

いろんな言語触ってみた

# 目次

- どんなものを作ったか
- ベンチマーク測ってみた
- 解説
- 各言語の所感
- 感想
- おまけ

どんなものを作ったか

とりあえず、よくあるCSVバッチを作ってみた  
使用する言語はPHP、Go、Ruby、Python

ベンチマーク測ってみた

# サーバースペック

できるだけ汎用的、バースト系以外のインスタンスで計測

インスタ ンスサイ ズ	vCPU	メモ リ (GiB)	インスタ ンススト レージ (GiB)	ネットワ ーク帯域 幅 (Gbps)	EBS 帯 域幅 (Mbps)
m5.large	2	8	EBS の み	最大 10	最大 4,750

# 計測方法

1. 1万行(1.6MB)のCSVを読み込んでDBに登録
2. 登録したDBのデータをCSVに吐き出す
3. 入力CSVと出力CSVを突合して確認
4. 計10回実行してその平均を出す

# 実行結果

realが実際の時間。userがプログラム実行時間。sysがOSの処理時間。

順位	言語	real	user	sys
1	Go 1.15.5	7.246s	0.332s	0.323s
2	PHP 8.0.2	7.709s	0.390s	0.126s
3	Python 3.8.5	7.982s	0.648s	0.128s
4	Ruby 3.0.0	8.850s	2.022s	0.148s



• • • ホンマか？

# 解説

## 言語でそんなに速度が変わらなかった理由

- 各言語よく見ると実際の実行時間(real)が8秒くらいなのに対して、プログラムとOSの実行時間(user+sys)合わせても1秒ほどなのです。(rubyはなんか2秒いってる)
- じゃ残りの7秒はというと、ネットワークI/O、ディスクI/O、DBの処理待ちとなるようです

言語	real	user	sys	他I/O
Go 1.15.5	7.246s	0.332s	0.323s	6.591s
PHP 8.0.2	7.709s	0.390s	0.126s	7.193s
Python 3.8.5	7.982s	0.648s	0.128s	7.206s
Ruby 3.0.0	8.850s	2.022s	0.148s	6.680s

つまり今回作った自分のベンチでは、各言語の処理に1秒もかかっていなかった所以说語による差がほとんど出なかったという訳ですね。

ベンチ作るのがむずかしい

# 各言語の所感

# Python

## 良かった所

- いろんな言語を使ってみてPythonが1番良かった！
- Pythonとcsvの相性が良すぎる！bomの有り無しを指定できたり、数値はダブルクォーテーションで囲まないように指定できたり、かゆいところに手が届く！
- 書き方もシンプルで好き

## 悪かった所

- インデントがブロックの区切りとなるので、インデントを間違えて最後の集計をloop回実行しちゃったり

# Ruby

## 良かった所

- 2番目にRubyが良かった！
- pで簡単にデバッグできたり、returnの省略やメソッドのかっこ省略できたり、書くの楽！
- 後置if便利

## 悪かった所

- 色々省略できたり、ラクに書ける分逆に読みづらくなる可能性

# PHP

## 良かった所

- 3番目にPHP!
- 書きやすい。読みやすい。大体のことはまあできる

## 悪かった所

- 他の言語使ってからだと、ちょっと書くのめんどくさい。  
(;、変数に\$)けど逆にそれが読みやすいかもしれない。



# Go

## 良かった所

- 4番目にGo！
- 型があるってやっぱええなあ
- defer便利！

## 悪かった所

- 例外が発生しない（返り値でくる）ので自分で処理する必要があつてめんどくさい&バグの原因になる
- 三項演算子が無い

# 感想

- go言語書きたい！と思ってたけど、実際書いてみると使いづらいことが分かった！
- pythonは実際使ってみるとすごく良い言語だと思った！画像処理とか、文章の処理に面白そうなライブラリも多くてもっと触ってみたいと思った！

おまけ

今回は1万行DBに登録する際に、一行ずつinsertしているので  
ここがボトルネックになっていました（あえて）  
この部分をちゃんとバルクインサートにしてあげると・・・

## PHP 8.0.2

	<b>real</b>	<b>user</b>	<b>sys</b>
そのまま1万行(1.6MB)	7.709s	0.390s	0.126s
バルクインサート1万行 (1.6MB)	0.754s	0.470s	0.015s
バルクインサート10万行 (16.6MB)	6.835s	4.722s	0.149s

バルクインサートにしたら1/10くらい早くなった～

このペースだと

- 100万行(166MB) 60秒
- 1000万行(1.6GB) 10分
- 1億行(16GB) 100分

1億行(16GB) 100分くらい行くとPHPで戦うの辛くなりそうかな??

おしまい