

Ruby 講義

第13回

Sinatraで作る簡単Webアプリ

五十嵐邦明

twitter : igaiga555

<http://www.facebook.com/kuniaki.igarashi>



2013.7.11 at 一橋大学
ニフティ株式会社寄附講義
社会科学における情報技術と
コンテンツ作成III

五十嵐邦明 講師 株式会社万葉



twitter: igaiga555

<https://github.com/igaiga/>

<http://www.facebook.com/kuniaki.igarashi>

濱崎 健吾

Teaching Assistant
fluxflex, inc(米国法人)



twitter: hmsk

<https://github.com/hmsk/>

<http://www.facebook.com/hamachang>

先週の復習

目次

Gem

—
Fragments from
my work box
With love from
Audrey

Gem

Rubyのライブラリ管理システム

Gem

ライブラリ：
他のプログラムから読み込んで利用するためのプログラム

たくさんの便利なライブラリがネット上に公開されています。それらをインストールするなど、管理してくれる仕組みがGemです。

Gemの例

- Excelファイルを読み書きする
 - twitterなど有名なWebサービスへのアクセスを簡単にしてくれる
 - 画像をリサイズしたりエフェクトをかける
- などなど数万種類のライブラリが公開されています

**710,437,847
downloads**

of 40,705 gems cut since July 2009

Welcome to your community RubyGem h

Find your gems easier, publish them faster, and have

do

LEARN

Install RubyGems 1.8.24

Ruby's premier packaging system

Browse the Guides

In depth explanations, tutorials, and references

Gem Specification

Your gem's interface to the world

SHARE

gem update --system

Update to the latest RubyGems version

gem build foo.gemspec

Build your gem

gem push foo-1.0.0.gem

Deploy your gem instantly

gemを紹介、管理しているサイト
RubyGems.org is the Ruby community's gem hosting service. Instantly publish your gems and install them. Use the API to interact and find out more information about available gems. Become a contributor and enhance the site with your own changes.

<http://rubygems.org>

Gemの使い方

gemパッケージのインストール

1. shell からgemコマンドを使ってインストール

```
$ gem install gem名
```

2. コードで require "gem名"

```
require "gem名"  
# ここでgem を使うコードを書きます。
```

Gemの使い方

インストール済のgemパッケージ一覧を表示

shell からgem list コマンドで一覧表示

```
$ gem list
```

ここから今週の内容

Sinatra

Put this in
your pipe

[README](#)
[DOCUMENTATION](#)
[CONTRIBUTE](#)
[CODE](#)
[CREW](#)
[ABOUT](#)

```
require 'sinatra'  
get '/hi' do  
  "Hello World!"  
end
```

and smoke it
Sinatra gem で作る簡単Webアプリ

Webアプリって？

ブラウザでアクセス
できるアプリ



餅 せんざい 七草粥 おしるこ 黒豆

[検索](#)
[人気検索](#)

 クックパッドへ
ようこそ

 みんなのオススメ
レシピ

 あなたのレシピを
のせる

[クックパッドID \(無料\) を登録する](#) | [ログイン](#)

毎日の料理を楽しみに クックパッド

今日はパパの休肝日！出汁にこだわったお味噌汁をご用意♪



本日のピックアップレシピ

[残ったおせち活用☆黒豆きなこシフォン](#) by ヒメとそちゃん

黒豆の和風な甘さときなこの香りたっぷりのしっとりシフォンです☆...

[土 七草粥で今年も健康な一...](#) by momorecipe

[金 *ダシが決め手！うちの...](#) by shokenママ

[木 和風豆腐ハンバーグ*～...](#) by *夏海*

[水 鶏肉と野菜のポン酢煮込...](#) by rinocojp

[火 かぶのほっこり*和風力...](#) by patisier

[月 母が作る*父のふるさと...](#) by kaopomme♪

話題のレシピ

つくれば10人おめでとう！

・[豚バラかたまり☆簡単！...](#) 20:38

by kana89626

・[フライドポテト風餃...](#) 20:21

by ぢる助

・[*可愛いマイメロちゃん...](#) 19:43

by ブラック☆ラビ

・[簡単♪ヘルシー♪お豆腐...](#) 19:37

by ぶりぶりぶりん

・[【キャラ弁*可愛いハー...](#) 19:04

by ブラック☆ラビ

・[豆腐と人参の卵炒め \(簡...](#) 18:37

by kuro羊

・[あられ大根と梅の納豆餅](#) 18:23

by カルン

[バックナンバー»](#)

受賞レシピが

はなまる×クリッキー
料理選手権



我が家家のNo.1おかず

この広告を評価してください

イーモバイルが月3,217円で

@niftyの最大42M高速ポケット
1/31迄！今だけ12,000円キャッシュバック

www.nifty.com

"買取王子"楽チン買取

電話一本で自宅まで集荷にお伺い
で色々なものを宅配買取

www.kaitoriouji.jp



お夜食にぴったり♪



手軽でかわいい！



肉と野菜をバランスよく



あっという間に一品完成



お店級♪



あつあつ♪



レシピ



特集！

大根レシピ

COOKPAD

ただの片づけ術ではあります
モノへの執着を捨て、心もスッキリ
され
Yamada

Rails製

自宅でメガネを試してから買う 5本まで無料でお試しできます

くわしく

閉じる

知的に働く
メガネこの5本を
まとめて試
すヴィヴィッドムーン VM-
11356-100
25,000円ブルガリ BV-1034T-
4047
52,500円コーチ FRANCESCA
30,450円ディオール CD3759-
PDE
26,000円アミバリ TR372-14
27,300円デキるビジ
ネスマンこの5本を
まとめて試
す元 GEN-50-10
33,000円美 SHAPE SN-002-1
23,000円ジーパーツ Z-35
23,100円元 GEN-50-14
33,000円美 SHAPE SN-002-2
23,000円個性派クリ
エイティブこの5本を
まとめて試
すアランミクリ AL1053-
0202
42,000円トラクション GEMINI
Tourmaline
32,550円アクセル RIVA235
34,650円プラトイ TeA-2
31,500円59 ヒステリック
CHAPLIN-gold
21,000円凛々フェミ
ニンこの5本を
まとめて試
すブルガリ BV-1034T-
4029
52,500円ティアラ TIA-46A-
8124
31,650円シルク S-1041-103
12,125円シルク S-1041-103
26,750円

Rails製

Oh my Glasses

Twitterへようこそ
周りの人々や、興味のある企業や団体などの組織
で、今何が起きているのかを見つけましょう。

ユーザー名、またはメールアドレス

パスワード

保存する・パスワードを忘れた場合はこちら

Twitter始めませんか？登録する

名前

メールアドレス

パスワード

では、早速ですが、
Webアプリをつくっ
てみましょう

sinatra gem を使ったWebアプリ作り演習(/)

1.spreadsheets gem をインストールします。

\$ gem install sinatra

2.以下のコードを書いて好きな名前で保存します。 (例えばserver.rb)

```
require "sinatra"
```

```
get "/" do  
  "Hello world!"  
end
```

3.実行します。右下のような表示が出ます。

\$ ruby server.rb

```
== Sinatra/1.4.2 has taken the stage on 4567 for  
development with backup from Thin  
-> Thin web server (v1.5.1 codename Straight Razor)  
-> Maximum connections set to 1024  
-> Listening on localhost:4567, CTRL+C to stop
```

4.ブラウザから以下のURLへアクセスします。

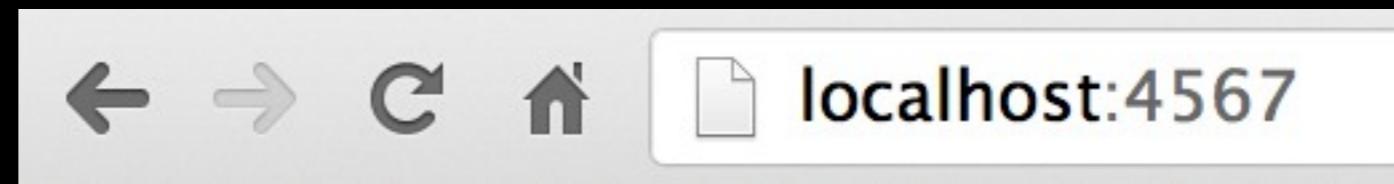
<http://localhost:4567> "Hello world!"と表示されれば成功です。

※コードを変更した場合は3.で実行しているshellを Ctrl+c で停止させ、再実行します。
(Win環境で終了しない場合、Ctrl+PAUSEを試してください。)

Webアプリの仕組みの 簡単な説明

Webアプリの基本動作

ブラウザにURLを
入力してEnter



リクエスト

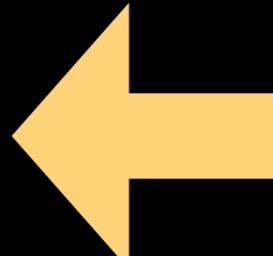
URL : `http://localhost:4567/`

Web Server

Web App

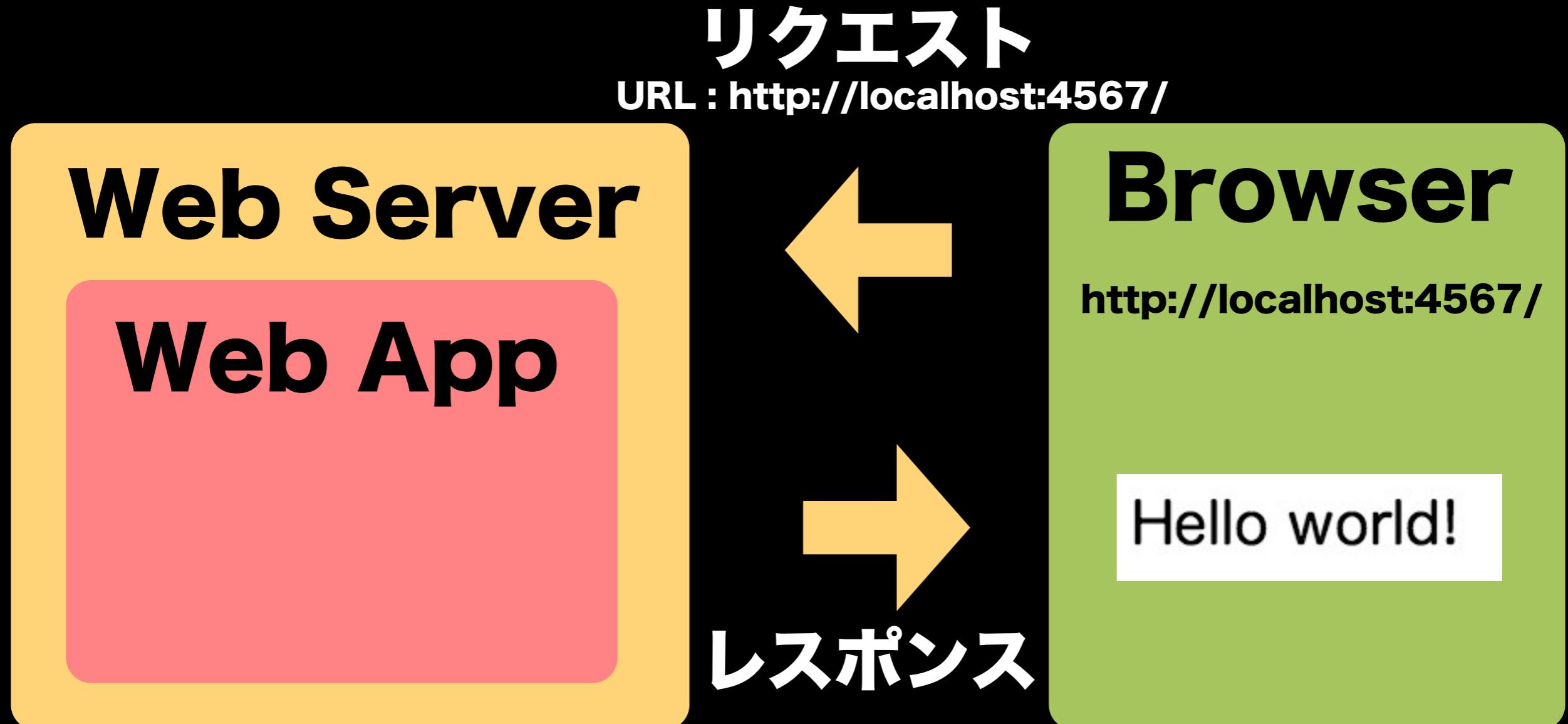
Browser

`http://localhost:4567/`



リクエストが飛びます。

Webアプリの基本動作



レスポンスが返ってくる

レスポンスを
ブラウザが表示

インターネットへのアクセス

Internet

Web Server

Web App

リクエスト

URL :
<http://cookpad.com/>



レスポンス

Browser

<http://cookpad.com/>



WebServerは通常はネットの向こうにある

どのWeb Server へアクセスするかは リクエストのURLで決まる

Internet

Web Server

Web App

リクエスト

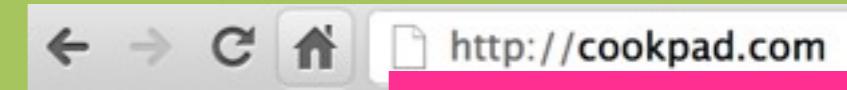
URL :

<http://cookpad.com/>



レスポンス

Browser

 http://cookpad.com



リクエストのURLが <http://cookpad.com> の場合は、
クックパッドさんの運営するWeb Server へ繋がります。
URLはインターネット上の住所みたいなものです。

先ほど作ったアプリの場合は
どうなっているかと言うと

URL localhost は自分のマシン localhost は自分のPCを表す特別なアドレスです。

自分のPC

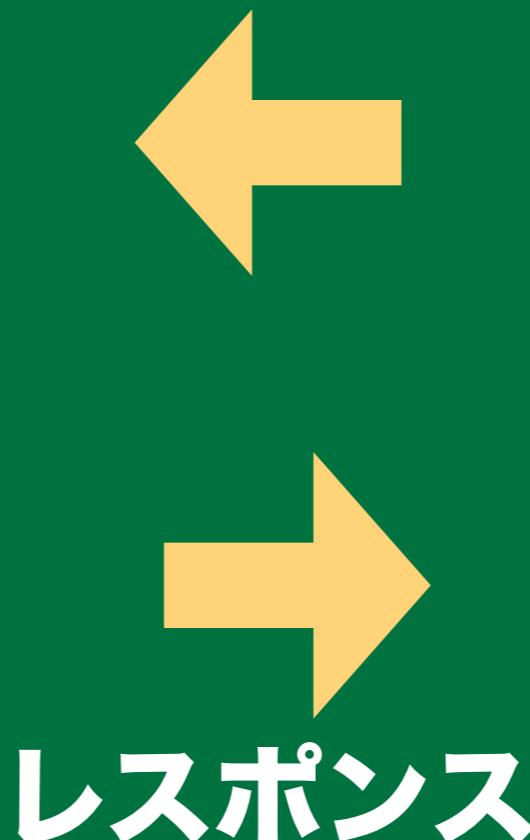
Web Server

Web App

```
require "sinatra"  
  
get "/" do  
  "Hello world!"  
end
```

リクエスト

URL : <http://localhost:4567/>



Browser

localhost:4567

Hello world!

Web App(アプリ)の部分がみんなさんが書いたコードです。
(Web Server はsinatra とRubyがうまいことやってくれます。)

では、URLの:より後ろの 4567 とは何でしょうか？

URLの説明

http://localhost:4567

アドレス ポート

URLの : より後ろはポートを表します。

ポートは、同じアドレス内で通信先を特定する番号です。

例えば、アドレスが駅の名前、ポートが駅のホーム〇番線、
とイメージすると分かりやすいかもしれません。

localhost :

特別なアドレスです。自分のマシンを指します。

4567 :

ポート番号です。4567番ポートを指します。

先ほど作ったWebアプリの動作概略



リクエスト
URL : <http://localhost:4567/>



次はWebアプリの動きを詳しく見ていきます

Web Server

localhost 4567番ポートに来た
リクエストをWebAppへ届けます

Web App

URLのアドレスとポートより
後ろの部分をパスと言います。
この場合、パスは / になります。

```
require "sinatra"

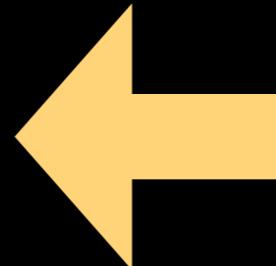
get "/" do
  "Hello world!"
end
```

パスが / のリクエストに対して
"Hello world!"というレスポンス
を返すWebアプリとして動作します

Webアプリの動作

リクエスト

URL : <http://localhost:4567/>



Browser

localhost:4567

Hello world!

受け取ったレスポンスを表示

レスポンス

次は日本語を表示させる演習です。

sinatra gem を使ったWebアプリ作り演習(/jp)

1.前の手順で実行したWeb サーバーが起動している場合は Ctrl+c で停止させます。

(Win環境で終了しない場合、Ctrl+PAUSEを試してください。)

※コードを書き換えた場合も一度停止→実行する必要があります。

2.以下のコードを書いて好きな名前で保存します。(jp.rbとします)

```
# coding: utf-8
require "sinatra"

get "/jp" do
  "こんにちは！"
end
```

(ファイル名は、どんな名前でも構いません。別の名前で保存しても大丈夫です。)

3.実行します。

```
$ ruby jp.rb
```

```
== Sinatra/1.4.2 has taken the stage on 4567 for
development with backup from Thin
>> Thin web server (v1.5.1 codename Straight Razor)
>> Maximum connections set to 1024
>> Listening on localhost:4567, CTRL+C to stop
```

4.ブラウザから以下のURLへアクセスします。

<http://localhost:4567/jp> "こんにちは！"と表示されれば成功です。

sinatra gem を使ったWebアプリ作り演習(/jp)解説

get の後に書く文字列のリクエストURLパスに応答します。

```
# coding: utf-8
require "sinatra"

get "/jp" do
  "こんにちは！"
end
```

このコードの場合は
URL: **http://localhost:4567/jp**
(パスの部分が "/jp")
にアクセスされた場合に
"こんにちは！"
をレスポンスとして返します。

次は現在時刻を表示させる演習です。

sinatra gem を使ったWebアプリ作り演習(/time)

現在時刻を表示させるWebアプリを作つてみましょう。

1.前の手順で実行したWeb サーバーが起動している場合は Ctrl+C で停止させます。

(Win環境で終了しない場合、Ctrl+PAUSEを試してください。)

※コードを書き換えた場合も一度停止→実行する必要があります。

2.以下のコードを書いて好きな名前で保存します。 (time.rbとします。)

```
require "sinatra"
```

```
get "/time" do
  Time.now.to_s
end
```

3.実行します。

```
$ ruby time.rb
```

```
== Sinatra/1.4.2 has taken the stage on 4567 for
development with backup from Thin
>> Thin web server (v1.5.1 codename Straight Razor)
>> Maximum connections set to 1024
>> Listening on localhost:4567, CTRL+C to stop
```

4.ブラウザから以下のURLへアクセスします。

<http://localhost:4567/time>

2013-06-30 11:25:17 +0900

のように現在時刻が表示されれば成功です。 36

sinatra gem を使ったWebアプリ作り演習(/time)解説

この課題から、`Time.now` のようなRubyコードを実行した結果を
ブラウザで表示する方法を学べます。

```
require "sinatra"  
  
get "/time" do  
  Time.now.to_s  
end
```

`Time.now`は現在時刻を**Time class**オブジェクトで返します。
`to_s` メソッドを使うと、**String class**オブジェクトへ変換されます。

Stringオブジェクトであればブラウザへ表示できるので、
`to_s` メソッドなどで変換することで処理結果を表示することができます。

また、`Time.now` はリクエストごとに実行されるので、
リロードするごとにその時点の現在時刻が表示されます。

次はアクセスしたパスを
表示させる演習です。

sinatra gem を使ったWebアプリ作り演習(パス)

アクセスしたパスを表示させるWebアプリを作つてみましょう。

1.前の手順で実行したWeb サーバーが起動している場合は Ctrl+C で停止させます。

(Win環境で終了しない場合、Ctrl+PAUSEを試してください。)

※コードを書き換えた場合も一度停止→実行する必要があります。

2.以下のコードを書いて好きな名前で保存します。(path.rbとします。)

```
require 'sinatra'
```

```
get "/:word" do
  params[:word]
end
```

3.実行します。

```
$ ruby time.rb
```

```
== Sinatra/1.4.2 has taken the stage on 4567 for
development with backup from Thin
>> Thin web server (v1.5.1 codename Straight Razor)
>> Maximum connections set to 1024
>> Listening on localhost:4567, CTRL+C to stop
```

4.ブラウザから以下のURLへアクセスします。

<http://localhost:4567/hi>

/hi というパスにアクセスされればhiが、
/book というパスなら book が表示されます。

sinatra gem を使ったWebアプリ作り演習(パス)解説

get 後の文字列の中に :word のようにシンボルを書くと、リクエストされたパスを params[:word] で受け取れます。

```
require 'sinatra'  
  
get "/:word" do  
  params[:word]  
end
```

http://localhost:4567/hi

上記のURLでアクセスした場合、
params[:word] は "hi" を返します。

※ここでの :word は任意の名前にできます。
get ":foo" do ~ と書いた場合は
params[:foo] でパスを取れます。

これを応用すると、例えば血液型占いなど、
アクセスされたURLに応じて表示を変えるアプリを作れます。
時間がある人はチャレンジしてみてください。

參考資料

書式

Rubyコード

`puts "abc"`

実行結果

"abc"

実行結果

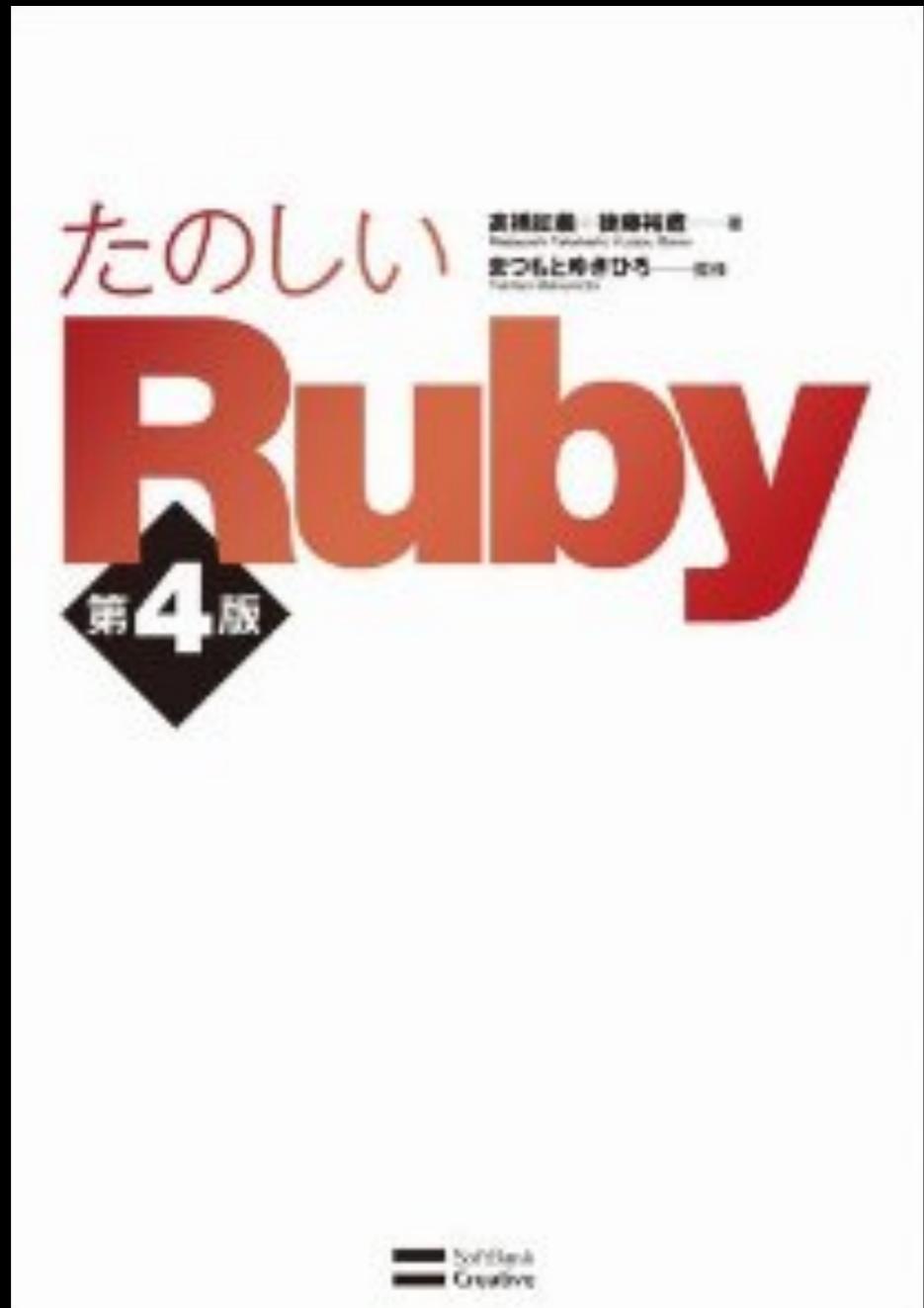
文中では `#=>` で書きます

`p 1+2 #=> 3`

shellコマンド

`$ ls`

教科書：たのしいRuby



<http://www.amazon.co.jp/dp/4797372273/>



お買い求めは
大学生協または
ジュンク堂池袋店で

課題チェック用の付箋の書き方

上の方に学籍番号と名前を書いてください

学籍番号

名前

※この辺に講師陣がクリアした課題番号を書いていきます。

講義資料置き場

過去の資料がDLできます。

<https://github.com/igaiga/hitotsubashi-ruby-2013>

雑談・質問用facebookグループ

<https://www.facebook.com/groups/hitotsubashi.rb>

- 加入/非加入は自由です
- 加入/非加入は成績に関係しません
- 参加者一覧は公開されます
- 参加者はスタッフ(講師・TA)と昨年、今年の受講者です
- 書き込みは参加者のみ見えます
- 希望者はアクセスして参加申請してください
- 雑談、質問、議論など何でも気にせずどうぞ~
- 質問に答えられる人は答えてあげてください
- 講師陣もお答えします
- 入ったら軽く自己紹介おねがいします