

書いた人 : moriq
!ActiveResource を試してみないか資料

!インストール

ActiveResource をインストールする。

```
$ gem install activeresource --source http://gems.rubyonrails.org -y
```

ついでに Beta Gems をインストールしておくとうい。バージョンが揃うので。

```
$ gem update --source http://gems.rubyonrails.org -y
```

!サンプルアプリ (サーバ)

サンプルアプリを作成する。

database は Sqlite3 にしておく。

```
$ rails demo -d sqlite3
```

まず安直に generate scaffold_resource で作成。

```
$ script/generate scaffold_resource post title:string created_on:date body:text published:boolean
```

```
$ rake db:migrate
$ script/server
```

config/routes.rb を確認する。モデルを確認する。コントローラを確認する。

```
respond_to do |format|
  format.xml { render :xml => @post }
end
```

xml format でアクセスしてみる。

```
http://localhost:3000/posts/new.xml

<?xml version="1.0" encoding="UTF-8"?>
<post>
  <body></body>
  <created-on type="date"></created-on>
  <published type="boolean"></published>
  <title></title>
</post>
```

次に generate resource で作成。

```
$ script/generate resource person name:string
```

config/routes.rb を確認する。モデルを確認する。コントローラを確認する。

!ActiveResource (クライアント)

ActiveResource を使ってクライアントを書いてみる。

参考: [active_resource/README](#)

```
# client.rb:
require 'rubygems'
gem 'activeresource'
require 'active_resource'

class Person < ActiveResource::Base
  self.site = 'http://localhost:3000/'
end

matz = Person.new(:name => 'matz')
p matz.save #true
p matz.id #1

#matz = Person.find(1)
#matz.name = 'matz'
#p matz.save #true

#puts matz.to_xml
```

post のコントローラを参考にアクションを書いてみる。

```
# people_controller.rb:
class PeopleController < ApplicationController
  def index
    @people = Person.find(:all)
    render :xml => @people
  end

  def show
    @person = Person.find(params[:id])
    render :xml => @person
  end

  def new
    @person = Person.new
    render :xml => @person
  end

  def create
    @person = Person.new(params[:person])
    if @person.save
      flash[:notice] = 'Person was successfully created.'
      render :xml => @person, :status => :created, :location => person_url(@person)
    else
      render :xml => @person.errors
    end
  end
end
```

```
def update
  @person = Person.find(params[:id])
  if @person.update_attributes(params[:person])
    flash[:notice] = 'Person was successfully updated.'
    head :ok
  else
    render :xml => @person.errors
  end
end
end
```

Person オブジェクトの XML 表現を確認。

<http://localhost:3000/person/1.xml>

```
<?xml version="1.0" encoding="UTF-8"?>
<people>
  <person>
    <id type="integer">1</id>
    <name>matz</name>
  </person>
</people>
```

!応用

- * validation を設定してみる。
- * has_one, has_many 関連を設定してみる。
- * Optimistic locking: 409 Conflict response を設定してみる。

参考: [active_resource/CHANGELOG](#)