

AzureStone (あーじゅ・すとーん)

<http://www.azurestone.org/>

# 脱KENT様方式

— 読書百編 義 自ずから通ず —

# 注意

- ◎ **これが正しいと主張しているわけではありません。**
- ◎ **私の経験上**の話になります。
- ◎ **FUD**な発言をしている可能性が**大い**にあります。気をつけて下さい。
- ◎ 私は、様々な人の人間性を否定しません。
- ◎ もちろん**KENT**さんの人間性を否定していません。

# 自己紹介

- ◎ AzureStone(あーじゅ・すとーん)
- ◎ 年齢
  - 現役女子高生より10歳くらい年上です…。
- ◎ 職場
  - もうすぐ退職です。
- ◎ 性別
  - ご覧のとおりだと思います。
  - **はい残念！男でした。**

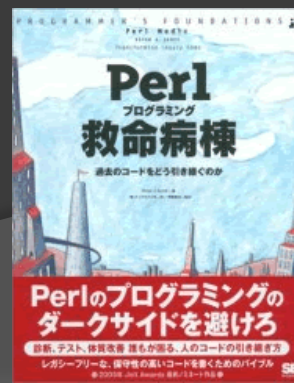
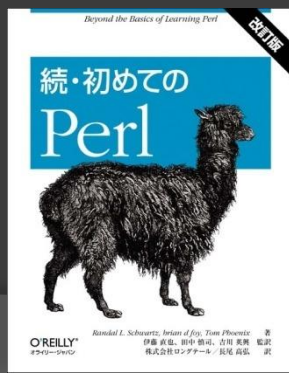
# 脱KENT様方式になるには？

## ◎ 結論

- 下記の書籍を**何回も**読む事が**鍵**です。
- 01.「初めてのPerl」
- 02.「続・初めてのPerl」
- 03.「Perl救命病棟」

## ◎ 私の場合は

- 02→01→03の順番に読んでいた。



# 自分のPerlスキルレベル

- ◎ オブジェクト指向をなんとなく理解し始めた
  - 概念は未だわからず…。
- ◎ `use warnings; use strict;` をつけるようになった
- ◎ `my`を宣言するようになった
- ◎ 継承を勉強し始めた
- ◎ `POE`でプログラムは作った事ある
- ◎ `Tripletail`を愛用しています
- ◎ CPAN Authorではない。
- ◎ `ORマッパー`がわからない
  - 一つのテーブルを一つのクラスとして扱えるぐらいしか知らない。
  - ということで未だDBIを使用しています。
  - とういか使ったことない。
  - `WEB+DB`買い損ねた…。どうにかして手に入れたい…。

# 最近の出来事(自分の事)

- ◎ **Moose**(発音:も〜〜え〜?)は、Catalystだけのためプラグインモジュールを作成するための単なるSDKみたいな物と勘違いしていた。もちろん自分のせい。
  - freenodeの#catalyst-ja内の会話が、きっかけで調べていた。。。
  - その後、id:yusukebeさんに教えて貰いました。
  - この後、はこべさんが詳しく分かりやすく皆さんに説明してくれると思います。
- ◎ Moose(発音:むーせ?)の件とってもわかるでしょ? スキルレベルが。。。

# ターゲット:いずれか

- ◎ Perlプログラミングを**生業**にされている方
- ◎ 職場で開発指定言語として**Perl**を指定
- ◎ **周り** or 自分が、KENT様方式
- ◎ **自分はKENT様方式ではない！**
- ◎ **KENT WEB**のCGIを活用している
- ◎ **新人教育**どうしよう・・・
- ◎ 職場で「**ググレカス！**」と突き返す方

アンケート

KENTさんを

ご存じの方！

?? / 40人中



# KENT様(KENTさん)

## ◎ KENT WEBの管理人

- 過去に様々なCGIを公開されている
- Perl基礎入門を公開

## ◎ 書籍多数

- <http://www.kent-web.com/pubc/book/>
- 是非購入してご覧下さい。



# KENT WEB

KentWeb  
フリーソフトで作るインタラクティブページ  
- CGI/Perlプログラム -

検索

PR> KentWebがプロデュースするレンタルサーバ【KentNet】  
PR> 失敗しないレンタルサーバの選び方

累計: 156,512,134  
本日: 87,521  
昨日: 190,271

- CGIプログラム -

- 掲示板
- チャット
- カウンター
- パスワード制限
- データ収集加工
- 買物カート
- パラエティ系
- ユーティリティ

- お知らせ -

- 新着情報
- 週間ダウンロードランキング
- プログラム利用規定
- フリーソフト規定
- シェアウェア規定
- サポートコーナー
- よく頂く質問(FAQ)
- 過去ログ検索

C'S SERVER 詳細はこちら  
高機能アクセス解析ツール  
SiteTracker8 標準搭載!  
【PR】レンタルサーバ C'S SERVER (Secure)

耐障害性 × 機能選択型 × 低料金  
【PR】【人気】月額315円からのレンタルサーバ

- 書籍執筆のご案内 -

- 書籍執筆インデックス
- ワンランク上のPerl/CGIプログラミング
- 最速攻略CGI/Perl即入門
- PerlでCGI(改訂版)

- リファレンス・コーナー -

- Perl基礎入門 (初心者向けPerl文法)
- 自宅マシンWWWサーバ化計画 (Windows)
- インストールの巻
- オフラインでCGIを動かさそうの巻

# KENT様方式ってなんですか？

- ◎ 主にPerlプログラミング作法の事を指します。
- ◎ つまりC (言語) のK&Rスタイルのような**コーディングスタイル**を指します。
- ◎ そのコーディングスタイルにKENTさんのスタイルが表れているのでKENT様方式(KENT様スタイル)と読んでいます。

# KENT様方式:01

- ◎ use warnings;を使用しない
- ◎ use strict;を使用しない

# KENT様方式:02

- ◎ Jcode.plを使用する
- ◎ `$ENV { 'REQUEST_METHOD' }` や `$ENV { 'CONTENT_LENGTH' }` を自前で decode する。
  - ソースコードを見る限り特別な理由があるように見えない。

# KENT様方式:03

- ◎ local変数 or global変数だらけ
- ◎ my宣言が見あたらない

# KENT様方式:04

- ◎ HTMLとロジックが同一
- ◎ HTMLテンプレート方式ではない
- ◎ また、入力値をサニタイズ(セキュリティ)

# KENT様方式:05

- ◎ 非オブジェクト指向
- ◎ つまり簡潔に言うとPerl 4スタイル
  - Perl 4をわざわざ動作させるサーバは、セキュリティ大丈夫？(OSのサポートなど)



# KENT様方式:06

- ◎ 非オブジェクト指向なのでリファレンスやデリファレンスを使用しない。
- ◎ つまりサブルーチンへは、参照渡しではなく値渡しになる。

# KENT様方式:07

- ◎ TESTコードが、見あたらない
  - 様々な環境で動作させる事を目的にしているはずなのにテストコードが見あたらない。

# KENT様方式:08

- ◎ ソースコードが、設定ファイル
  - ヒューマンエラーの危険性
  - セキュリティ的にやばい

# KENT様方式:09

## ◎ 例外処理が見あたらない

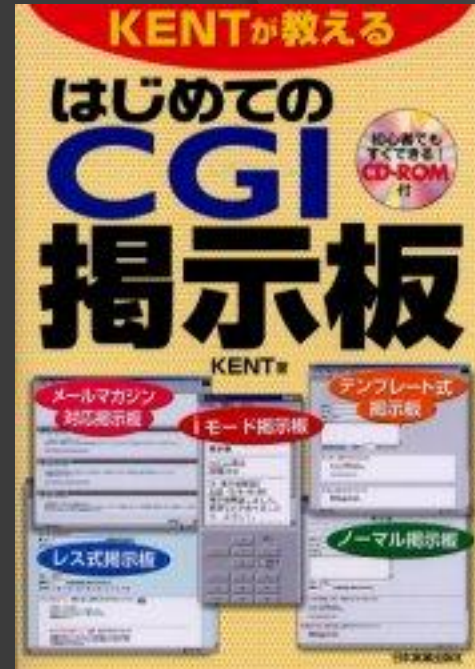
- `eval { hogehoge }; $@` が見あたらない。
- または、`die`

# KENT様方式:10

- ◎ CPANモジュールを使用しない
  - 使用する必要性がないのではなく、使用しないことを意識している。

# 自分は、KENT様方式から出発

- ◎ 一番最初に参考にした書籍
  - 「KENTが教える はじめてのCGI掲示板」
  - つまり最初は、CGIを設置していただけ。
- ◎ 読んだのは、17歳(高校生)の頃
  - Perl4→Perl5移行されている時期
  - PHP3リリース、インターシップ先では大盛り上がり



# 自分は、KENT様方式から出発

## ◎ 次に参考にした書籍

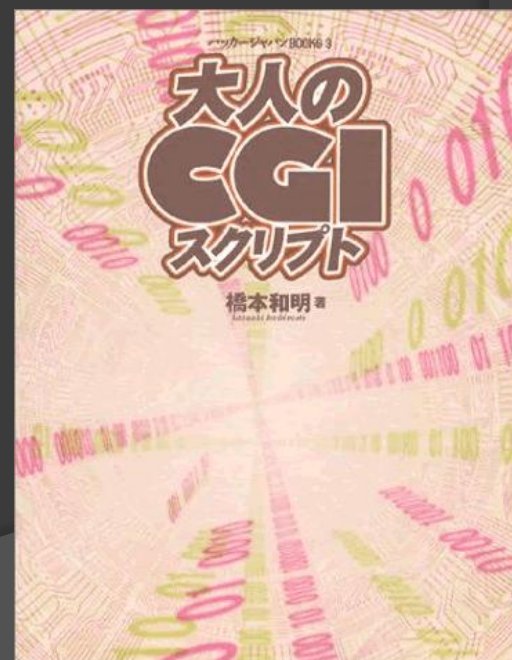
- 「Perl/CGI 逆引き大全 600の極意」
- つぎは、ソースコードを見てエディタコピペ
- Blogブーム(当時は、Weblogと言われていた?)



# 自分は、KENT様方式から出発

## ◎ 次に参考にした書籍

- 「大人のCGIスクリプト」
- つぎも、ソースコードを見てエディタに手動コピペ





# 自分は、KENT様方式から出発

## ◎ バイブル

- 「CGI&Perlポケットリファレンス」



# 自分は、KENT様方式から出発

## ◎ 当時の価値観・行動（スタイル）

- できるだけモジュールを使用しないようにする
- `cgi-lib.pl` と `jcode.pl` は持ち歩く
- 俺々ライブラリを構築しておき持ち歩く
- 独自のHTMLテンプレート作成
- `use warnings; use strict;` なにそれ？
- `my`宣言？なにそれ？

とまあそんな感じでした…。

# 脱KENT様方式のきっかけ

- ◎ **TripletaiL**が使いたかっただけ。
  - しかし、使うには様々な課題（**試練**）があった。
- ◎ **課題**
  - **リファレンス(デリファレンス)**ってなに？
  - ってかCPANシェルで**モジュールインストール**するにはどうするの？
  - え！？CPANシェルでinstall TripletaiLとすると**エラー**でインストールできない！！なんで！？
  - **オブジェクト指向**ってなに！？

# なぜフレームワークを使おうと思ったのか？

- ◎ **結果的に**Tripletailになったけど。なんだかの**フレームワーク**を使用したかった。
- ◎ 作成されたプログラムが**属人化**してしまう。
- ◎ 俺々ライブラリを保守し続けなければいけない。
- ◎ **移植性0(ゼロ)**
- ◎ デバッグは、**printデバッグ**
- ◎ **コピペの嵐**…。
  - 実はこれが一番きつい。
  - 無駄にループ(**値渡し**だから)

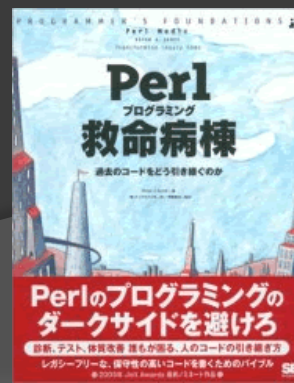
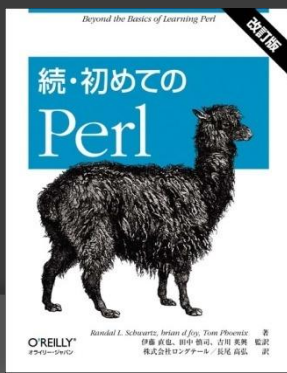
# 脱KENT様方式になるには？

## ◎ 結論

- 下記の書籍を**何回も**読む事が**鍵**です。
- 01.「初めてのPerl」
- 02.「続・初めてのPerl」
- 03.「Perl救命病棟」

## ◎ 私の場合は

- 02→01→03の順番に読んでいた。



# 重要な事

- ◎ ここにお集まりの方は、**恐らく既に脱KENT**様方式だと思います。
- ◎ **底上げの重要性:あなた以外**
  - 知っていた者勝ち? → そうかな?どうかな?
  - 全ての人間が、プログラミングに前向きではない?
  - 実は、趣味でコード書いている人間は少ない?

# 初心者って？ 中級者って？

- ◎ いったいどこからが**脱初心者**？
  - どこまで勉強すれば**初心者**から抜けられる？
  - 初心者か中級者かどうかよりも**レベルわけ**すればいいじゃない？
- ◎ Perlプログラマのレベル10 – **Perlプログラミング救命病棟**より
  - naoyaさんのblogの記事
  - <http://d.hatena.ne.jp/naoya/20050809>

# lapis25さんの愚痴

2007-08-17

■ [\[perl\]Perler Magazine創刊\(希望\)](#) ☆☆☆☆☆☆☆☆

『Perlish Magazine』は、Perl に関する技術記事はもちろんのこと、Perlish へのインタビューやエッセイ、その他をお届けするウェブ雑誌です。

<http://jp.rubyist.net/PerlMa/wiki.cgi?page=FrontPage>

みんな自分のブログに書くからなあ。発信の仕方が違うだけなんだろうが、アンテナの性能や指向性の違いで見つけにくい人たちもいる。でも、そういう人たちこそ引き上げることが必要な気がしている。KENTを笑いものにするのもいいけど、あんたらはそれに変わる何かをやっているのか？と訊きたい。高度なことをやっている先頭集団のみが加速していく状況では、先が見えない

自分はまったくもって先頭集団ではないけれど、何かやってみようか

[Permalink](#) | [コメント\(3\)](#) | [トラッキングバック\(0\)](#) | 12:33 [FB](#) [2 users](#) [☰](#)

引用：<http://d.hatena.ne.jp/lapis25/20070817/1187321598>



# fbis (エフビーアイズ) さんの例

foreach文

ハッシュ

配列操作関数push,shiftなど

packageについて

グローバル (パッケージ) 変数とレキシカル変数の違い

インクルードパスとパッケージの関係

関数とメソッド

Perlでのオブジェクト指向プログラミング

bless関数

リファレンス

strictプラグマ

~~ 略 ~~

引用:

<http://d.hatena.ne.jp/fbis/20060504>

<http://d.hatena.ne.jp/fbis/20060510/1147253289>

# まとめ

- ◎ 怒らず知らない事へんな指摘の仕方をせず  
みんなで底上げしていきましょう！
- ◎ まず、書籍を読みましょう！

# お礼

本日は、私の発表を聞いて下さって  
誠にありがとうございます。

またこの場を提供して下さった  
Kansai.pmのご協力戴いた方々  
ありがとうございます。