

## Webプログラミング 第1回 講義概要、Rubyの基本

永森光晴  
nagamori@slis.tsukuba.ac.jp

研究室は春日研究棟412号室

### 講義の概要

- Webアプリケーションの作り方の基礎を学ぶ
  - CGI や DB などの基本が中心
  - Javascript, 正規表現
  - FlashやJava サブレットなどは扱わない
- RubyやHTMLといくつかの道具(ソフトウェア)を組み合わせて実現
- 1コマ目解説、2コマ目演習、課題を提出

### Webアプリケーション

- Webブラウザを使った動的コンテンツの生成
- 例えば、検索システム、ウェブメール、ブログ、掲示板、チャット、ゲーム等々

```

    graph LR
      Browser[Webブラウザ] -- URL --> Server[Webサーバ]
      Server <--> Program[プログラム]
      Program <--> DB[(DB)]
      Program <--> Files[ファイル]
  
```

### 講義の予定

• 10/4 概要、Ruby入門	• 11/15 Javascript
• 10/11 CGIの基礎(1)	• 11/22 Ruby on Rails (Webアプリケーションフレームワーク)
• 10/18 CGIの基礎(2)	• 11/29 (休講)
• 10/25 DBの利用 (SQLite3)	• 12/6 まとめ、レポート解説
• 11/1 ERB, セッション, Cookie	• 12/13 (予備)
• 11/8 正規表現、セキュリティ	

*Embedded Ruby*

### 成績評価

- 出席、課題の提出、最終レポートによって評価する

### 授業に関する情報

<http://www.kc.tsukuba.ac.jp/~nagamori/wp/>

### 参考図書

- まつもとゆきひろ、オブジェクト指向スクリプト言語Ruby、ISBN4756132545
- 高橋征義、たのしいRuby第3版、ISBN4797357401
- 青木峰郎、Rubyレシピブック第3版、ISBN4797359986

## 今日の演習

- 授業のための準備およびRubyスクリプトの作成と実行をする(以降で解説)
- 「Rubyの基礎を固めよう」を、文中の例を動かしながら読む
- 今回は以下のセクションは後から構わない
  - パターンマッチング
  - メソッドに渡すブロック

## 授業のための準備

1. Ubuntu にログインする
2. ターミナルを開く
3. 授業用ファイルを置くディレクトリを作る

```
mkdir -p ~/www_local/wp
```

4. エディタ(emacsやviなど)を使って、~/www\_local/wpの下にtest.htmlという名前のHTML文書を作成する(中身は何でもよい)

5. ブラウザから表示できることを確認

```
http://edu.mast.tsukuba.ac.jp:8080/~自分のID/wp/test.html
```

## Ruby スクリプトの基本(CGIではない)

```
~/www_local/wp/sample1-1.rb
```

```
#!/usr/bin/env ruby
print "Hello World!¥n"} 処理内容を記述する
```

- 1行目の「#!/usr/bin...」はお約束(この授業では必ず記述する)
- エディタは何を使ってもよいが、プレーンテキスト形式で保存する
- ファイル名の拡張子には .rb を付ける

## マジックコメント

- スクリプトの中で日本語を使う場合にはマジックコメントを入れる

```
#!/usr/bin/env ruby
# encoding: utf-8
print "こんにちは!¥n"
```

- 以下のエラーが出る場合はマジックコメントを入れる

```
% ruby hello.rb
hello.rb:3: invalid multibyte char (US-ASCII)
hello.rb:3: invalid multibyte char (US-ASCII)
```

## まずはターミナルでRubyスクリプトを実行

- cdコマンドで~/www\_local/wpに移動
- 以下のコマンドでファイルに実行属性を付与
 

```
chmod a+x sample1-1.rb
```
- ls -lでファイルの実行属性が「-rwxr-xr-x」が付いているか確認する
- ターミナルでRubyスクリプトを実行
 

```
./sample1-1.rb
```

 または
 

```
ruby sample1-1.rb
```

## irbコマンド

- Rubyスクリプトを対話的に実行するにはirbコマンドを利用する
- 終了はcontrol+dかexitと入力する

<http://doc.ruby-lang.org/ja/1.9.3/library/irb.html>

```
irb
irb(main):001:0> x = "Hello world!"
=> "Hello world!"
irb(main):002:0> print x.length
12=> nil
irb(main):003:0> exit
```

### C言語との大きな違い

- Rubyはインタプリタ →コンパイルは不要
- 変数宣言は不要
- 文は改行かセミコロン(;)で終わる
- すべてがオブジェクト
- 例外処理  
...などなど

### メソッド呼び出し

- Rubyで扱うものはすべてオブジェクト  
– 文字列、数字、配列、etc...
- オブジェクトに定義されているメソッド(操作)を呼び出すと結果が返ってくる

オブジェクト.メソッド名 (p1, p2, ...)

"hello world!".length → 12

"hello world!".upcase → "HELLO WORLD!"

[1, 2, 4, 8].length → 4

[1, 2, 4, 8].first → 1

### クラスとインスタンス(p.32)

- オブジェクトの種類をクラスと呼ぶ
- あるクラスから生成されたオブジェクトを、そのクラスのインスタンスと呼ぶ
- C言語の構造体のように、独自のクラスを定義することができる
  - 例は、人に関する情報を扱う Person クラス
  - Personクラスのインスタンスは、鈴木さんや佐藤さんといった具体的な人(Person)

### Rubyでの記述(プリントp.33)

```
class Person
  def initialize(name, addr, tel)
    @name = name
    @addr = addr
    @tel = tel
  end
  def get_name
    return @name
  end
end
```

```
suzuki = Person.new("鈴木太郎", "大阪市", "06..")
satou = Person.new("佐藤花子", "西宮市", "0798...")
puts suzuki.get_name
puts satou.get_name
```

鈴木太郎  
佐藤花子

実行結果

### 今日の演習

- 授業のための準備およびRubyスクリプトの作成と実行をする(以降で解説)
- 「Rubyの基礎を固めよう」を、文中の例を動かしながら読む
- 今回は以下のセクションは後から構わない
  - パターンマッチング
  - メソッドに渡すブロック

### 注意

- 文字化けをする場合は、ファイルの文字コードが UTF-8 になっているか確認する
- 日本語を使う場合マジックコメントを入れる  
#!/usr/bin/env ruby  
# encoding: utf-8
- ヒアドキュメント(p.46,47)を使う場合、「<<」の前(左)には空白を入れる

```
print <<EOF
```