 def photo  
75   @merchandise = Merchandise.find(params[:id])  
76   send_data @merchandise.photo,  
77             :filename => @merchandise.file_name, :type=>@merchandise.file_type  
78 end  
79  
80 private  
81 # Use callbacks to share common setup or constraints between actions.  
82 def set_merchandise  
83   @merchandise = Merchandise.find(params[:id])  
84 end  
85  
86 # Never trust parameters from the scary internet, only allow the white list through.  
87 def merchandise_params  
88   params.require(:merchandise).permit(:category, :name, :price, :max_order, :photo)  
89 end  
90  
91
```

# Createメソッドの修正

```
# POST /merchandises
# POST /merchandises.json
def create
  if params[:merchandise][:photo]
    @file = params[:merchandise][:photo]
    @stat = @file.tempfile.stat
    @merchandise = Merchandise.new(
      :category => params[:merchandise][:category],
      :name => params[:merchandise][:name],
      :price => params[:merchandise][:price],
      :max_order => params[:merchandise][:max_order],
      :file_name => @file.original_filename,
      :file_type => @file.content_type,
      :photo => @file.read
    )
    respond_to do |format|
      if @merchandise.save
        format.html { redirect_to @merchandise, notice: 'Merchandise was successfully created.' }
        format.json { render action: 'show', status: :created, location: @merchandise }
      else
        format.html { render action: 'new' }
        format.json { render json: @merchandise.errors, status: :unprocessable_entity }
      end
    end
  else
    respond_to do |format|
      format.html { render action: 'new', notice: 'Photo is not attached.' }
      format.json { render json: @merchandise.errors, status: :unprocessable_entity }
    end
  end
end
```

元々定義されていた部分の前後に、かぶせるように追加。

引用符号に注意してコピーして下さい。

# Merchandises\_controller # create

---

```
35 # POST /merchandises.json
36 def create
37   if params[:merchandise][:photo]
38     @file = params[:merchandise][:photo]
39     @stat = @file.tempfile.stat
40     @merchandise = Merchandise.new(
41       :category => params[:merchandise][:category],
42       :name => params[:merchandise][:name],
43       :price => params[:merchandise][:price],
44       :max_order => params[:merchandise][:max_order],
45       :file_name => @file.original_filename,
46       :file_type => @file.content_type,
47       :photo => @file.read
48     )
49     # @merchandise = Merchandise.new(merchandise_params)
50     respond_to do |format|
51       if @merchandise.save
52         format.html { redirect_to @merchandise, notice: 'Merchandise was successfully created.' }
53         format.json { render :show, status: :created, location: @merchandise }
54       else
55         format.html { render :new }
56         format.json { render json: @merchandise.errors, status: :unprocessable_entity }
57       end
58     end
59   else
60     respond_to do |format|
61       format.html { render action: 'new', notice: 'Photo is not attached.' }
62       format.json { render json: @merchandise.errors, status: :unprocessable_entity }
63     end
64   end
65 end
```

# 商品登録画面での画像受付け

---

コントローラでの追加には、create / updateの両方の修正が必要となるが、画面は app/views/merchandise/\_form.html.erbだけの修正で可能となる。

以下を追加する。

```
<div class="field">  
  <%= f.label :photo %><br>  
  <%= f.file_field :photo %>  
</div>
```

# views/merchandises/\_form.html.erb

```
7     <%= @merchandise.errors.full_messages.each do |msg| %>
8       <li><%= msg %></li>
9     <%= end %>
10    </ul>
11  </div>
12  <%= end %>
13
14  <div class="field">
15    <%= f.label :category %><br>
16    <%= f.text_field :category %>
17  </div>
18  <div class="field">
19    <%= f.label :name %><br>
20    <%= f.text_field :name %>
21  </div>
22  <div class="field">
23    <%= f.label :price %><br>
24    <%= f.number_field :price %>
25  </div>
26  <div class="field">
27    <%= f.label :max_order %><br>
28    <%= f.number_field :max_order %>
29  </div>
30  <div class="field">
31    <%= f.label :photo %><br>
32    <%= f.file_field :photo %>
33  </div>
34  <div class="actions">
35    <%= f.submit %>
36  </div>
37  <%= end %>
38
```

# Index画面での画像表示

---

views/merchandise/index.html.erbで、一覧画面で商品一覧と一緒に画像が表示できるようにする。

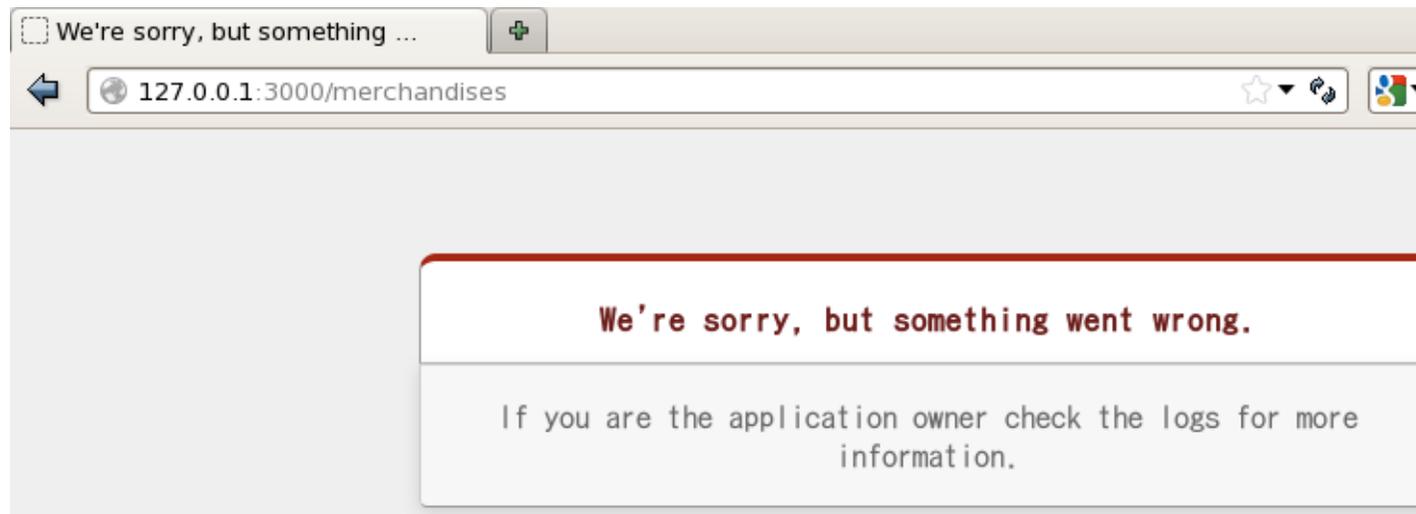
# この後のスライドの作業で注意!

---

Rails 4から、この画面がよく出るようになった。

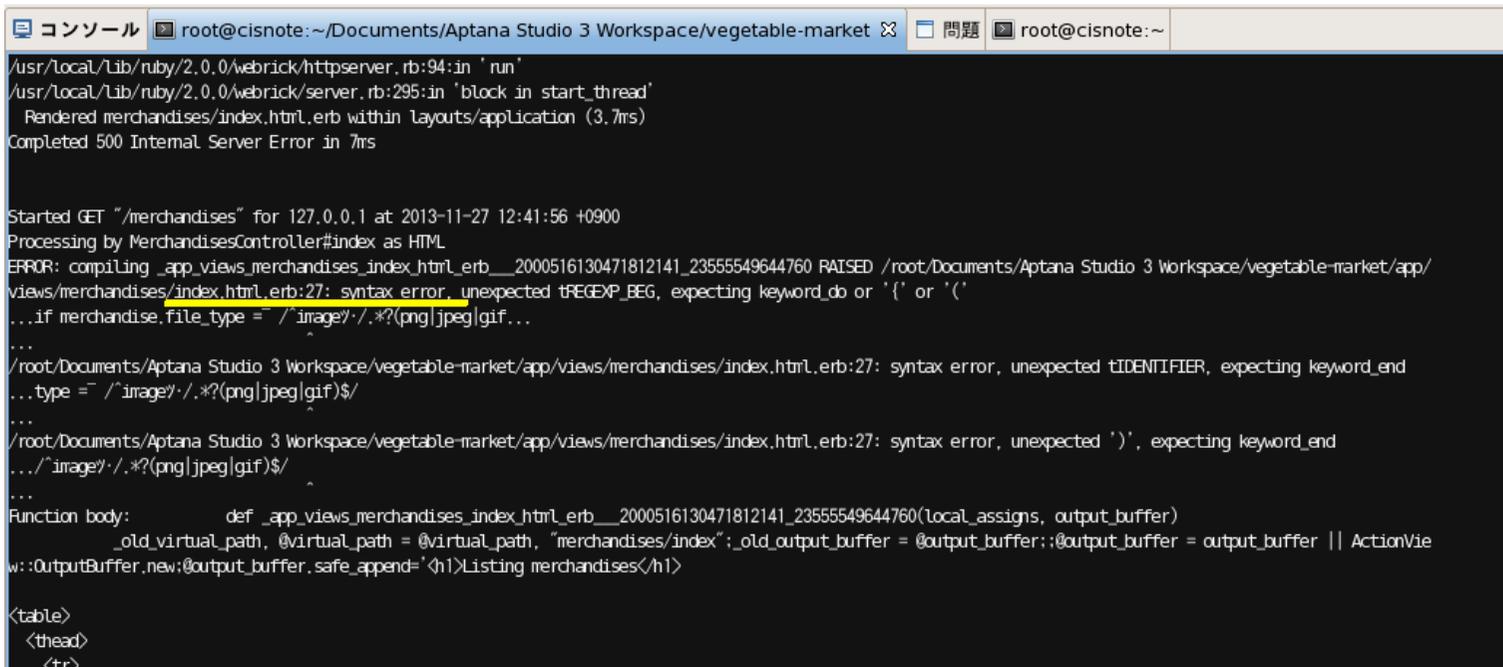
「どうしたらいいのでしょうか!?!」

こんな時は、コンソールを見る。



# (続き)エラーの対処方法

サーバが走っているコンソールを、良く見て下さい。  
どのファイルの何行目がおかしいか、だいたい表示されています。(これだけで対処できない場合は、教師またはTAへ)  
コピーの文字化けで、よくこれが出るようです。



```
コンソール root@cisnote:~/Documents/Aptana Studio 3 Workspace/vegetable-market 問題 root@cisnote:~
/usr/local/lib/ruby/2.0.0/webrick/httpserver.rb:94:in 'run'
/usr/local/lib/ruby/2.0.0/webrick/server.rb:295:in 'block in start_thread'
  Rendered merchandises/index.html.erb within layouts/application (3.7ms)
Completed 500 Internal Server Error in 7ms

Started GET "/merchandises" for 127.0.0.1 at 2013-11-27 12:41:56 +0900
Processing by MerchandisesController#index as HTML
ERROR: compiling _app_views_merchandises_index_html_erb__2000516130471812141_23555549644760 RAISED /root/Documents/Aptana Studio 3 Workspace/vegetable-market/app/
views/merchandises/index.html.erb:27: syntax error, unexpected tREGEXP_BEG, expecting keyword_do or '{' or '('
...if merchandise.file_type =~ /^image\/.*?(png|jpeg|gif)...
...
/root/Documents/Aptana Studio 3 Workspace/vegetable-market/app/views/merchandises/index.html.erb:27: syntax error, unexpected tIDENTIFIER, expecting keyword_end
...type =~ /^image\/.*?(png|jpeg|gif)$/
...
/root/Documents/Aptana Studio 3 Workspace/vegetable-market/app/views/merchandises/index.html.erb:27: syntax error, unexpected ')', expecting keyword_end
.../^image\/.*?(png|jpeg|gif)$/
...
Function body:      def _app_views_merchandises_index_html_erb__2000516130471812141_23555549644760(local_assigns, output_buffer)
                    old_virtual_path, @virtual_path = @virtual_path, "merchandises/index"; old_output_buffer = @output_buffer; @output_buffer = output_buffer || ActionVie
w::OutputBuffer.new(@output_buffer, safe_append='<h1>Listing merchandises</h1>

<table>
  <thead>
    <tr>
```

# views/merchandises/index.html.erb

---

```
<tbody>
  <% @merchandises.each do |merchandise| %>
    <tr>
      <td><%= merchandise.category %></td>
      <td><%= merchandise.name %></td>
      <td><%= merchandise.price %></td>
      <td><%= merchandise.max_order %></td>
      <td>
        <% if merchandise.photo %>
          <p>
            <% if merchandise.file_type =~ /^image\/.*?(png|jpeg|gif)$/ %>
              <%= image_tag url_for({:action => 'photo', :id=> merchandise.id,
                :filename => merchandise.file_name}),
                :alt => merchandise.file_name %>
            <% end %>
          <br />
          </p>
        <% end %>
      </td>
      <td><%= link_to 'Show', merchandise %></td>
      <td><%= link_to 'Edit', edit_merchandise_path(merchandise) %></td>
      <td><%= link_to 'Destroy', merchandise, method: :delete, data: { confirm: 'Are
you sure?' } %></td>
    </tr>
  </tbody>
```

この黒い  
部分を追加

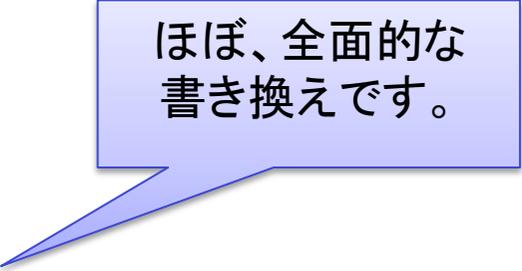
# Views/merchandises/index.html.erb

```
15 </thead>
16
17 <tbody>
18   <%= @merchandises.each do |merchandise| %>
19     <tr>
20       <td><%= merchandise.category %></td>
21       <td><%= merchandise.name %></td>
22       <td><%= merchandise.price %></td>
23       <td><%= merchandise.max_order %></td>
24     <td>
25       <%= if merchandise.photo %>
26         <p>
27           <%= if merchandise.file_type =~ /^image\/.*?(png|jpeg|gif)$/ %>
28             <%= image_tag url_for({:action => 'photo', :id=> merchandise.id,
29               :filename => merchandise.file_name}),
30               :alt => merchandise.file_name %>
31           <%= end %>
32         <br />
33       </p>
34     <%= end %>
35   </td>
36   <td><%= link_to 'Show', merchandise %></td>
37   <td><%= link_to 'Edit', edit_merchandise_path(merchandise) %></td>
38   <td><%= link_to 'Destroy', merchandise, method: :delete, data: { confirm: 'Are you sure?' } %></td>
39 </tr>
40 <%= end %>
41 </tbody>
42 </table>
43
44
```

# merchandises\_ontroller # update

---

```
# PATCH/PUT /merchandises/1
# PATCH/PUT /merchandises/1.json
def update
  if params[:merchandise][:photo]
    @file = params[:merchandise][:photo]
    @stat = @file.tempfile.stat
    respond_to do |format|
      if @merchandise.update_attributes(
        :category => params[:merchandise][:category],
        :name => params[:merchandise][:name],
        :price => params[:merchandise][:price],
        :max_order => params[:merchandise][:max_order],
        :file_name => @file.original_filename,
        :file_type => @file.content_type,
        :photo => @file.read
      )
        format.html { redirect_to @merchandise, notice: 'Merchandise was successfully updated.' }
        format.json { head :no_content }
      else
        format.html { render action: 'edit' }
        format.json { render json: @merchandise.errors, status: :unprocessable_entity }
      end
    end
  else
    respond_to do |format|
      format.html { render action: 'edit', notice: 'Photo is not attached.' }
      format.json { render json: @merchandise.errors, status: :unprocessable_entity }
    end
  end
end
```



ほぼ、全面的な  
書き換えです。

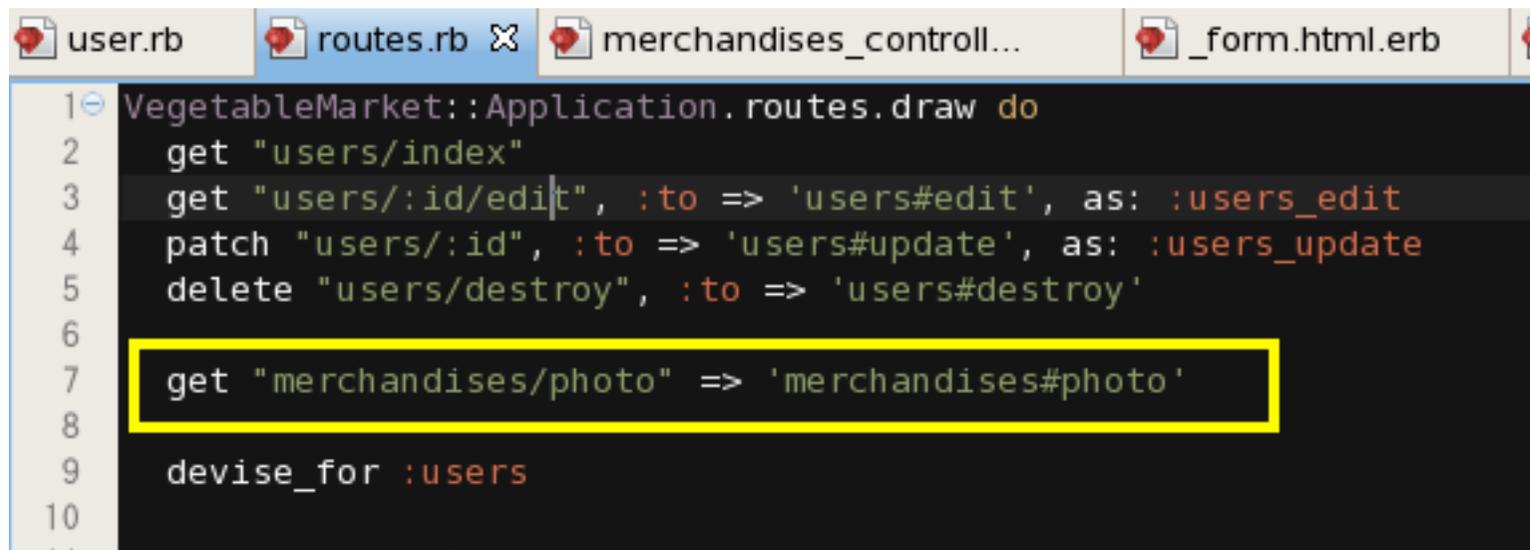
# merchandises\_controller # update

```
user.rb routes.rb merchandises_controller.rb ✕ _form.html.erb index.html.erb
58 # PATCH/PUT /merchandises/1
59 # PATCH/PUT /merchandises/1.json
60 def update
61   if params[:merchandise][:photo]
62     @file = params[:merchandise][:photo]
63     @stat = @file.tempfile.stat
64     respond_to do |format|
65       if @merchandise.update_attributes(
66         :category => params[:merchandise][:category],
67         :name => params[:merchandise][:name],
68         :price => params[:merchandise][:price],
69         :max_order => params[:merchandise][:max_order],
70         :file_name => @file.original_filename,
71         :file_type => @file.content_type,
72         :photo => @file.read
73       )
74         format.html { redirect_to @merchandise, notice: 'Merchandise was successfully updated.' }
75         format.json { head :no_content }
76       else
77         format.html { render action: 'edit' }
78         format.json { render json: @merchandise.errors, status: :unprocessable_entity }
79       end
80     end
81   else
82     respond_to do |format|
83       format.html { render action: 'edit', notice: 'Photo is not attached.' }
84       format.json { render json: @merchandise.errors, status: :unprocessable_entity }
85     end
86   end
87 end
88
```

# config/routes.rb

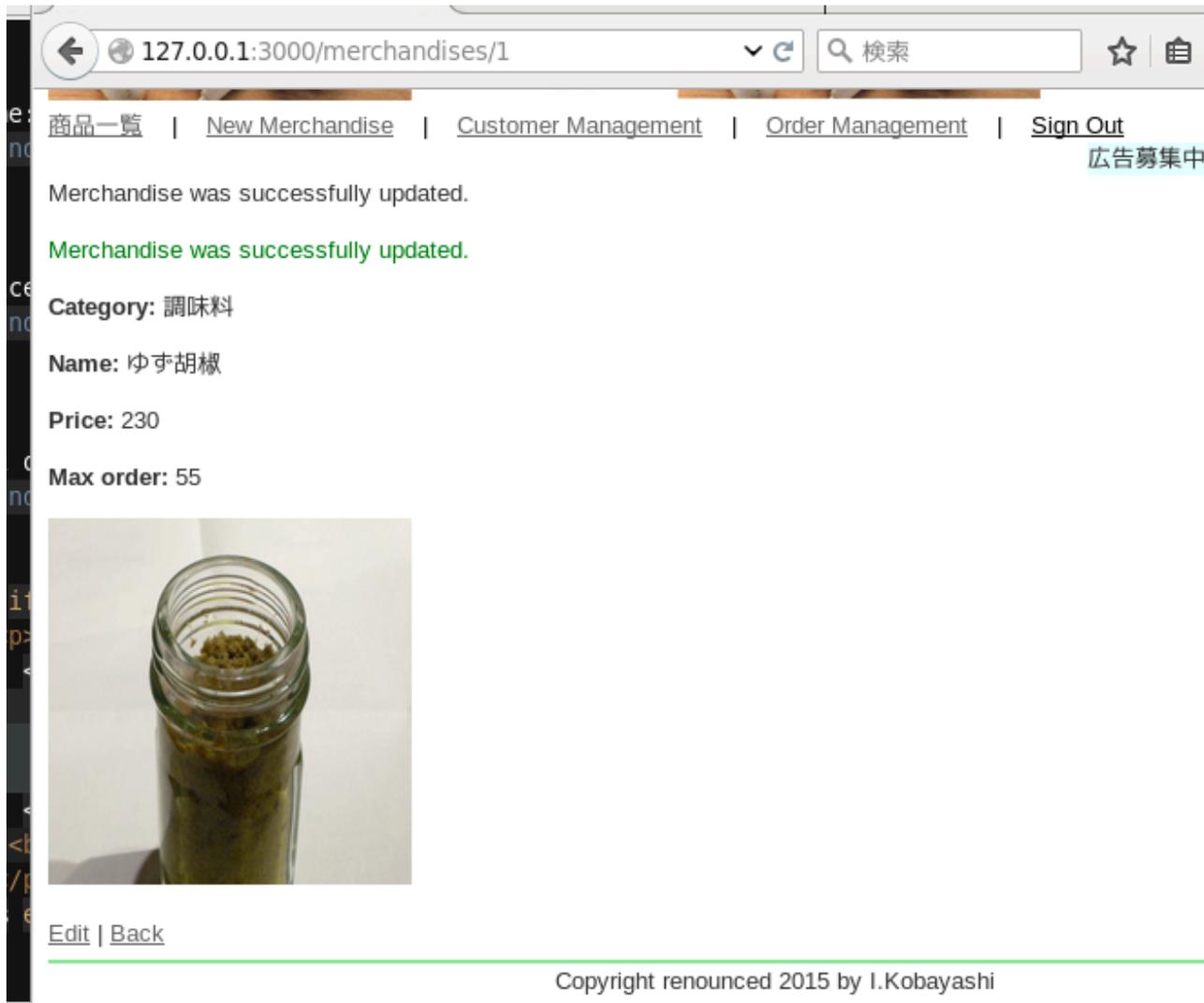
---

Routes.rbに、  
get `merchandises/photo` => `merchandises#photo`  
を追加する。



```
user.rb routes.rb x merchandises_controll... _form.html.erb
1 VegetableMarket::Application.routes.draw do
2   get "users/index"
3   get "users/:id/edit", :to => 'users#edit', as: :users_edit
4   patch "users/:id", :to => 'users#update', as: :users_update
5   delete "users/destroy", :to => 'users#destroy'
6
7   get "merchandises/photo" => 'merchandises#photo'
8
9   devise_for :users
10
11
```

# 写真をアップロードした画面



127.0.0.1:3000/merchandises/1

商品一覧 | [New Merchandise](#) | [Customer Management](#) | [Order Management](#) | [Sign Out](#)

広告募集中

Merchandise was successfully updated.

Merchandise was successfully updated.

Category: 調味料

Name: ゆず胡椒

Price: 230

Max order: 55



[Edit](#) | [Back](#)

Copyright renounced 2015 by I.Kobayashi

# 画像の大きさ

---

画像の大きさを小さくしてみました。

image\_tagのパラメータを追加

また、テーブル表記に罫線を追加しました。

ネットでググると一発で答えが出ますので、皆さんは各自の工夫で追加してみてください。

# 今日の最終的な画面

VegetableMarket

127.0.0.1:3000/merchandises



小金井  
野菜むら

List of merchandises | [New Merchandise](#) | Customer Management #(not ready) | Order Management #(not ready) | [Sign Out](#)

## Listing merchandises

Category	Name	Price	Max order	Photo			
加工品	ゆず唐辛子	200	15		Show	Edit	Destroy
生鮮食料品	きゅうり	10	100		Show	Edit	Destroy

[New Merchandise](#)  
[Sign out](#)

### Koganei Vegetable Market

[Sign Out](#)  
この下は、ROLEによって分岐すべき・・・  
[See vegetables](#)  
[Shopping Cart](#)  
[Purchase](#)  
[Trace](#)  
[About Us](#)

# \_right\_bar.html.erb

---

```
<h3>Koganei Vegetable Market</h3>
```

```
<% if current_user %>
```

```
  <%= link_to "Sign Out",  
    destroy_user_session_path, :method=>"delete" %> <br />
```

```
<% else %>
```

```
  <%= link_to "Sign In", new_user_session_path %> <br />
```

```
  <%= link_to "Sign Up", new_user_registration_path %> <br />
```

```
<% end %>
```

```
この下は、ROLEによって分岐すべき・・・<br />
```

```
  <%= link_to "See vegetables", merchandises_path %> <br />
```

```
Shopping Cart<br />
```

```
Purchase<br />
```

```
Trace<br />
```

```
About Us<br />
```

# ‘Add to Cart’ボタン

最終形からの、コードです。  
画像の表示も行いました。

```
<% @merchandises.each do |merchandise| %>
  <tr>
    <td><%= merchandise.category %></td>
    <td><%= merchandise.name %></td>
    <td><%= merchandise.price %></td>
    <td><%= merchandise.max_order %></td>
    <td>
      <% if merchandise.photo %>
        <p>
          <% if merchandise.file_type =~ /^image¥/.?*(png|jpeg|gif)$/ %>
            <%= image_tag url_for({:action => 'photo', :id=> merchandise.id,
              :filename => merchandise.file_name}), :size=>"50x50",
              :alt => merchandise.file_name %>
          <% end %>
        <br />
        </p>
      <% end %>
    </td>
    <% if current_user %>
      <% if current_user.role_id == 2 %>
        <td><%= link_to 'Show', merchandise %></td>
        <td><%= link_to 'Edit', edit_merchandise_path(merchandise) %></td>
        <td><%= link_to 'Destroy', merchandise, method: :delete, data: { confirm: 'Are you sure?' } %></td>
      <% elsif current_user.role_id == 3 %>
        <td>
          <%= form_tag "/carts/create/"+merchandise.id.to_s, :method => 'POST' do %>
            Amount:
            <%= tag :input, { :type => 'text', :name => 'how_many', :size=>"6" } %>
            <%= submit_tag 'Add to Cart', :name => 'Purchase' %>
          <% end %>
        </td>
      <% end %>
    <% end %>
  </tr>
<% end %>
```

# Add to Cart ボタン

```
orders_controller.rb  _form.html.erb  index.html.erb x
22     <td><%= merchandise.category %></td>
23     <td><%= merchandise.name %></td>
24     <td><%= merchandise.price %></td>
25     <td><%= merchandise.max_order %></td>
26     <td>
27       <%= if merchandise.photo %>
28         <p>
29           <%= if merchandise.file_type =~ /^image\/.*?(png|jpeg|gif)$/ %>
30             <%= image_tag url_for({:action => 'photo', :id=> merchandise.id,
31               :filename => merchandise.file_name}), :size=>"50x50",
32               :alt => merchandise.file_name %>
33           <%= end %>
34         <br />
35       </p>
36     <%= end %>
37   </td>
38   <%= if current_user %>
39     <%= if current_user.role_id == 2 %>
40       <td><%= link_to 'Show', merchandise %></td>
41       <td><%= link_to 'Edit', edit_merchandise_path(merchandise) %></td>
42       <td><%= link_to 'Destroy', merchandise, method: :delete, data: { confirm: 'Are you sure?' } %></td>
43     <%= elsif current_user.role_id == 3 %>
44     <td>
45       <%= form_tag "/carts/create/"+merchandise.id.to_s, :method => 'POST' do %>
46         Amount:
47         <%= tag :input, { :type => 'text', :name => 'how_many', :size=>"6" } %>
48         <%= submit_tag 'Add to Cart', :name => 'Purchase' %>
49       <%= end %>
50     </td>
51   <%= end %>
52 <%= end %>
53 </tr>
54 <%= end %>
```

# 画面の切り替え

---

Current\_user.role\_idの値によって、表示を切り替えている  
点に注目して下さい。

```
<% if current_user %>
  <% if current_user.role_id == 2 %>
    <td><%= link_to 'Show', merchandise %></td>
    <td><%= link_to 'Edit', edit_merchandise_path(merchandise) %></td>
    <td><%= link_to 'Destroy', merchandise, method: :delete, data: { confirm: 'Are you sure?' }
%></td>
  <% elsif current_user.role_id == 3 %>
    <td>
      <%= form_tag "/carts/create/"+merchandise.id.to_s, :method => 'POST' do %>
        Amount:
        <%= tag :input, { :type => 'text', :name => 'how_many', :size=>"6" } %>
        <%= submit_tag 'Add to Cart', :name =>'Purchase' %>
      <% end %>
    </td>
  <% end %>
<% end %>
```

# 画面の切り替え例

```
11 <% if current_user && current_user.role_id == 2 %>
12   <th></th>
13   <th></th>
14   <th></th>
15   <% end %>
16 </tr>
17 </thead>
18
19 <tbody>
20 <% @merchandises.each do |merchandise| %>
21 <tr>
22   <td><%= merchandise.category %></td>
23   <td><%= merchandise.name %></td>
24   <td><%= merchandise.price %></td>
25   <td><%= merchandise.max_order %></td>
26   <td>
27     <% if merchandise.photo %>
28     <p>
29       <% if merchandise.file_type =~ /^image\/.*?(png|jpeg|gif)$/ %>
30         <%= image_tag url_for({:action => 'photo', :id=> merchandise.id,
31                               :filename => merchandise.file_name}), :size=>"50x50",
32                               :alt => merchandise.file_name %>
33       <% end %>
34     <br />
35     </p>
36     <% end %>
37   </td>
38   <% if current_user %>
39     <% if current_user.role_id == 2 %>
40     <td><%= link_to 'Show', merchandise %></td>
41     <td><%= link_to 'Edit', edit_merchandise_path(merchandise) %></td>
42     <td><%= link_to 'Destroy', merchandise, method: :delete, data: { confirm: 'Are you sure?' } %></td>
43     <% elsif current user.role id == 3 %>
```

# 一般ゲストの画面

127.0.0.1:3000/merchandises

野菜むら

Login: fugafuga@hosei.ac.jp

List of merchandises | [Shopping Cart](#) | [Orders](#) | About Us #(not ready) | [Sign Out](#)

translation missing: ja.devise.sessions.user.signed\_in

## Listing merchandises

Category	Name	Price	Max order	Photo	Amount:	Add to Cart
加工品	ゆず唐辛子	200	15		<input type="text"/>	<input type="button" value="Add to Cart"/>
生鮮食料品	きゅうり	10	100		<input type="text"/>	<input type="button" value="Add to Cart"/>

Copyright (C) 2013, "renounced" by Ikuo Kobayashi  
Contact us at hoge@fuga.fugafuga.org

Koganei Vegetable Market

[Sign Out](#)  
[See shopping cart](#)  
[See order statuses](#)  
[See vegetables](#)  
[About us](#)

# ショップ店員モードの画面

127.0.0.1:3000/merchandises

Google

小金井  
野菜むら

List of merchandises | [New Merchandise](#) | [Customer Management](#) | [Order Management](#) | [Sign Out](#)

translation missing: ja.devise.sessions.user.signed\_in

**Listing merchandises**

Category	Name	Price	Max order	Photo			
加工品	ゆず唐辛子	200	15		Show	Edit	Destroy
生鮮食料品	きゅうり	10	100		Show	Edit	Destroy

Go to Cashier

Login: mogamoga@hosei.ac.jp

Koganei Vegetable Market

[Sign Out](#)  
[See vegetables](#)  
[About us](#)

# カート処理の準備ができていないので・・・

以下のオプションを追加します。

`:disabled => true`

```
36     <% end %>
37   </td>
38   <% if current_user %>
39     <% if current_user.role_id == 2 %>
40       <td><%= link_to 'Show', merchandise %></td>
41       <td><%= link_to 'Edit', edit_merchandise_path(merchandise) %></td>
42       <td><%= link_to 'Destroy', merchandise, method: :delete, data: { confirm: 'Are you su
43     <% elsif current_user.role_id == 3 %>
44     <td>
45       <%= form_tag "/carts/create/"+merchandise.id.to_s, :method => 'POST' do %>
46         Amount:
47         <%= tag :input, { :type => 'text', :name => 'how_many', :size=>"6" } %>
48         <%= submit_tag 'Add to Cart', :name => 'Purchase', :disabled => true %>
49       <% end %>
50     </td>
51   <% end %>
52 <% end %>
53 </tr>
```

# カートの設計

---

## Cart Class

belongs\_to :user

(user has\_many carts)

belongs\_to :merchandise

(merchandise has\_many carts)

properties:

user\_id

merchandise\_id

number of items

# カートの生成

---

ユーザが商品一覧画面で「Add to Cart」ボタンをクリックした時に、新たにショッピングカートを作成します。個々の記録では、「購入数」も一緒に処理します。

カートの一覧は、「全てのユーザの購入品」を表示することがないように、ログインしているユーザのショッピングカートの一覧だけに絞り込みます。ユーザは、この段階では購入を取り消して、ショッピングカートを廃棄したり、購入数を変更できるものとします。

このため、ここではscaffoldではなく、モデルとコントローラとを独立に生成することにします。

# カートモデルの生成

---

Type the following command (in one line);  
rails generate model Cart user\_id:integer  
merchandise\_id:integer  
number\_of\_items:integer

```
sqlite> .schema
CREATE TABLE "attachments" ("id" INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, "merchandise_id" integer, "name" varchar(255), "size" integer, "content_type" varchar(255), "content" blob, "created_at" datetime, "updated_at" datetime);
CREATE TABLE "carts" ("id" INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, "user_id" integer, "merchandise_id" integer, "number_of_items" integer, "created_at" datetime, "updated_at" datetime);
CREATE TABLE "customers" ("id" INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, "user_id" integer, "name" varchar(255), "zipcode" varchar(255), "street_address" varchar(255), "city" varchar(255), "phone" varchar(255), "created_at" datetime, "updated_at" datetime);
CREATE TABLE "merchandises" ("id" INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, "category" varchar(255), "name" varchar(255), "price" integer, "max_order" integer, "created_at" datetime, "updated_at" datetime, "photo" blob, "file_name" varchar(255), "file_type" varchar(255));
CREATE TABLE "order_items" ("id" INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, "order_id" integer, "merchandise_id" integer, "number_of_items" integer, "created_at" datetime, "updated_at" datetime);
```

# 今日の欠席課題

---

ユーザ認証によって、切り替えたメニューを添付して下さい。

細かい説明は不要です。画面コピーをつけて下さい。