

# Ruby 初級者向けレッスン第 18 回

okkez @ Ruby 関西

2008 年 03 月 15 日

## 今回の内容

- 文字列処理とは
- 文字列処理の実用例
- 文字列処理の基礎
- 演習 (group work)
- まとめ

## 今回のゴール

- Ruby による文字列処理の便利さを実感する

## 文字列処理とは

- 読んで字の如く、文字列を処理すること
- 検索したり、置換したりする

## 文字列処理の実用例

- ログ解析
- テキスト解析
- テキスト置換
- HTML 解析
- コードジェネレータ
- などなど

## Ruby でのアプリケーション例

- RTools
- RD 形式で記述された文書をさまざまなフォーマットに変換する
- この資料も元は RD で記述して TeX -> dvi -> pdf に変換している
- RDoc
- Ruby のソースコードに埋め込まれているドキュメントのフォーマット
- HikiDoc
  - Wiki クローンの一つである Hiki で使われているライブラリ
- rak
  - 最近、流行の Ruby による grep の実装。
  - 使い始めるとかなり便利

## 文字列処理の基礎

- よく使われるクラス
- よく使われるイディオム
- よく使われるライブラリ

## よく使われるクラス

- String
  - 文字列のクラス
- Regexp
  - 正規表現のクラス
- IO (またはそのサブクラス)
  - 入出力を扱うクラス

## よく使われるイディオム

```
case str
when /aaa/
  # 何か処理
when /bbb/
  # 何か処理
when /ccc/
  # 何か処理
end

# 行ごとに処理を行い、条件を満たした所で終了する
lines.each do |line|
  break if /aaa/ =~ line
  # 何か処理
end

# arr に含まれる要素の出現頻度をカウントする
hash = Hash.new{|h, k| h[k] = 1 }
arr.each{|elem| hash[elem] += 1}
```

## よく使われるライブラリ

### CSV

- 標準添付
- CSV ファイルを処理する
- かなり遅い
- 速い実装はサードパーティ製として存在

### ReXML

- 標準添付
- XML を解析したり組み立てたりする
- 遅い

### strscan

- 標準添付
- 高速に文字列を解析することができる

## yaml

- 標準添付
- YAML 形式のファイル进行处理する
- Ruby のオブジェクトを子の形式でダンプすることもある

## win32ole

- Windows 上でごにょごにょするのに便利
- Windows 上で Excel ファイルや Word ファイルから文字列を抜き出したり
- Excel ファイルや Word ファイルを作成したりできる
- 他にも用途はいろいろ

## hpricot

- HTML の解析を行うライブラリ
- 使い方が独特
- gem でインストールできる

## 演習問題

### テキスト解析

ある英文を解析してみましょう。

- 文字数
- 単語数
- 行数
- 文字別の出現頻度
- 単語別出現頻度

## ログ解析

Apache のログを解析してみましょう。

- '/index.php' へのアクセスは何回？
- 最初の Mac ユーザのアクセスはいつ？
- Google 経由のアクセスは何回？
- 何曜日のアクセスが一番多い？
- ブラウザごとのアクセス数ランキングは？

## ヒント

Apache のログの例 (実際は一行)

```
10.136.161.73 - - [26/Jun/2005:05:48:37 +0900]
"GET /20040414.html HTTP/1.0" 200 2240 "-"
"DoCoMo/1.0/P505iS/c20/TB/W20H10"
```

前から順に、IP アドレス、(略)、ユーザ、時刻、メソッド、URI、プロトコル、ステータス、リンク元、ブラウザ

詳しくは Apache のマニュアルを参照してください。

使用するとよいかもしれないメソッド

- IO.open
- IO#close
- IO#readlines
- IO#each
- Hash.new
- String#split
- String#gsub
- Array#<<
- Array#push
- Hash#[]=
- Regexp#=~
- Regexp#===
- などなどこれ以外にも多数あると思います

## まとめ

- String, Array, Regexp などすべてのメソッドのマニュアルを読むべし
- 自分の PC にリファレンスマニュアルをインストールしておくべし
- 最初はシンプルに徐々に複雑にしよう

## 今後の情報源

### 公式 Web サイト

<http://www.ruby-lang.org/>

### Ruby リファレンスマニュアル刷新計画

<http://doc.loveruby.net/refm/api/>

### 日本 Ruby の会

<http://jp.rubyist.net/>

### Rubyist Magazine

<http://jp.rubyist.net/magazine/>

### okkez weblog

<http://typo.okkez.net/>