はこだて未来大×企業エンジニア 大LT2026

他の言語をRubyっぽくしてみた

自己紹介

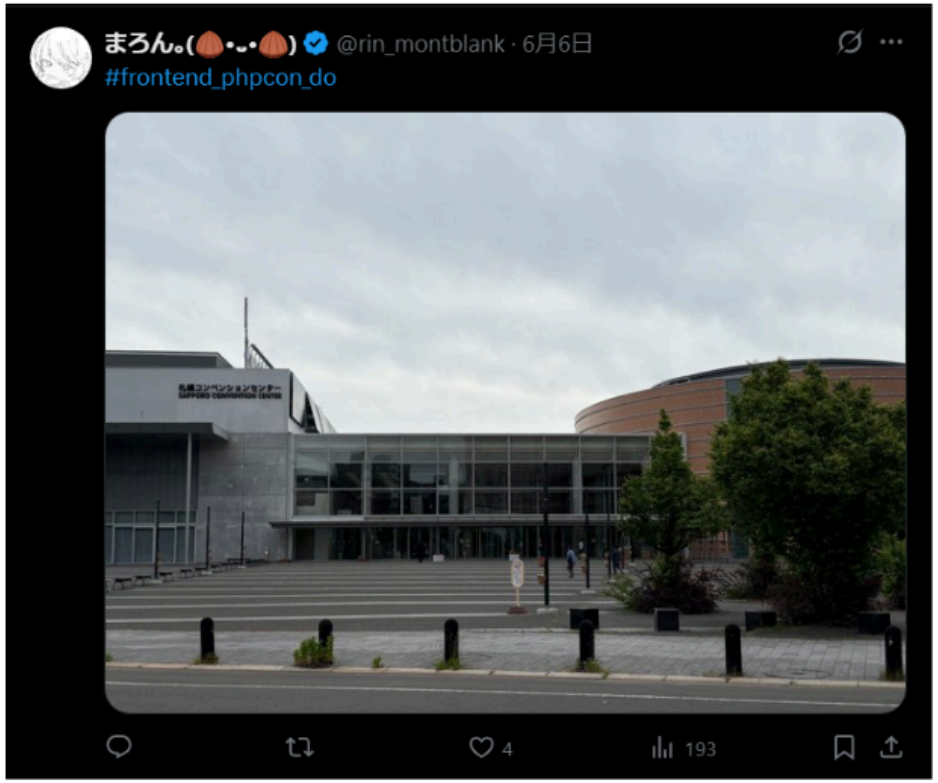
- まろん。 / 村田凛空
- Likes:
  - アマチュア言語処理系
  - JavaScript・AltJS(CoffeeScript等)
  - ネタ開発
  - OSS活動
  - ドメイン収集(最近買った: **gijutusharin.li**)
- Links:
  - https://montblank.fun/
  - Twitter(旧X): rin_montblank
  - Misskey: @m@misskey.otnc.dev
  - GitHub: otnc

自己紹介 — 最近の活動・趣味

旅行

自己紹介 — 最近の活動・趣味

技術カンファレンスに参加



自己紹介 — 最近の活動・趣味

温泉

自己紹介 — 最近の活動・趣味

今朝

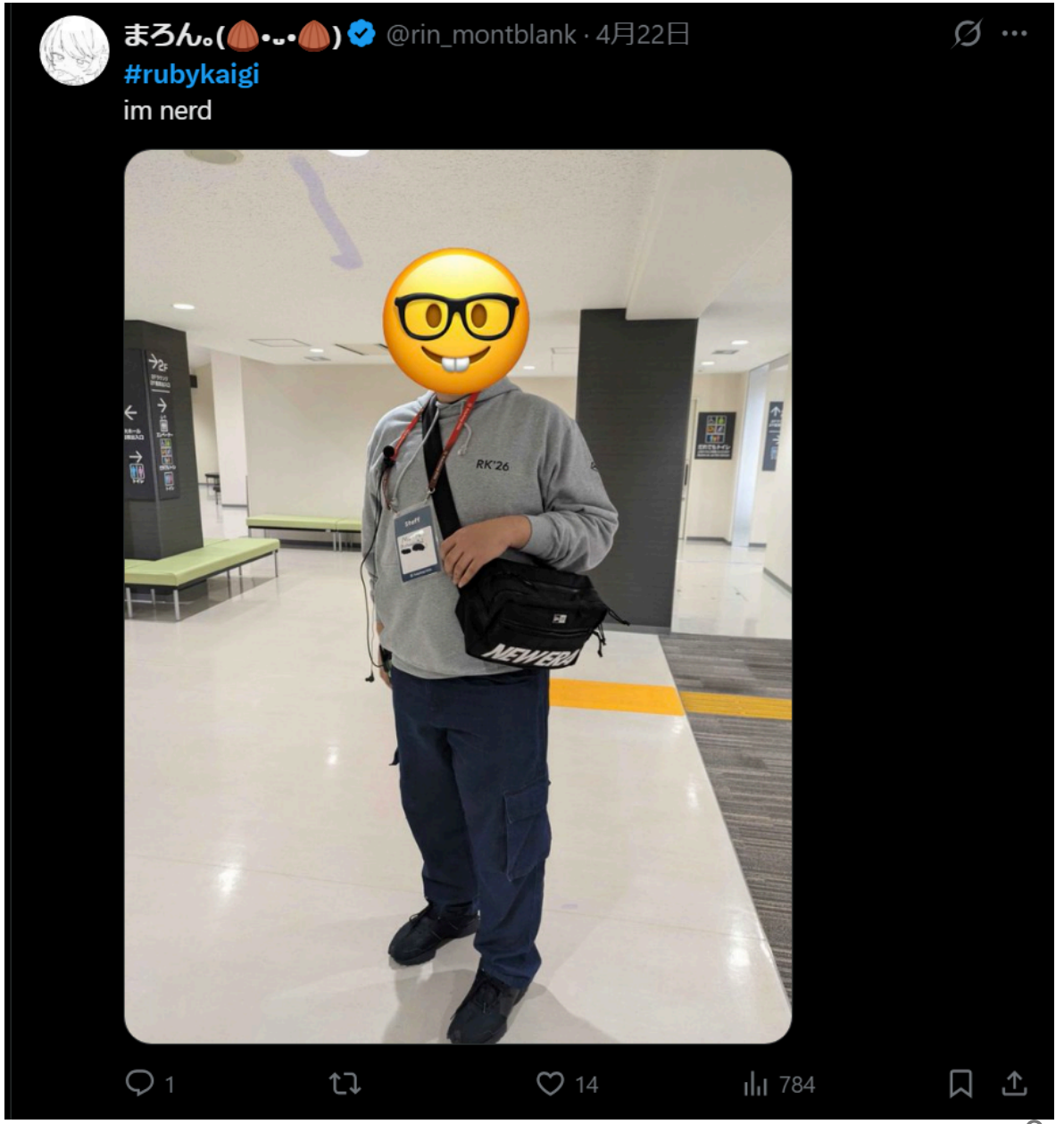
本編

目次

- 概要
- デモ
- まとめ

RubyKaigi 2026に参加した

Staff(Helper)として参加
諸事情で2日目から不在
(1日目のみ参加)
いくつかの発表を聴いた



参加して、思った

Ruby触りたいなぁ…

でも、ﾒﾝﾄﾞｸｻｲ

- プログラミング言語のここがめんどくさい
  - 学ぶ・覚える
    - 文法
    - 主要ライブラリ
  - 使う機会がない
    - それ既に知ってる別の言語でよくない？

そこで、

発想の転換

他の言語をRubyっぽく見せれば実質Rubyでは

やってみる

- 普段からよく使用するJavaScriptとCoffeeScriptで試してみる

それぞれの言語

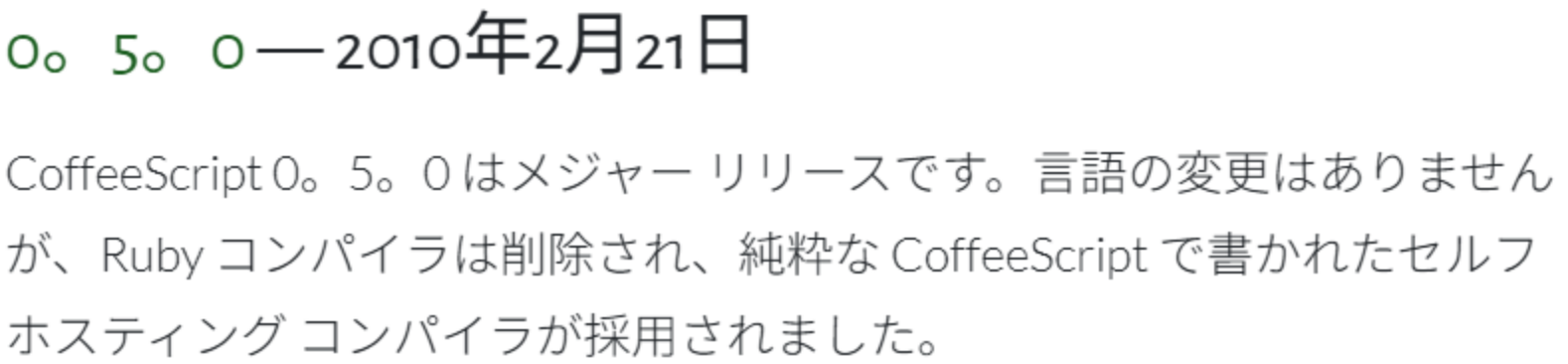
JavaScript

- `console.log` で出力する
- `{` `}` でブロックを表現する(インデントベースではない)
- Rubyと同時期に公開

それぞれの言語

CoffeeScript

- AltJSの一種
- Ruby, Python, Haskellなどに影響されている(インデントベース)
- `v0.5.0` 以前はRuby製のコンパイラを使用していた



エアプが考えるRubyの特徴

- `puts`, `p` 等で出力する
- インデントベース
- `end` がつく

フェーズ1：関数を偽装

`console.log` … どう見てもJS

```
console.log("hi, " + name);
```

⬇️ `puts` … Rubyっぽい

```
puts("hi, " + name);
```

フェーズ1：関数を偽装

```
function greet(name) {
  if (!name) {
    return "hello";
  }
  return "hi, " + name;
}
console.log(greet("marron."));
```

```
function greet(name) {
  if (!name) {
    return "hello";
  }
  return "hi, " + name;
}
puts(greet("marron."));
```

種明かし

事前に定義

`puts` はほぼ `console.log` のwrap

```
const { puts } = require("fxxkinmethod/ruby");
```

実装イメージ

```
// fxxkinmethod/ruby
export const puts = (...args) => {
  /* 中身はただの console.log */
  console.log(...args);
};
```

フェーズ2：文法を偽装

インデントベースに見せかける（括弧・セミコロンを消す）

```
function greet(name) {
  if (!name) {
    return "hello";
  }
  return "hi, " + name;
}
puts(greet("marron."));
```

```
function greet(name)
  if (!name)
    return "hello"
  return "hi, " + name
puts(greet("marron."))
```

フェーズ2：文法を偽装

仕上げに `end` を付ける

```
function greet(name)
  if (!name)
    return "hello"
  end
  return "hi, " + name
end

// ...
```

もう完全にRubyの見た目 … でも、これ本当に動くの？

種明かし

括弧もセミコロンも画面の右端に追いやってるだけ

```
function greet(name)
  if (!name)
    return "hello"
  end
  return "hi, " + name
end

// ...
```

```
let end=null;
{
{
;}

;}
```

➡️ 中身は元の言語のまま。なので動く

フェーズ3：自動化・デモ

手動でインデントをそろえたりするのは面倒 ➡️ **ツールにやらせる**

他の言語のコードをインデントベース言語(風)に変換するツールを作成した

indentier

indentier について

Prettier とほぼ同じコマンドの使い方、設定ファイルの項目

`npm install -D indentier`

`npx indentier --write ./src/ --mode ruby`

今回のデモで使用する例 `.indentierrc.json`

```
{
  "mode": "default", "tabWidth": 2, "useTabs": false,
  "offset": 20, "minColumn": 60,
  "brackets": true, "semicolon": true, "comma": true,
  "ruby": { "variableName": "end", "injectDeclaration": true, "smartEnd": true },
  "plugins": ["@indentier/plugin-coffee"]
}
```

indentier の実装(簡易)

各行末の記号を正規表現で取り出して右端に隠す

- AST構築等は無し、テキスト処理のみ
- `end` の挿入：`{` と `}` の対応を追いかけて、`if` や `function` のブロックが閉じているところに `end` を差し込む

```
function greet(name)                                    {
  if (!name)                                            {
    return "hello"                                      ;} // <--
  end
  return "hi, " + name                                  ;} // <--
end
```

デモ

デモ結果

関数偽装（fxxkinmethod）× 文法偽装（indentier）

➡️ Rubyの見た目っぽくなった、しかもそのまま動く

無理だったこと・難しかったこと

- 言語の制約
  - JS/TS/Coffee
    マクロ機構がない ➡️ `def` などの新しい構文を導入できない
    `do` がすでに**予約語** ➡️ `do...end` 構文作れない
- 言語の自由さ
  - インデントベースではない言語は書き方の自由度が高い
    ➡️ パースとその後の処理がうまくいかないケースがあった
    (見られたケースは修正済みだがまだあるかも)
  - 正規表現では限界があるかも

まとめ

こんなもの作ってる暇があるなら、

Rubyを学ぼう‼️

ありがとうございました

今回作ったモノ

- 今回のスライドとコード
  https://github.com/oto-lt/ltfes2026_fun
- 簡易関数偽装ライブラリ (npm)
  https://github.com/otnc/fxxkinmethod
  https://www.npmjs.com/package/fxxkinmethod
- 簡易文法偽装自動化ライブラリ (npm)
  https://github.com/indentier/indentier
  https://www.npmjs.com/package/indentier
  - CoffeeScript用のプラグイン
    https://github.com/indentier/plugin-coffee
    https://www.npmjs.com/package/@indentier/plugin-coffee