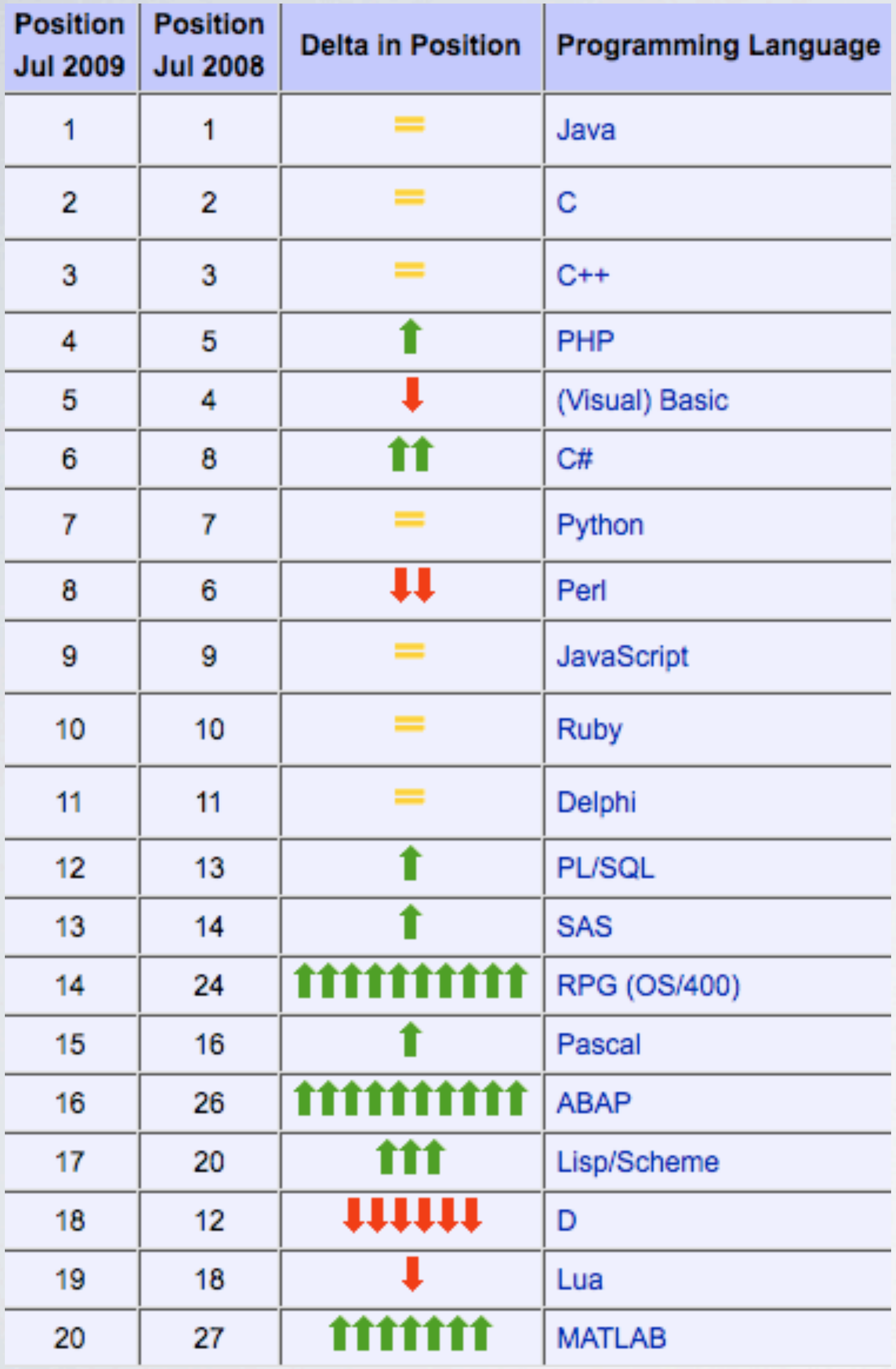
網頁系統開發十倍速的魔法

Happy , Fun & More free time

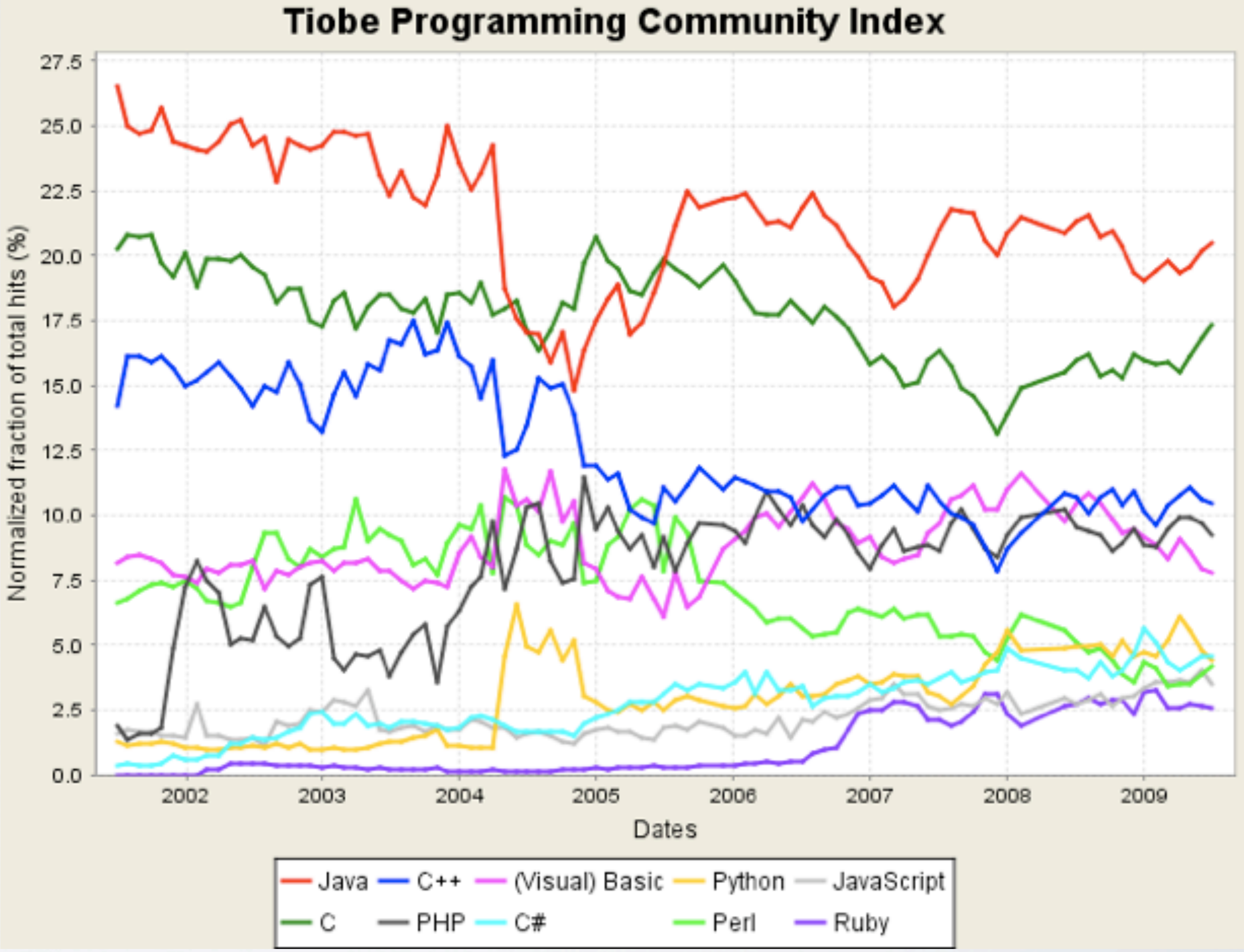
RUBY ON RAILS

網頁系統開發十倍速的魔法

Happy , Fun & More free time



Position Jul 2009	Position Jul 2008	Delta in Position	Programming Language
1	1	=	Java
2	2	=	C
3	3	=	C++
4	5	⬆	PHP
5	4	⬇	(Visual) Basic
6	8	⬆⬆	C#
7	7	=	Python
8	6	⬇⬇	Perl
9	9	=	JavaScript
10	10	=	Ruby
11	11	=	Delphi
12	13	⬆	PL/SQL
13	14	⬆	SAS
14	24	⬆⬆⬆⬆⬆⬆⬆⬆⬆⬆	RPG (OS/400)
15	16	⬆	Pascal
16	26	⬆⬆⬆⬆⬆⬆⬆⬆⬆⬆	ABAP
17	20	⬆⬆⬆	Lisp/Scheme
18	12	⬇⬇⬇⬇⬇⬇	D
19	18	⬇	Lua
20	27	⬆⬆⬆⬆⬆⬆⬆	MATLAB



Ruby

a language , like JAVA

Rails

a framework , like Spring

Ruby On Rails : 使用Ruby語言來實作Rails

Ruby

Object-Oriented Script Programming Language
松本行弘（matz）開發，遵守 Ruby License。

1995 年 12 月才正式公開

Perl 的發音與 6 月的誕生石 pearl（珍珠）
Ruby 以 7 月的誕生石 ruby（紅寶石）

一切以人為本
減少程式設計時候的不必要的瑣碎時間
令程式設計師高興

人性化，而不是
一味從機器的角度設想

我們是主人，他們是僕人

Ruby

一個script語言，像是PHP，不用編譯

完整的物件繼承，像是Java

任何東西都是物件，沒有Null只有Nil
因為Ruby內的Nil(Null)也是物件

Ruby

```
1.class => Fixnum
9999999999.class => Bignum
"Hello World!".class => String
Time.now.class => Time
nil.class => NilClass

(1..99).class => Range
[1,2,3].class => Array
(1..3).to_a => [1, 2, 3]
```

Ruby

```
class A
  def test_1
    puts “I’m test_1”
    self.test_3
  end
  def self.test_2(temp)
    puts “I’m test_2” + temp
    test_4
  end
  private
  def test_3
    puts “I’m test_3”
  end
  def self.test_4
    puts “I’m test_4”
  end
end
class B < A
end
```

```
a = A.new
a.test_1 #=> “I’m test_1”
             “I’m test_3”
A.test_2(“ Hi”) #=> “I’m test_2 Hi”
                #=> “I’m test_4”
b = B.new
b.test_4 #=> error , method missing
B.test_4 #=> “I’m test_4”
B.test_3 #=> error , method missing
b.test_3 #=> error , call private
```

Ruby

有”for”但沒人用
當迴圈變得更有趣時......

當一切都變得直觀且Fun時......

Ruby

```
3.times do |x|                              count0
    puts “count” + x.to_s                   count1
end                                         count2
[“A”,”b”,”C”].each do |item|                I’m A
    puts “I’m ” + item                      I’m b
end                                         I’m C

puts “:)” if “I’m happy” || nil             :)

puts “:(” unless “Money” || nil

Im_A , Im_B = Im_B , ImA                    (swap)
```

Ruby

沒有Overloading但我們有Symbol和Hash

Symbol : 一個特別的指標，like String but not String
Hash : 一個特別的儲域，like Array but Free Style

Ruby if Overloading

```
def eat(dinner,drink)
    puts “I eat #{dinner}”
    eat(drink)
end
def eat(drink)
    puts “& I drink #{drink}”
end

eat(“egg”,“cola”)

#=> I eat egg & I drink cola
```

a badway......

Ruby   if Hash & Symbol

```
def eat(something)

    puts "I eat #{something[:dinner]} "
    puts something[:drink] ? "& I drink #{something[:drink]}" : nil

end

eat({:dinner => “egg”,  :drink => “cola”})
eat(:dinner => “egg”,  :drink => “cola”)
eat :dinner => “egg”,  :drink => “cola”

#=> I eat egg & I drink cola
```

Fun :)

Ruby

Meta-Programming

Ruby

```
# Coded by CFC <cfcsky (at) gmail (dot) com>
# make something if you want

def make something
  puts "I've made #{something}"
end

def you want
  return want
end

something, want = "something", true
make something if you want
```

```
#=> I've made something
```

Ruby

不會用？？問它！！

Ruby

```
"HiHi".methods
```

```
["chop!", "concat", "decode_b", "breakpoint", "upcase!", "collect", "require", "taguri", "kconv", "each_cons", "suppress",
"methods", "method", "sort", "member?", "chop", "taguri=", "replace", "each_with_object", "include?", "each_with_index",
"daemonize", "taint", "instance_variable_defined?", "to_datetime", "present?", "class_eval", "at", "empty?", "crypt", "toeuc",
"chomp!", "downcase!", "map", "unpack", "pluralize", "match", "metaclass", "slice!", "to_yaml_style", "__id__", "enum_cons",
"singleton_methods", "sort_by", "to_set", "titleize", "dclone", "object_id", "chomp", "subclasses_of", "tr!", "respond_to?",
"index_by", "foreign_key", "zip", "to_str", "start_with?", "nil?", "intern", "strip!", "capitalize!", "inject", "succ", "tap", "__send__",
"require_library_or_gem", "protected_methods", "require_or_load", "humanize", "grep", "tainted?", "strip", "tr_s!", "each",
"dump", "untaint", "instance_of?", "first", "starts_with?", "iseuc", "equal?", "scan", "b64encode", "lstrip!", "swapcase!", "partition",
"squish", "with_options", "classify", "returning", "display", "is_utf8?", "hex", "private_methods", "hash", "find", "send", "lstrip",
"instance_values", "`", "delete!", "upcase", "kind_of?", "enable_warnings", "freeze", "ends_with?", "succ!", "issjis", "last", "rstrip!",
"eql?", "ljust", "next", "to_json", "id", "tr", "tojis", "is_complex_yaml?", "oct", "public_methods", "acts_like?", "underscore",
"detect", "many?", "require_association", "casecmp", "instance_eval", "instance_variable_names", "squeeze!", "blank?", "rstrip",
"%", "to_enum", "downcase", "is_a?", "silence_stderr", "each_line", "isutf8", "all?", "end_with?", "rjust", "squish!",
"silence_warnings", "to", "mb_chars", "type", "tosjis", "is_binary_data?", "*", "tr_s", "camelize", "split", "+", "to_d",
"instance_variables", "find_all", "none?", "insert", "singularize", "demodulize", "frozen?", "chars",
"copy_instance_variables_from", "enum_for", "capitalize", "silence_stream", "to_time", "any?", "to_a", "to_param", "next!",
"each_byte", "center", "acts_like_string?", "extended_by", "min", "each_char", "upto", "from", "class", "delete", "constantize",
"toutf8", "to_sym", "reverse", "to_xs", "encode64", "sub!", "instance_variable_get", "<=>", "to_yaml",
"load_with_new_constant_marking", "length", "select", "==", "between?", "===", "enum_with_index", "duplicable?", "to_i",
"swapcase", "extend", "to_s", "to_query", "sum", "sub", ">=", "extend_with_included_modules_from", "max", "<=", "index",
"group_by", "tableize", "clone", "squeeze", "toutf16", "camelcase", "decode64", "gsub!", "require_dependency", "debugger",
"instance_variable_set", "=~", "ord", "size", "<", "dasherize", "parameterize", "reject", ">", "to_f", "enum_slice", "entries", "slice",
"inspect", "[]", "remove_subclasses_of", "instance_exec", "try", "gsub", "rindex", "each_slice", "reverse!", "[]=", "dup",
"to_yaml_properties", "to_date", "count", "titlecase", "unloadable", "<<"]
```

Ruby

不好用？？覆寫它！！

```
class A
    def b=(value)
        @b = value + 3
    end
    def b
        @b
    end
end
a = A.new
a.b = 3
puts a.b          #=> 6
```

Now , 快了幾倍了呢
現在還只是Ruby

如果加上Rails

No SQL , 真正的MVC

你要如何追得上ROR的開發速度

Ruby+Rails

```
Time.now - 1.day + 1.month + 1.year
#=>Sat Aug 14 08:00:38 +0800 2010
```

2004年7月，DHH(David Heinemeier Hanson)

Rails

「不要讓自己太累」（Don't Repeat Yourself）
「約定勝於配置」（Convention Over Configuration）

Ruby On Rails

一個FrameWork語言，像是Spring

完整的Routing，完整的MVC，像是Struts

默認大於配置，丟棄所有設定檔，像是????

什麼叫

MVC

Model , View , Control

(DB , Class) , (How to show) , (Flow)

Ruby On Rails : the DB & configuration

為什麼需要配置？

使用者表的名字叫什麼？

user? users ? account? accounts? login?

使用者表的主鍵叫什麼？

pk? sid? id? name? pkey?

使用者表有個外鍵連到book表叫什麼？

book_pk? book_sid? book_id? book_name? book_pkey?

到底怎樣是對的，怎樣是不好的？

Ruby On Rails : the DB & configuration

其實不需要配置，in Rails

使用者表的名字叫什麼？
你想要的model名稱加複數：users
使用者表的主鍵叫什麼？
主鍵一律都是：id
使用者表有個外鍵連到book表叫什麼？
對方model名稱不加複數的id：book_id

Hey，連思考時間都省下來了

Ruby On Rails : 續MVC

M V C

Table : Users[id:integer , name:string]

Model : User

Controller : @user = User.find(3)

View : <%=@user.name%>

it’s ActiveRecord

Ruby On Rails : ActiveRecord CRUD

C

R

U

D

```
@user = User.new(:name => “init me”)
@user.name = “JokerCatz”
@user.save
@user.update_attributes(:name => “MEOWs”)
@user.destroy
```

Hey , It’s NO SQL!! **又比你快了多少？**
but**還不夠，**DB & Table**要自己建起來？**
NoNoNo~ , It’s Joke in Rails

Ruby On Rails : Migration_01 config your database.yml

database.yml

```
development:
  adapter: sqlite3
  database: db/development.sqlite3
  pool: 5
  timeout: 5000

test:
  adapter: sqlite3
  database: db/test.sqlite3
  pool: 5
  timeout: 5000

production:
  adapter: sqlite3
  database: db/production.sqlite3
  pool: 5
  timeout: 5000
```

```
development:
  adapter: mysql
  encoding: utf8
  database: xxx_d
  username: root
  password: xxx
test:
  adapter: mysql
  encoding: utf8
  database: xxx_t
  username: root
  password: xxx

production:
  adapter: mysql
  encoding: utf8
  database: xxx_p
  username: root
  password: xxx
```

Ruby On Rails : Migration_02 Migrate it!!!!

```
$ ruby script/generate migration init_table
      create  db/migrate
      create  db/migrate/20090101010101_init_table.rb
```

```
class InitTable < ActiveRecord::Migration
  def self.up
    create_table :users do |t|
      t.string :name
      t.timestamps
    end
  end
  def self.down
    drop_table :users
  end
end
```

Ruby On Rails : Migration_03 Version it!!!!

```
$ ruby script/generate migration add_info_of_user
      create   db/migrate
      create   db/migrate/20090101010102_add_info_of_user.rb
```

```
class AddInfoOfUser < ActiveRecord::Migration
  def self.up
     add_column :user , :info:text
  end
  def self.down
     remove_column :user , :info
  end
end
```

Ruby On Rails : Migration_04 Use it!!!!

```
$rake db:create
$rake db:migrate
...info...(version 20090101010102)
$rake db:migrate:down VERSION=20090101010101
...info...(varsion 20090101010101)
$rake db:migrate:up VERSION=20090101010102
...info...(version 20090101010102)
$rake db:drop
```

Okay!!連DB連資料表版本都有了
But要自己code MVC好累.....
hmm...It's another joke in Rails

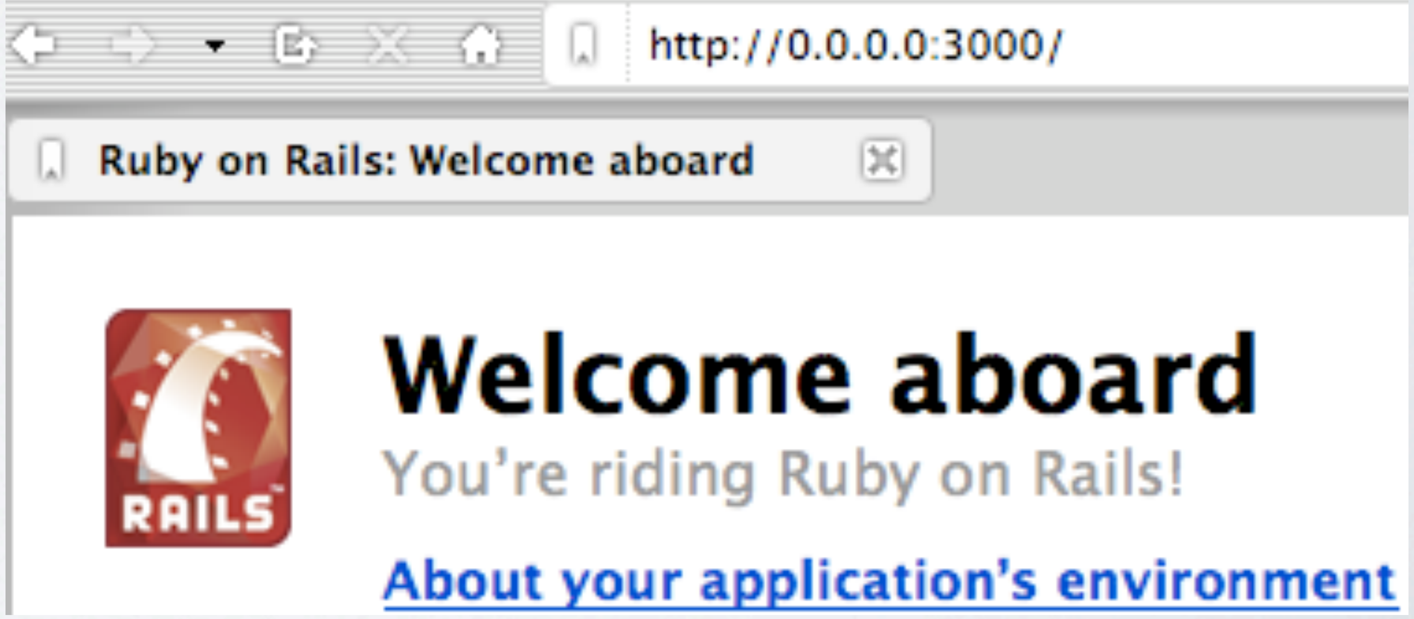
Ruby On Rails : Scaffold_01 : Migrate + MVC template

```
$ruby script/generate scaffold user name:string info.text

  ...
        create   app/views/users/index.html.erb
        create   app/views/users/show.html.erb
        create   app/views/users/new.html.erb
        create   app/views/users/edit.html.erb
  ...
        create   app/controllers/users_controller.rb
  ...
        create     app/models/user.rb
  ...
        create     db/migrate/20090715024027_create_users.rb
```

Ruby On Rails : Scaffold_02 : Migrate it & Run

```
$rake db:create
$rake db:migrate
$ruby script/server
  => Booting Mongrel
  => Rails 2.3.2 application starting on http://0.0.0.0:3000
  => Call with -d to detach
  => Ctrl-C to shutdown server
```



http://0.0.0.0:3000/

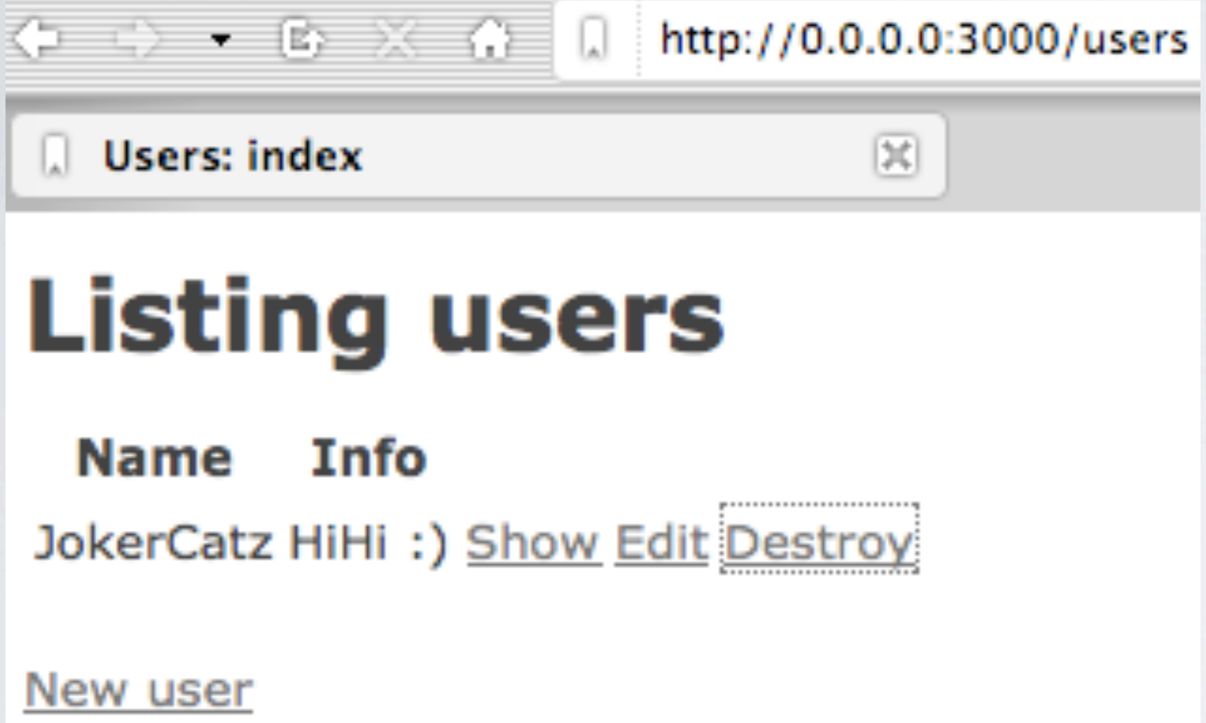
Ruby on Rails: Welcome aboard

Welcome aboard

You're riding Ruby on Rails!

About your application's environment

Ruby On Rails : Scaffold_03 CRUD



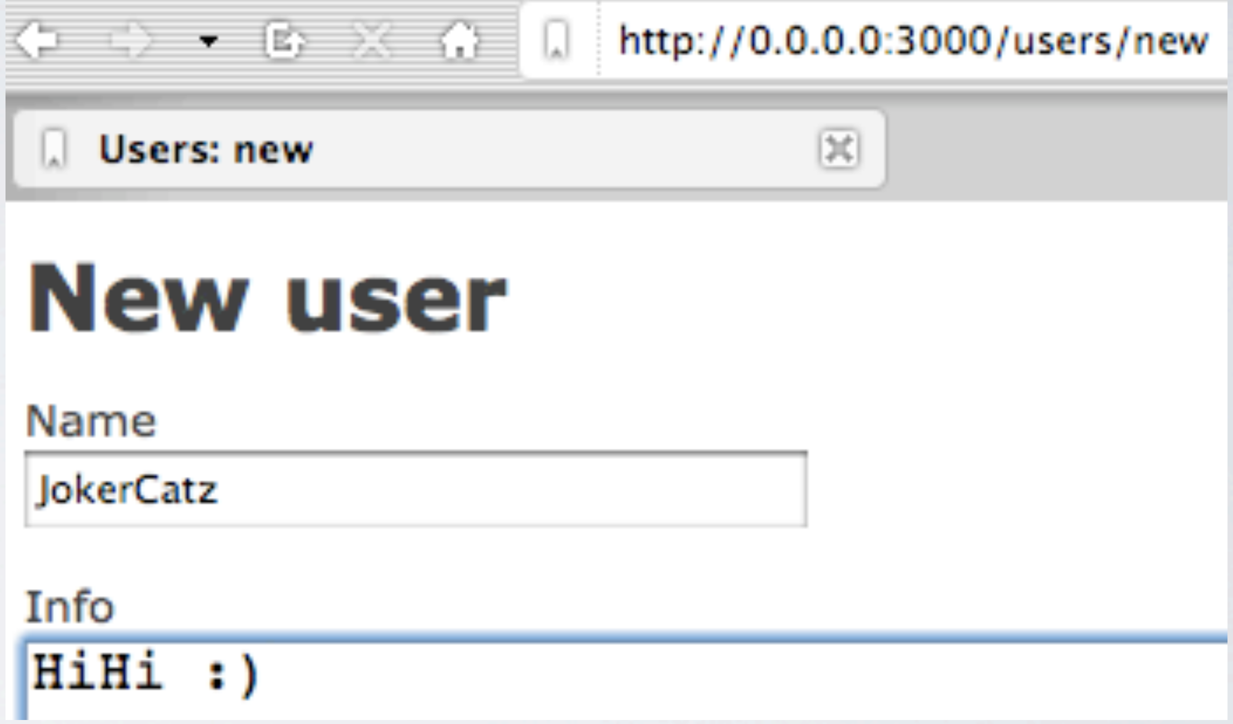
http://0.0.0.0:3000/users

Users: index

Listing users

Name	Info			
JokerCatz	HiHi :)	Show	Edit	Destroy

New user



http://0.0.0.0:3000/users/new

Users: new

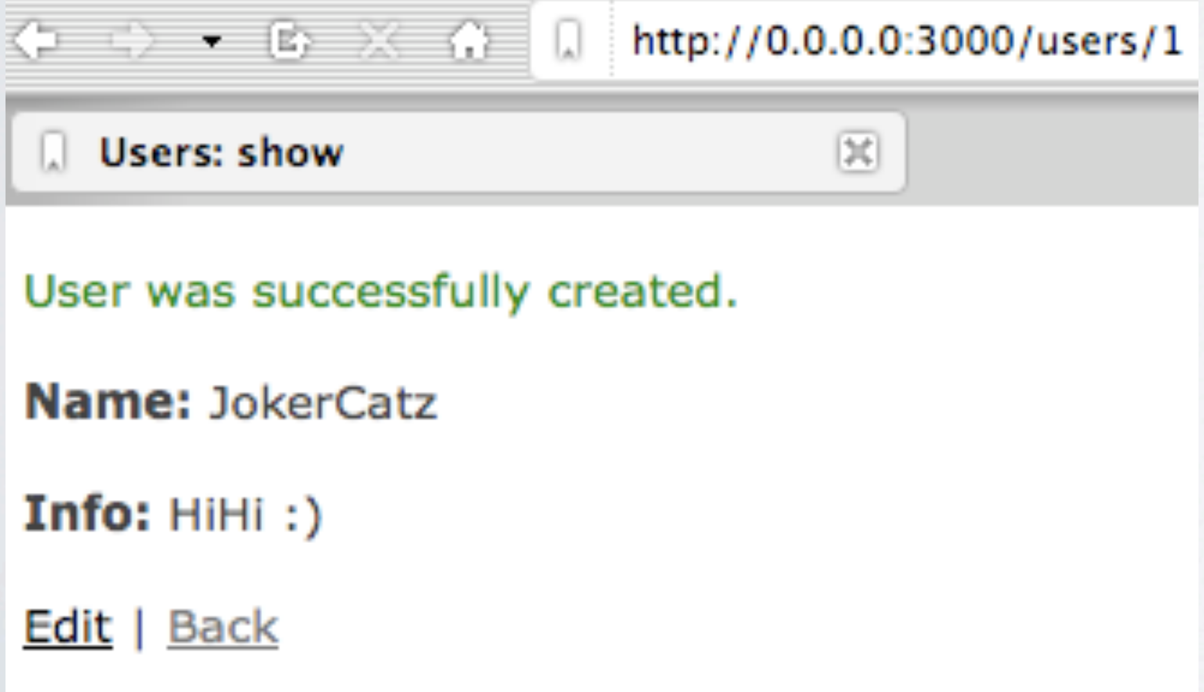
New user

Name

JokerCatz

Info

HiHi :)



http://0.0.0.0:3000/users/1

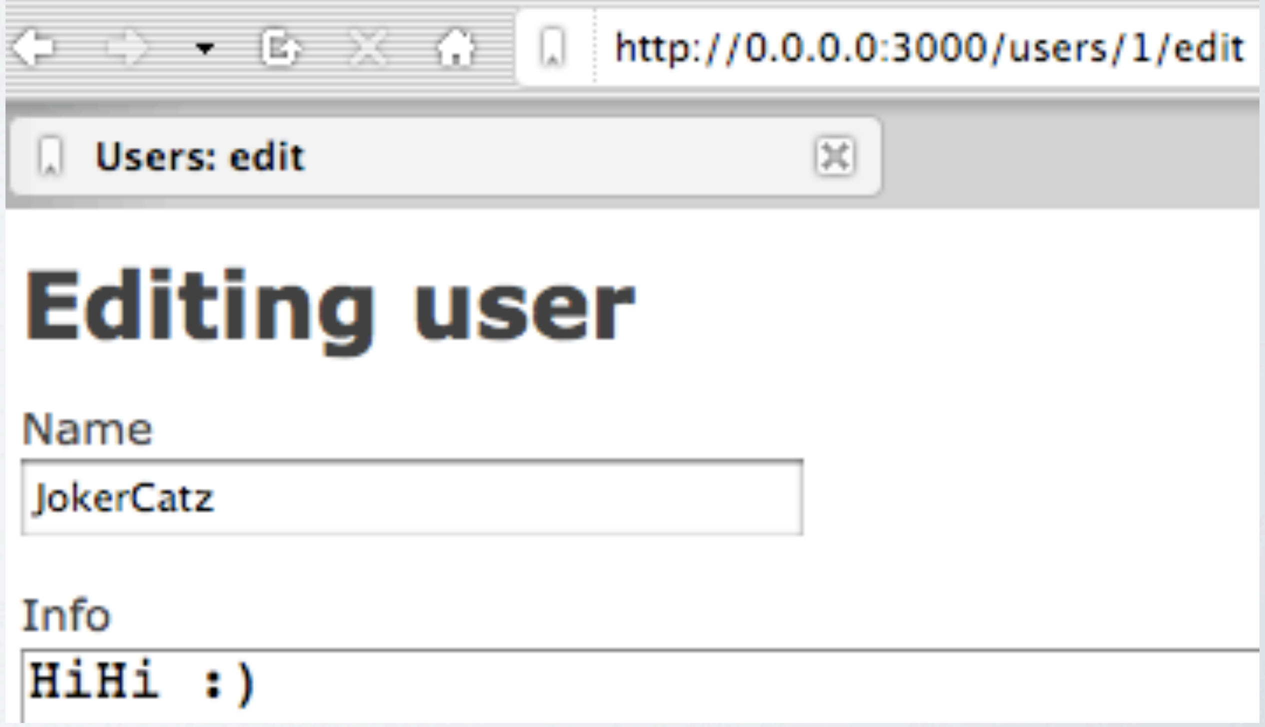
Users: show

User was successfully created.

Name: JokerCatz

Info: HiHi :)

Edit | Back



http://0.0.0.0:3000/users/1/edit

Users: edit

Editing user

Name

JokerCatz

Info

HiHi :)

Ruby On Rails : ActiveRecode : Base Relation

Has_One	1User => 1Profile
Has_Many	1Book => nPage
Belongs_to	nPage => 1Book 1Profile => 1User

Has_and_Belongs_to_Many	nOrder => nItem
	nItem => nOrder

Ruby On Rails : ActiveRecode : Use Base Relation

Has_One   @user.profile #=> profile

Has_Many   @book.page #=> Array[page]

Belongs_to   @profile.user
@page.book

Has_and_Belongs_to_Many
@order.Item #=> Array[item]
@item.order #=> Array[order]

Ruby On Rails : ActiveRecode : Addon & Plugin Relation

Acts_as_tree	1User => 1Parent(User) 1User => nChild(User)
Acts_as_nested_set	like Acts_as_tree 但用一個Select就可求得全部的子代
單表繼承	Single Table Inheritance
萬用Model	1Post => find(Owner) => 1User(or something)

Ruby On Rails : ActiveRecode : Use Another Relation

Acts_as_tree	@user.parent #=> user @user.child #=> Array[user]
Acts_as_nested_set	like Acts_as_tree
單表繼承	@animal #=> mouse(or something)
萬用Model	@post.owner #=> user(or something)

Ruby On Rails : the Plugin

More In **GitHub**

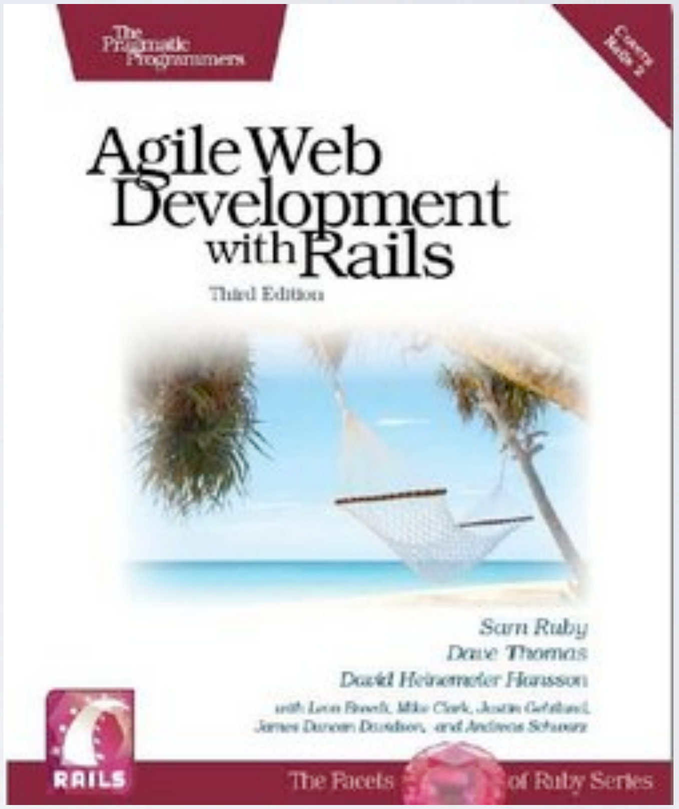
attachment_fu
recaptcha
awesome_nested_set
restful-authentication
acts-as-taggable-on
acts_as_ferret
acts_as_list
will_paginate
jrails
ym4r_gm

Ruby On Rails : End

More In Rails :

Cache
Rack
Route
Layout
Template
Merb
Gems(Ruby)

The “Hello World!!”


The Pragmatic Programmers
Covers Rails 2
Agile Web Development with Rails
Third Edition
Sam Ruby
Dave Thomas
David Heinemeier Hansson
with Leon Breedt, Mike Clark, Justin Gehtland, James Duncan Davidson, and Andreas Schwarz
RAILS
The Facets of Ruby Series

Ruby Programming
向Ruby**之父學程式設計**
Ruby1.8.6

Agile Web Development with Rails
(must 3rd)
Rails2.2.0

Welcome!!

http://railsenvy.com/

http://railsfun.tw

Thanx

Masahji & Ruby’s fans :)