

Web System Development with Ruby on Rails



Day 10(29/Nov/2012)
Data Flow in Rails

Difference between private and protected (My Homework!)

There are two method types;
function type, and receiver type.

Private methods can be called only in function type. Protected methods can be called both in function type and receiver type.

Both private and protected methods can be called from self and subclasses.

Private and Protected sample

```
class Hello
  private
  def hello
    puts "hello world"
  end
end
```

```
class Hello2 < Hello
  def say
    hello
  end
End
```

```
Hello2.new.say
# => hello world
```

```
class Hello
  private
  def hello
    puts "hello world"
  end
end
```

```
Hello.new.hello
# => (NoMethodError)
```

Object#Method Class

```
class Foo
  def foo() "foo" end
  def bar() "bar" end
  def baz() "baz" end
end
```

```
obj = Foo.new
```

```
methods = { 1 => obj.method(:foo),
            2 => obj.method(:bar),
            3 => obj.method(:baz) }
```

```
p methods[1].call # => "foo"
p methods[2].call # => "bar"
p methods[3].call # => "baz"
```

<http://doc.ruby-lang.org/ja/1.9.3/class/Method.html>

Today's Theme

- Tracing the Data Flow in Rails environment when the system is operated via WEB screen.

Form input in html

- Data are transferred when users input data in a 'form_for' or a 'form_tag' area in the file.
- Those areas are surrounded by <form> tags in a rendered HTML file.

- `_form.html.erb`

```
_form.html.erb x memo.rb edit
1 <%= form_for(@memo) do |f| %>
2   <% if @memo.errors.any? %>
3     <div id="error_explanation">
4       <h2><%= pluralize(@memo.e
5
6     <ul>
7       <% @memo.errors.full_messa
8       <li><%= msg %></li>
9     <% end %>
```

[Listing memos](#) | [New Memo](#) | [Ruby Official Site](#) |

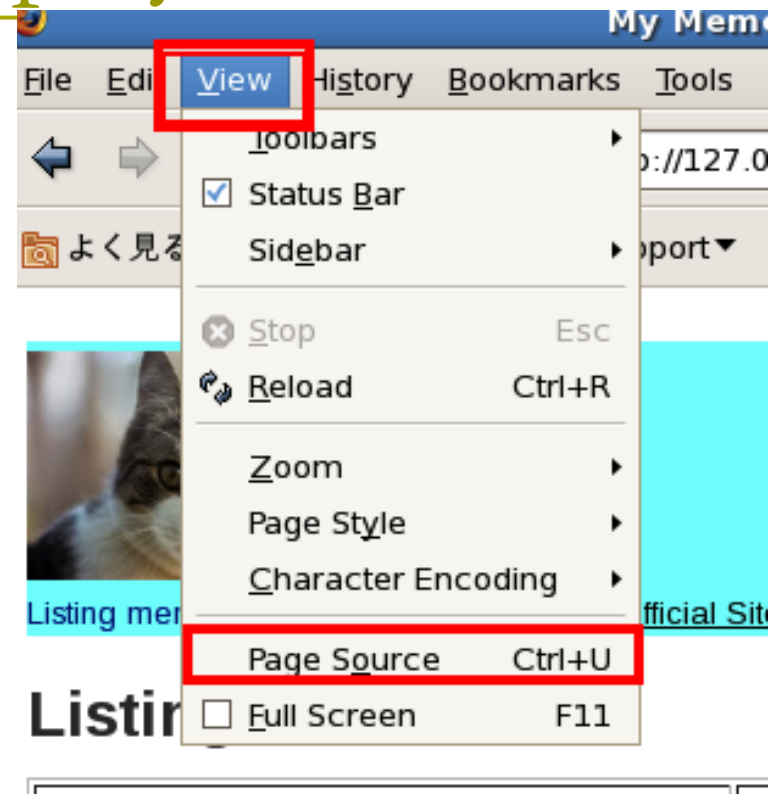
Editing memo

Content

Yesterday, it was rainy because of the typhoon.

Html source code display

- Form tag is generated in a html file from _form.html.erb file.
- action, class id, method and such parameters are generated.



```
<form accept-charset="UTF-8" action="/memos/1" class="edit_memo" id="edit_memo_1" metho  
  
  <div class="field">  
    <label for="memo_content">Content</label><br />  
    <textarea cols="40" id="memo_content" name="memo[content]" rows="20">  
Yesterday, it was rainy because of the typhoon.  
  </textarea>
```

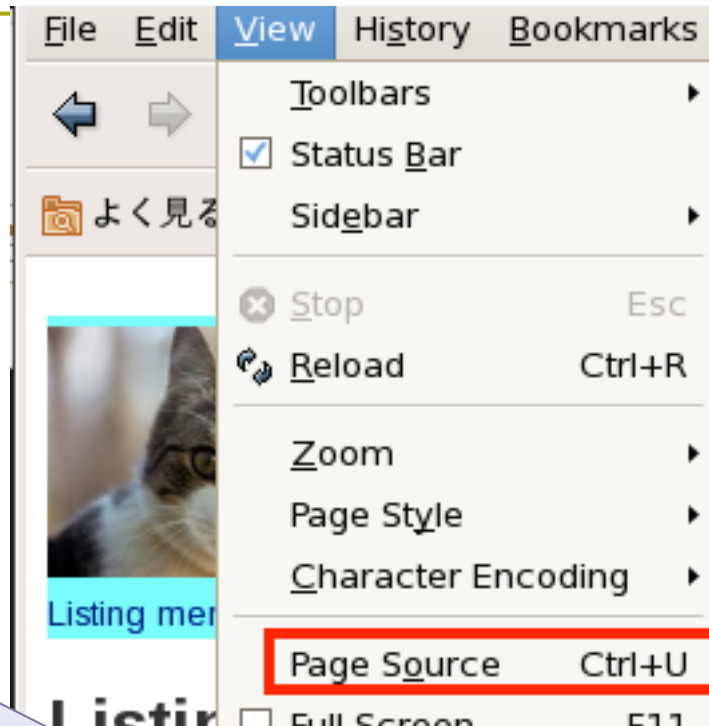
Read index.html.erb in the browser

```
</div>
<div id="left">
  <h1>Listing memos</h1>

  <table border="1">
    <tr>
      <th>Content</th>
      <th>Category</th>
      <th></th>
      <th></th>
      <th></th>
    </tr>

    <tr>
      <td>Yesterday, it was rainy because of tl
    </td>
    ...
  </table>

```



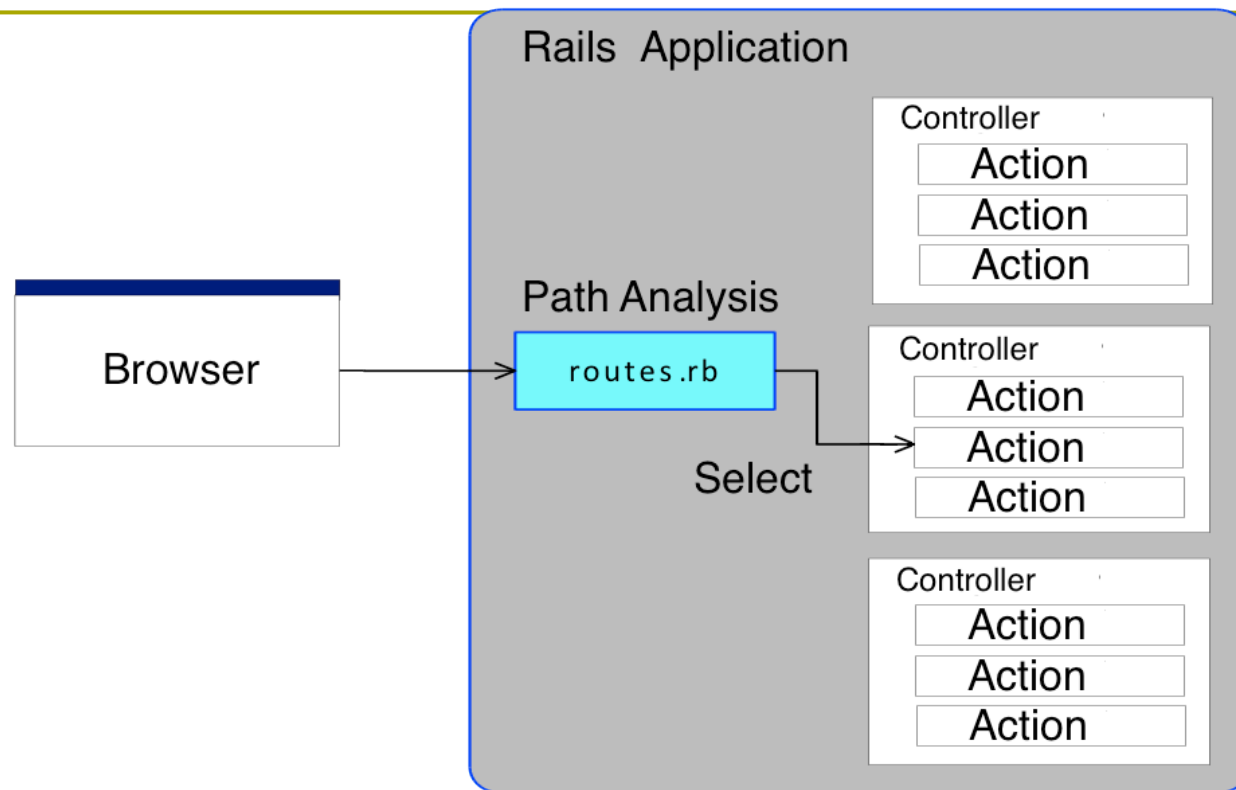
```
@page_title = 'app_title' %>
<h1><%= t :list_memo %></h1>

<table border="1">
  <tr>
    <th><%= t :content %></th>
    <th>Category</th>
    <th></th>
    <th></th>
    <th></th>
  </tr>

```

When you debug, always check the generated source code.

Data Transfer from Html to Ruby



- HTML POSTs data from FORM area in the file, then the path are analyzed with the information given in routes.rb.

memos_controller.rb

- The name of Memos' Controller is memos_controller.rb
- Each methods are 'Actions' of the controller.

```
<form accept-charset="UTF-8" action="/memos/1" class="edit_memo" :
```

- In form_tag of html, it declares action='memos/1'.
- Well, then, what does this '1' means?

Rake routes

In the GNOME command terminal, type
`rake routes`

```
[root@cisnote memopad]# rake routes
  categories GET    /categories(.:format)      categories#index
               POST   /categories(.:format)      categories#create
  new_category GET    /categories/new(.:format)  categories#new
  edit_category GET    /categories/:id/edit(.:format) categories#edit
    category GET    /categories/:id(.:format)  categories#show
               PUT    /categories/:id(.:format)  categories#update
               DELETE /categories/:id(.:format)  categories#destroy
    memos GET    /memos(.:format)          memos#index
               POST   /memos(.:format)          memos#create
  new_memo GET    /memos/new(.:format)      memos#new
  edit_memo GET    /memos/:id/edit(.:format) memos#edit
    memo GET    /memos/:id(.:format)      memos#show
               PUT    /memos/:id(.:format)      memos#update
               DELETE /memos/:id(.:format)      memos#destroy
[root@cisnote memopad]#
```

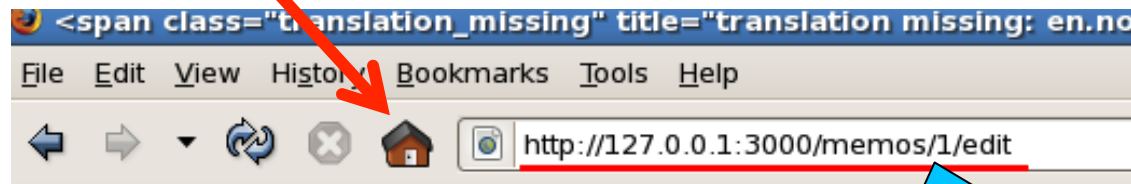
routes

```
memos GET /memos(.:format) memos#index
      POST /memos(.:format) memos#create
new_memo GET /memos/new(.:format) memos#new
edit_memo GET /memos/:id/edit(.:format) memos#edit
```

GET is a html method

Controller is memos, and 'action' is 'edit'

Path name in ruby program.
Ex. edit_memo is referred as 'edit_memo_path'



1 is an :id, in the memos table

```
13 <% @memos.each do |memo| %>
14 <tr>
15   <td><%= memo.content %></td>
16   <td><%= memo.category.name %></td>
17   <td><%= link_to (t 'show'), memo %></td>
18   <td><%= link_to (t 'edit'), edit_memo_path(memo) %></td>
19   <td><%= link_to (t 'destroy'), memo, method: :delete, data: { :confirm => t('confirm_delete') } %></td>
20 </tr>
21 <% end %>
```

POST, GET, UPDATE, and DELETE

- Data hand-in, from HTML file to the Server
- There are POST, GET, UPDATE, and DELETE methods in Rails 3 version.

Source of: <http://127.0.0.1:3000/memos/new> - Mozilla Firefox

```
<form accept-charset="UTF-8" action="/memos" class="new_memo" id="new_memo" method="post"><
```

HTTP command and ruby method

- Comment in quizsets_controller.rb compared with the method name

GET /memos	index
GET /memos/1	show
GET /memos/new	new
GET /memos/1/edit	edit
POST /memos	create
PUT /memos/1	update
DELETE /memos/1	destroy

Want to use search with String

Rails strongly recommend to use **GET** method, when we want to search with certain strings.

```
<%= form_tag(search_path, :method => "get") do %>
  <%= label_tag(:q, "Search for:") %>
  <%= text_field_tag(:q) %>
  <%= submit_tag("Search") %>
<% end %>
```

Types of HTTP methods

- GET Information Request
- POST Information Transfer
- HEAD Request only for Headers
- PUT Upload
- DELETE Resource Removal
- TRACE Call back Request
- OPTIONS Obtain methods which Server can reply

One Mystery

When we click 'edit' memo, and see the source code in the browser, there is 'post' method, specified as options.

```
<form accept-charset="UTF-8" action="/memos/1" class="edit_memo" id="edit_memo_1" method="post">
```

But, edit should send **PUT** method to call update method.

```
memo GET    /memos/:id(.:format)      memos#show
      PUT    /memos/:id(.:format)      memos#update
      DELETE /memos/:id(.:format)      memos#destroy
```

And actually, **PUT** method is sent to the controller!

Why?

```
Started PUT "/memos/1" for 127.0.0.1 at 2012-11-2
Processing by MemosController#update as HTML
  Parameters: {"utf8"=>"✓", "authenticity_token"=
GTEVHP4bmrw2vc=". "memo"=>{"content"=>"Yesterday.
```

Special Report Theme!

Does anyone solve this mystery?

If you solve this trick, and submit this report, it will be counted as an optional bonus score to your mark.

Parameter: params[] array

- At the last of URL string, add ? name=value
- Input parameter from the html screen (developed from erb) will be given to the Controller files written in ruby

Usable tags in the form

- Radio button(and buttons)
 - Exclusive selection; only one button can be selected
 - `form.radio_button`
- Check Box(input box)
 - Each Box can be independently checked
 - `form.check_box`
 - Image file upload field
 - `file_field`

Input tags in a form

- **String Input Field**
 - Input field for String, basically assume an input in one line.
 - `form.text_field`
- **Text Input Field**
 - Input field for Text, assume multi lines long text.
 - `form.text_area`
- **Drop Down List**
 - Select one item from the list
 - `form.select`

link_to

- When link is clicked, browser will show the linked page.

```
<%= link_to 'New Memo', new_memo_path %>
```

This statement will show 'New Memo' string, and when the user click this string, the action 'new' will be executed.

- In link_to method, 'new_memo_path' will be routed to call 'memos#new', as described in 'rake routes' reponse, means 'memos' controller, and 'new' action.

edit_memo GET /memos/:id/edit(.:format) memos#edit

```
@memos.each do |memo| %>  
<tr>  
  <td><%= memo.content %></td>  
  <td><%= memo.category.name %></td>  
  <td><%= link_to (t 'show'), memo %></td>  
  <td><%= link_to (t 'edit'), edit_memo_path(memo) %></td>  
  <td><%= link_to (t 'destroy'), memo, method: :delete, %></td>  
</tr>  
<%= end %>  
</table>
```

index.html.erb

```
1 <h1>Editing category</h1>  
2  
3 <%= render 'form' %>  
4  
5 <%= link_to 'Show', @category %> |  
6 <%= link_to 'Back', categories_path %>  
7
```

edit.html.erb | c...

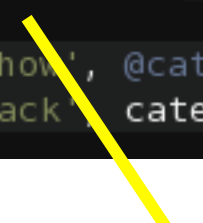
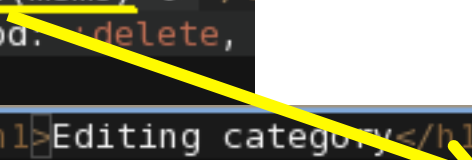
One memo record is selected

```
1 <%= form_for(@memo) do |f| %>  
2   <%= if @memo.errors.any? %>  
3     <div id="error_explanation">  
4       <h2><%= pluralize(@memo.errors.count, "error") %>  
5  
6     <ul>
```

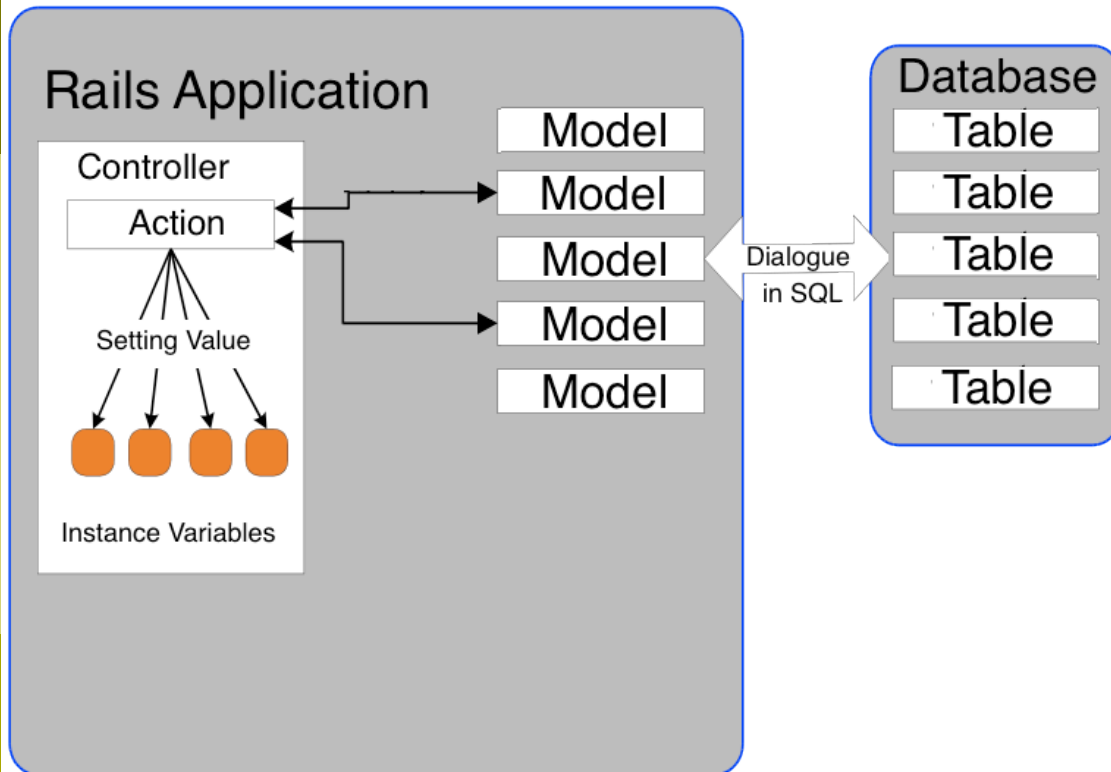
_form.html.erb

```
23 </div>  
24 <div class="actions">  
25   <%= f.submit %>  
26 </div>  
27 <%= end %>  
28
```

Controller



Find method of ActiveRecord



- Issue the SQL, fetch the value from DB, and returned in Hash Array.

```
@memos = Memo.find.all
```

```
@memos = Memo.find_all_by_category_id(params[:get][:categ_id])
```

```
@memo = Memo.find(params[:id])
```


respond_to / format.html

- From Rails 2, 'respond_to' is available
 - "format" in "respond_to do |format|" is the same name of action(method) in memos Controller.
- In respond_to block, html method is executed, and then json method is executed.
- If there is no json file in the given address, (action).html.erb is rendered.
Otherwise, do nothing.

```
26 def new
27   @memo = Memo.new
28
29   respond_to do |format|
30     format.html # new.html.erb
31     format.json { render json: @memo }
32   end
33 end
```

respond_to / format.xml

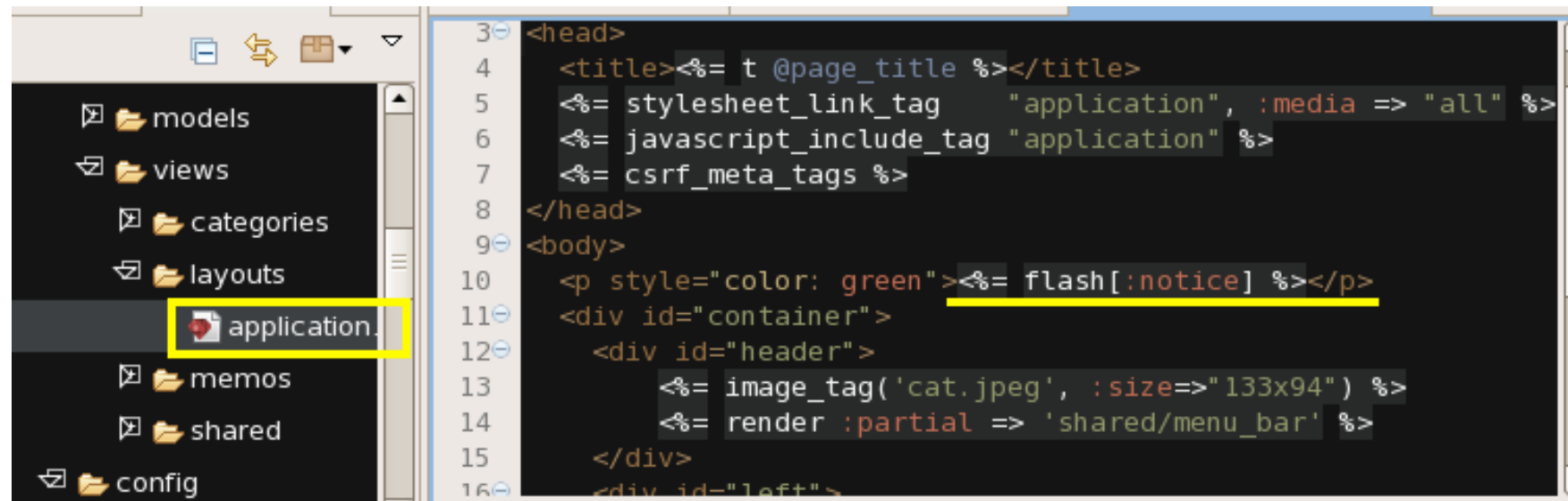
- In format.xml
Do nothing, when xml is not the extension in the address, and if the extension is xml,

```
{ render :xml => @question }  
(show @question in xml format)
```

is executed.
- This is the mechanism so that the program can be applied to json format, and other formats.
- We only have to think about index.html.erb, for a while.

Display area for :notice

- :notice is displayed in `<%= flash[:notice] %>` in the top of body part in views / layouts / application.html.erb
- “Success” message for question and such.



```
30 <head>
4   <title><%= t @page_title %></title>
5   <%= stylesheet_link_tag "application", :media => "all" %>
6   <%= javascript_include_tag "application" %>
7   <%= csrf_meta_tags %>
8 </head>
9 <body>
10  <p style="color: green"><%= flash[:notice] %></p>
11  <div id="container">
12    <div id="header">
13      <%= image_tag('cat.jpeg', :size=>"133x94") %>
14      <%= render :partial => 'shared/menu_bar' %>
15    </div>
16    <div id="left">
```

From Rails 3.0...

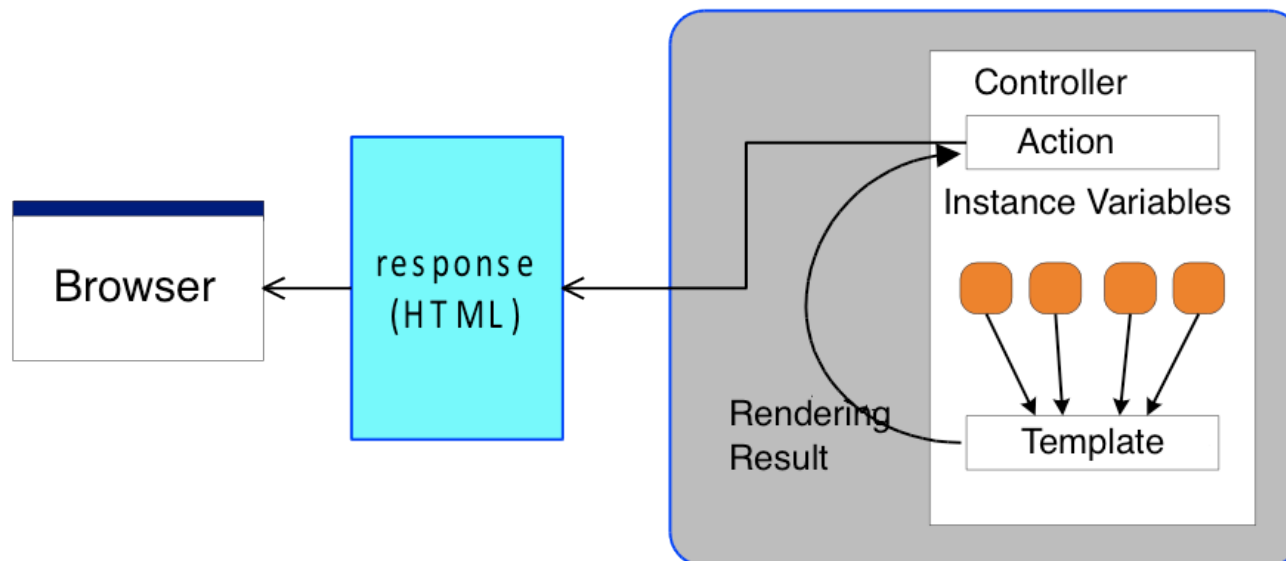
- Much simplified description of “respond_with” has become available ...
- From <http://api.rubyonrails.org/>
 - Search for “ActionController::Responder”

Sample:

```
respond_to :html, :json
def update
  @user = User.find(params[:id])
  @user.update_attributes(params[:user])
  respond_with @user, :location => users_path
end
```

Rendering

- Basically, html.erb file which has the same name with action name is rendered to generate the html file, and displayed.
- When `:template => TemplateName` is specified, that file will be displayed



New and Update

When we call 'New' action, `new.html.erb` will be rendered, then, new instance will be handed to 'create' action, and the result will be handed to `show.html.erb`.

When we call 'Edit' action, `edit.html.erb` will be rendered, then, updated instance will be handed to 'update' action, and the result will be handed to `show.html.erb`.

Today's practice

- Theme one:
 - Access to your friends PC page, and input memo.
 - Check the console screen, and confirm the input memo in the parameters.

Theme 1

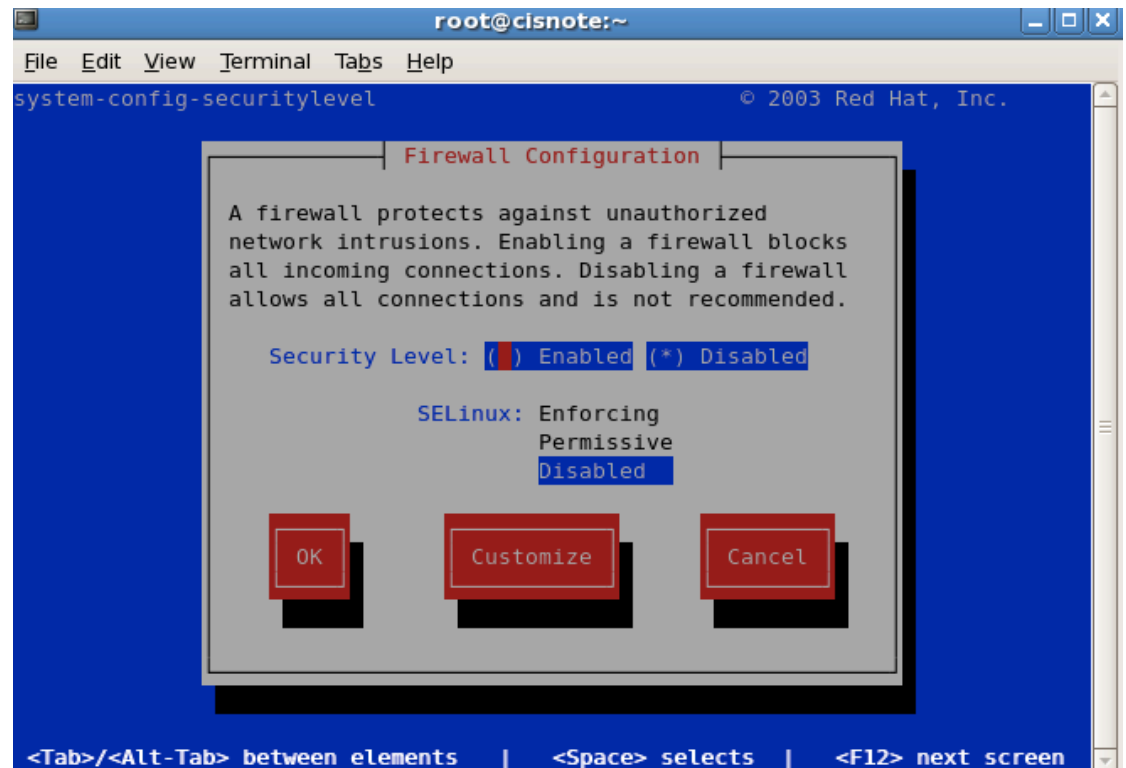
- Access to the WEB site in your friend's PC
- Step1 :Check your own URL (IP Address)
 - Usually, client PC access to the internet via DHCP server, so the IP address differs every time the PC is accepted to the network. With this theme, check that IP address, given by DHCP.

```
[root@cisnote ~]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0C:29:9B:FD:BC
          inet addr:192.168.252.130  Bcast:192.168.252.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:26 errors:15 dropped:0 overruns:0 frame:0
          TX packets:11 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1884 (1.8 KiB)  TX bytes:2862 (2.7 KiB)
          Interrupt:225 Base address:0x2024
```


Theme

Step 2: type

`system-config-securitylevel-tui`
to disable firewall.

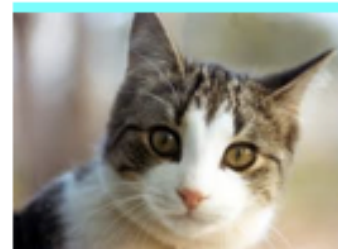
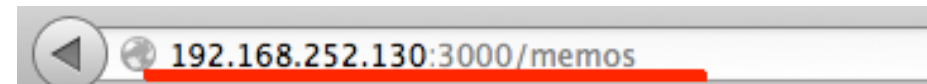
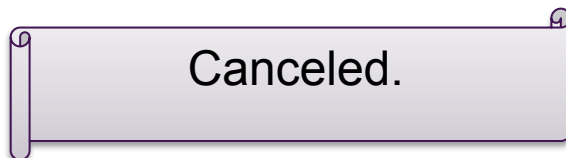


IP Address of your PC

- Click “Local Area Connection” and from “Detail” to go and get “IP v4” address.
- And..., the original plan was to let your friends access to your computer to show your WEB page, but due to the Network Connection Settings(NAT/Bridge in CentOS,) I failed to confirm the settings with my Virtual CentOS computer distribution for the WEB+DB class.

Show your own site on your friends PC

- Tell your own IP Address to your friends, and ask his/her IP address.
- The IP address used to be 127.0.0.1:3000, but this part should be replaced with your friends PC's address; like
- `http://192.168.0.11:3000`
- Your PC should show your friends site.



[Listing memos](#) | [New Memo](#) | [Ruby Off](#)

Listing memos

Content
Yesterday, it was rainy because of the typhc

[New Memo](#)

Check in Ruby console

- When your friends access to your PC, there is a reaction in the console

PC's IP address,
from which access
to your PC

```
GET "/assets/categories.js?body=1" for 192.168.252.1 at 2012-11-26 00:58:25 +0900
Served asset /categories.js - 304 Not Modified (1ms)
[2012-11-26 00:58:25] WARN Could not determine content-length of response body.
Set content-length of the response or set Response#chunked = true
```

```
Started GET "/assets/memos.js?body=1" for 192.168.252.1 at 2012-11-26 00:58:25 +0900
Served asset /memos.js - 304 Not Modified (1ms)
[2012-11-26 00:58:25] WARN Could not determine content-length of response body.
Set content-length of the response or set Response#chunked = true
```

Canceled.

New Theme

Check the server Access
only on your PC.

```
C:\Windows\system32\cmd.exe - ruby script\server
Completed in 0.07800 (12 reqs/sec) | Rendering: 0.04800 (61%) | DB: 0.01500 (19%) | 200 OK [http://192.168.0.11/memos]

Processing MemosController#search (for 192.168.0.2 at 2011-06-30 01:15:39) [POST]
Session ID: BAh7BzoMY3NyZi9pZCI1ODMyMGJjMzgxOTFjNGJlNjE4OWFiYjIiN2FkZjU4%0ANmM
iCmZsYXNoSUM6J0FjdGlvbknvbnRyb2xsZXI160kZsYXNo0jpbGbfzaEhh%0Ac2h7AAY6CkBlc2VkeWwA%
3D--f8da521cacb7d0ec90af6ec2142622862f39775c
Parameters: {"get"=>{"categ_id"=>"1"}, "commit"=>"", "authenticity_token"=>"9bb67c73d185cfa4c93d44af50140114ab074d93", "action"=>"search", "method"=>"get", "controller"=>"memos"}
SELECT * FROM `memos` WHERE (`memos`.`category_id` = '1')
Rendering template within layouts/memos
Rendering memos/search
SELECT * FROM `categories` WHERE (
SELECT * FROM `categories` WHERE (`categ
SELECT * FROM `categories`
Rendered shared/_menu_bar (0.00000)
Completed in 0.07800 (12 reqs/sec) | Rendering: 0.03200 (41%) | DB: 0.01600 (20%) | 200 OK [http://192.168.0.11/memos/search?method=get]
.0.11/memos/search?method=get]
```

IP address, from which accesses to your PC

Parameters in Form tags

Renderig Search.html.erb

SQL responding To the message

Now do int on your own PC

Your own IP address

Report Themes

- For Evaluation C or B
 - Attach the screen shots of the screen when your friends accessed to your PC, and show your friends PC's IP address.
- For Evaluation A
 - Explain what happened of the console messages when there is any access to screens.
- For Evaluation S
 - Add SQL database access explanation, correspondent to ruby programs in rails.

Absence Report

Read today's slides, and ask your friends to access to your PC, attach the screen shots of the console, that shows the access from the outside of your PC.

If you attach this screen shots, your absence record is changed to presence.

If you submit the regular report of the previous page, the regular report is counted as an absence report also.