



*PHPのレガシーコードを  
Rubyで焼き変えてみた*

@akira345  
*Hiroshima.rb*

# 自己紹介



- 本業：精神修行
- 所属：
  - Hiroshima.rb
  - 広島サーバユーザ友の会（仮）
  - 広島電子工作倶楽部（仮称）
  - WindowsPhoneハンズオンin広島
  - オープンセミナー2014@広島 実行メンバー
  - OSC広島実行メンバー
  - WTM時々インフラ担当

# 本題



- 目的:

非オブジェクト指向なレガシーコードを  
(出来る範囲で) オブジェクト指向的に  
持っていく。

出来ればテストも入れたい。

# 生け贄の準備



- 今回の生け贄:

- 十数年前に独学で習得中に自作した  
15Puzzleプログラム

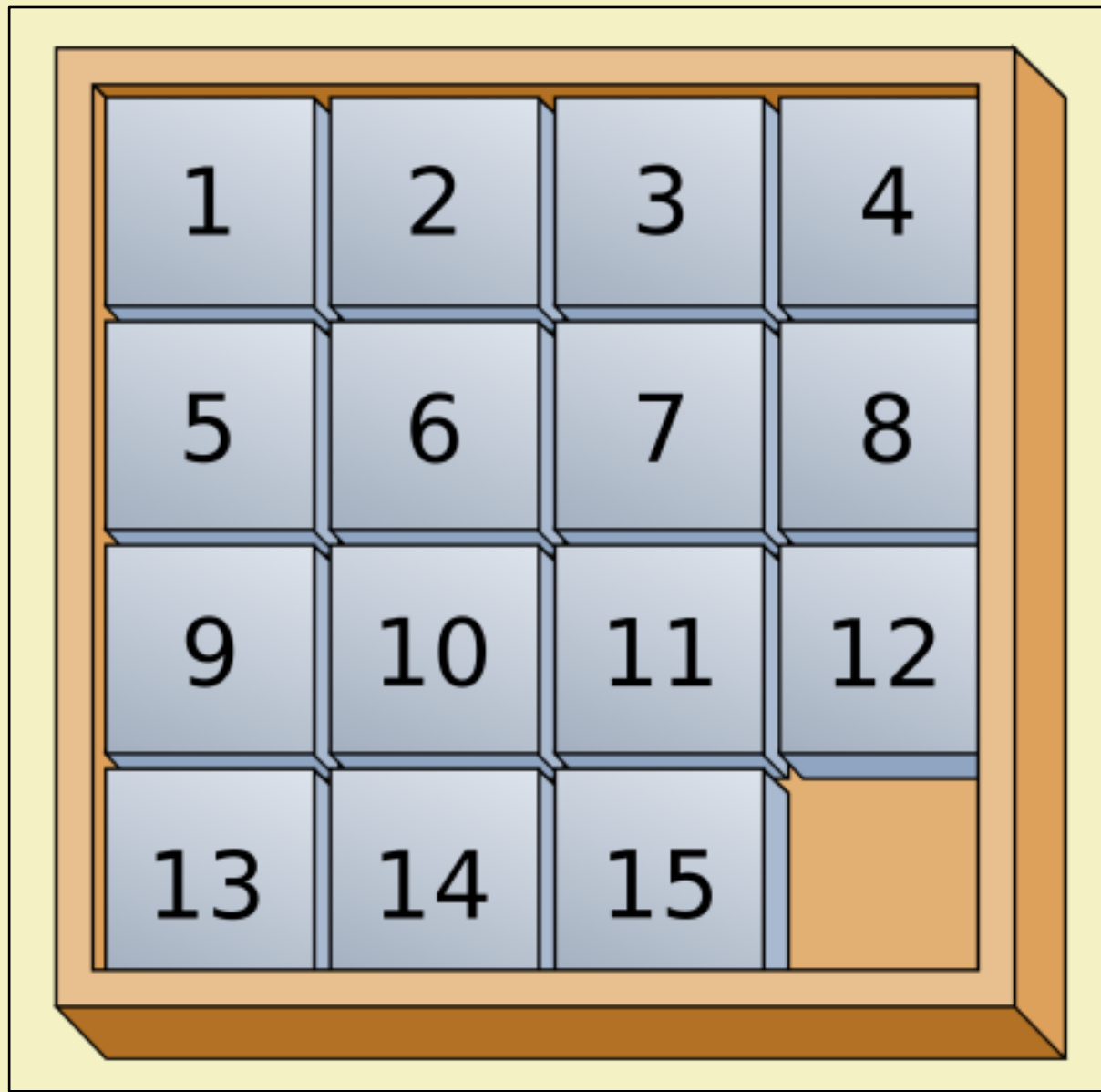
[https://github.com/akira345/Old\\_PHP\\_Samples/tree/master/15puzzle\\_old](https://github.com/akira345/Old_PHP_Samples/tree/master/15puzzle_old)

(以前Rubyでゲームを作られた方がいたので対抗してみた)

# 15Puzzle??



# 15Puzzle??



# レガシーな点



- 圧倒的に**古い!** (iniファイルなんてものがあったり)
- php4の出た頃の頃に書かれている
- 作った本人が**実はよく分かっていない**
- 山のような**フラグ管理**。よくわからない**変数**、**強引な処理**
- **変数一覧表**なるものがあるw
- 当然MVCなんて考えは**ない**
- 当時のポリシーとして、HTMLを極力PHPで出力しているので**作りがカオス**
- 当時自信作今黒歴史

# 第1段階



- いきなりRubyにするのはさすがに無理なので、まずはPHPで試作。
  - とはいえ、再設計すると、新規で作り直s(ry
- 当時の自分を恨みつつ、まずは簡単なビューとロジックの分離から。
  - この時、無理にファイルを分離しようとするとは色々悩むので、まずは同一ファイル内で、分離すると良い。
- ビューに埋め込んだロジックはなるべく分離。
  - 但し、この時はロジックを余り弄らないこと。テストが無いのでバグると酷い目に合う



# 第2段階



- ロジックの組換前に、機能ブロックを大まかに洗い出しておく。
- ロジック部分から、機能ブロックをくりだし、クラスメソッドとして纏める所からまずはスタート。
- 関数化されている箇所はそのままクラスメソッドとして仮実装。
- クラスメソッドはとりあえず全てPublic!
- テストコードが無いので、ロジックの組み換えは後。動く状態を極力キープ

# 機能ブロック



- このプログラムを構成する主な機能
  - 乱数生成
  - 動かす数字を格納する配列を生成
  - 数字が動かせるか判定する
  - 数字を動かす
  - パズルが完成したかチェックする

# ロジックに手をつける



- フローチャートを書くのは断念したOrz
- 基本は機能ロジックごとにカットアンドペーストでクラスメソッド化や、ロジックの組み換えを行った。
- 機能ブロックを動かす場合、該当箇所を全てコメントアウトし、最低限の実装に置換。動作確認しながら、ブロックの組み換えを行った。
- 複雑怪奇なifのネストは、中のロジックをprint('A')のようなものに書き換えて、動作の確認をしながら整理
- 変数名やフラグを少し整理。(fとかswとかrとか)

# セッション周り



- 昔はセッション変数に何でも突っ込んでいた
- オブジェクト指向になると、クラスにデータを持たせられるので、クラスをセッションに突っ込めばOK(時と場合による)

# 完成

---



[https://github.com/  
akira345/15Puzzle\\_For\\_OOP](https://github.com/akira345/15Puzzle_For_OOP)

ただしPHP版

# さて、**Ruby**に焼き変えるか。



- PHPをRubyに置き換えるので**楽勝!**(と思っていました)
- いまさらCGIクラスなんて使うと**レガシーに戻る。**  
かといって、この程度でRailsも**大げさ**
  - **Sinatra**を使おう。

<http://www.sinatrarb.com/intro-jp.html>

# sinatraではまる!



- あれ?画像でないぞ??
  - publicディレクトリが必要
- あれ?セッションに入れたクラスが読めないぞ?
  - enable :sessionsが必要
- get/postでロジック分離しないといけないぞ?
  - とりあえず共通関数作成で回避<-邪道
- などなど

# ひむひむへルプ!



- 絶賛**仕事**中のひむひむへ**襲撃**!!
  - インデントがカオス!
  - status=3って何?
  - このメソッド名、スペル違うよ...
  - 変数の初期化はどこでされているの?
  - ロジックが難しい。データの持たせ方が...
  - テストはどこ?
  - やりたい事は何となく分かるけど...う〜ん(悩)





---

教訓：コードレビューは大事。



# 教訓：コードレビューは大事。

- 大事な事なので(ry

# リファクタリングだ！



- クラスのテストをRspecで。
  - だけど、ランダムにデータができるので、余りテストが書けなかった・・
- ビューのテストが非常に難儀
  - sinatraでは、途中でprintとか出力しても無視されるので、printデバックができない。
  - pryをつかって、ブレークポイントを仕掛けてデバック
  - <http://shirusu-ni-tarazu.hatenablog.jp/entry/2012/06/24/051114>

# そんなこんなで

---



## 完成!(デモするよ)

🔴 [https://github.com/akira345/15Puzzle\\_For\\_Ruby](https://github.com/akira345/15Puzzle_For_Ruby)