

「考えるのはたのしい」

- とちぎRuby会議09 基調講演
- 2020年9月12日
- 五十嵐邦明 / igaiga

自己紹介

- 五十嵐邦明 / igaiga / twitter: igaiga555
- Rubyist歴 = tDiary ユーザー 2003年9月～
- Rails業務歴 = 万葉入社 2010年4月～
- 2019年7月 ガーネットテック373株式会社設立 (1人会社)
- フリーランスRailsエンジニア お仕事8社 経歴: <https://bit.ly/igaigaesa>
- 書籍: Railsの教科書, Ruby超入門, Rails学習ガイド, パーフェクトRails NEW
 - 書籍一覧: <https://bit.ly/igabooks>



とちぎRuby会議02 2009/10/24 (初参加)

- <https://regional.rubykaigi.org/tochigi02/>
- LT: 五十嵐 邦明(igaiga)(高専カンファレンス) 「Rindaのダブルスペースでタイマー"TwYM"をバージョンアップ」

13:00~13:05 Opening

13:05~14:15 「儲かるRuby」

社長さん、副社長さんに「儲かるRuby」の話をさせていただきます。

- 角征典さん, ワイクル株式会社
- 須藤功平さん, 株式会社クリアコード 代表取締役
- やまだあきらさん, 有限会社ヴァインカーブ 取締役副社長
- 和田卓人さん, タワーズ・クエスト株式会社 取締役社長
- 島田浩二さん, 株式会社 えにしテック 代表取締役

講演者の皆さんのプロフィールを[ここに](#)掲載しています。

14:30~15:30 ライトニングトーク

定員に達しましたので、申し込みを締め切ります。以下、発表順です。

- 波川 よしき (toRuby) 「儲かるドキュメント」
- 設楽 洋嗣 (株式会社えにしテック, Ruby札幌) 「クラウド時代の企業向けWebサイト構築手法のご提案」
- 高井 直人 (Akasaka.rb) 「エンタープライズRailsの真実」
- 池澤 一廣 (有限会社情報デザイン) 「Rubyで描くビジネスモデル」
- 佐藤 基裕 (toRuby) 「雑談コミュニケーションのススメ(インド人編)」
- 江渡 浩一郎 (独立行政法人 産業技術総合研究所) 「儲かるWiki」
- 五十嵐 邦明(igaiga)(高専カンファレンス) 「Rindaのダブルスペースでタイマー"TwYM"をバージョンアップ」
- 関 将俊 (druby.org) 「タイトル未定編」
- 藤岡 岳之 (xibbar) (有) ラビックス 田舎Ruby親方 「田舎でRuby」
- 角谷 信太郎 (日本Rubyの会 || (株)永和システムマネジメント) 「日本Rubyの会第6期の活動について」

15:40~16:40 toRuby勉強会

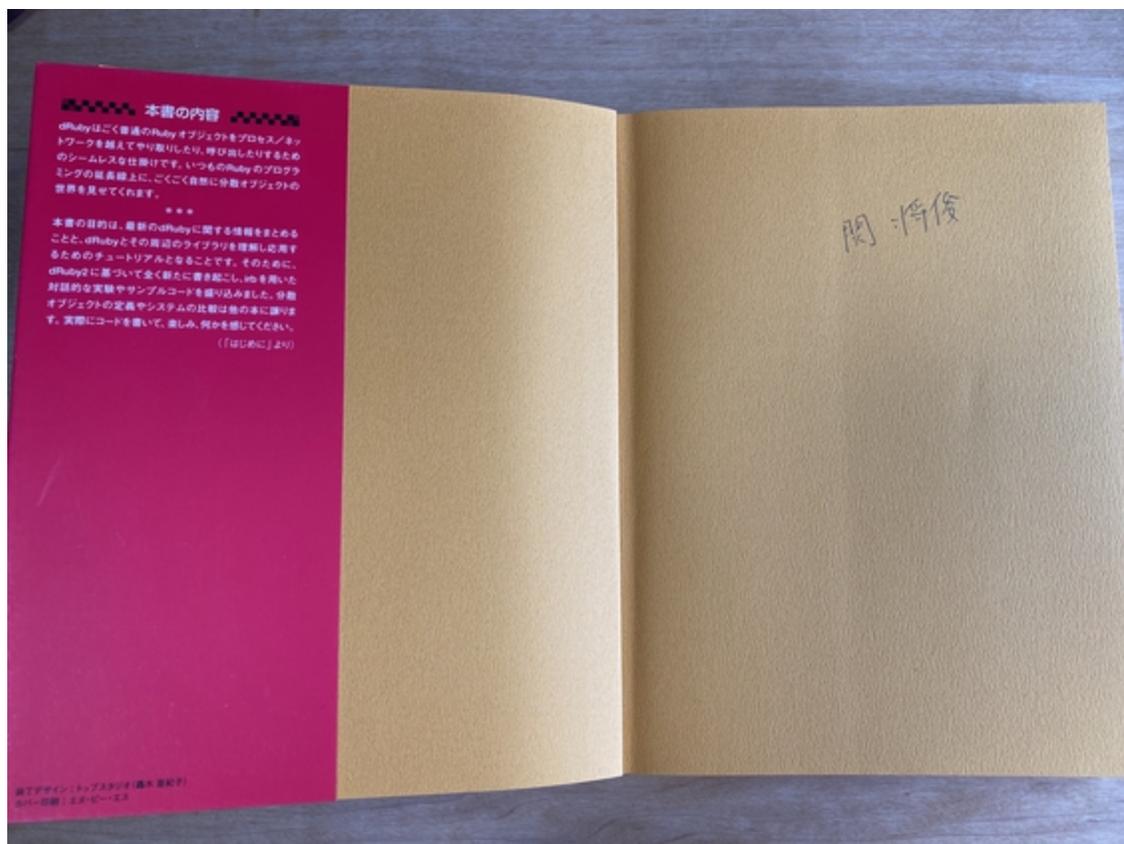
toRubyのいつもの勉強会を行います。

「Rinda」で何かをする」

[「dRubyによる分散・Webプログラミング」](#)を参考書として使います。PCで実際にプログラミングをしながらRindaのお勉強です。

2005/09/03 関さんと（一方的に）会う

- XP祭で関さんからdRuby本(初刷)にサインをもらう
- 初めて技術書に著者サインをもらってめっちゃ嬉しかった



とちぎRuby会議08 2019/06/29 (欠席)

とちぎRuby会議08

那須野が原ハーモニーホール
2019-06-29

Keywords

今年のとちぎRuby会議はこんな感じ！

新元号 同日開催 レポーター 五十嵐さん 呼べない
代理 関 20世紀 自由 入門 初Stripe

「考えるのはたのしい」

- とちぎ
 - たのしいところ
 - かんがえてるところ
- 考えるのはたのしい
- 今日は私が考えてることをいろいろ話します

ひとやすみ

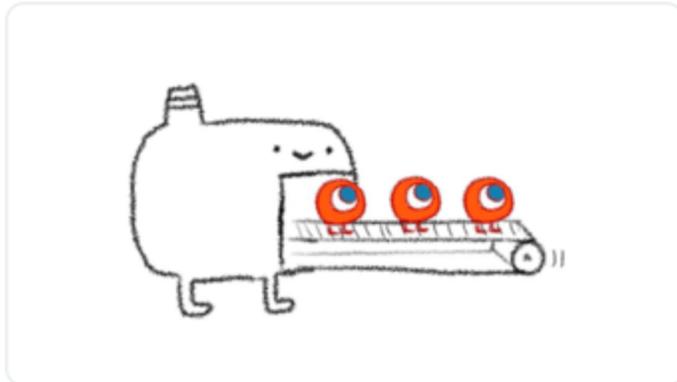
- とちぎのよさで話したいことを語り尽くせたか？
- ここから本題

いのちのかがやきオブジェクトさん



べこもち
@becolomochi

こうですかね🔥
(悪ふざけすぎる...???)



igaiga @igaiga555 · 8月26日

返信先: @becolomochiさん

```
class InochiNoKagayaki
```

```
end
```

```
InochiNoKagayaki.new #=> いのちの輝きオブジェクトさん
```

本を書くときに考えてること

- 分解できるとたのしい
- 特に、初出の概念を1つずつにするように分解したい

繰り返し処理を説明したい

- Array#each だとブロック(do end)とブロック変数(|x|)が同時に登場

```
[1,2,3].each do |x| # ←xをここではブロック変数と呼びます  
  puts x  
end
```

- 3.times だとブロックだけが登場

```
3.times do  
  puts "hi"  
end
```

つまり

- Array#each を最初に説明する
 - ブロックとブロック変数が同時に初出になる
- 3.timesを説明してからArray#eachを説明する
 - 3.timesでブロックが登場
 - つづいてArray#eachを説明するときにブロック変数が登場
 - 🌸 初出のものが1つになるように分解できた
 - ついでにループの回数が毎回同じ ➡ 可変と順序を踏む
 - ラッキー (たまたま)

if

- if 条件式 else end

```
if x == 1
  puts "1"
else
  puts "not 1"
end
```

- if 条件式 end
 - else節はあとから説明すれば良い

```
if x == 1
  puts "1"
end
```

- もっと分解できるか？

- 条件式

```
puts x == 1 #=> true または false
```

- しかもtrueやfalseの概念も独立に説明できて便利
-  こういう順序で初出の概念を順番に出せた
 - 条件式
 - if 条件式 end
 - if 条件式 else end

メソッドの引数と戻り値

- どっち?が良いかあんまりよくわからなかった例（苦労話）
- メソッドの引数と戻り値はどっちを先に説明するのがいい？
- 3つに場合分け
 - 1: 引数を渡さない、戻り値を返さない
 - 2: 引数を渡さない、戻り値を返す
 - 3: 引数を渡す、戻り値を返さない
- 1を最初に説明するとして
- 2と3はどちらを先に説明しますか？

引数を渡さない、戻り値を返さない

メソッドを呼び出す

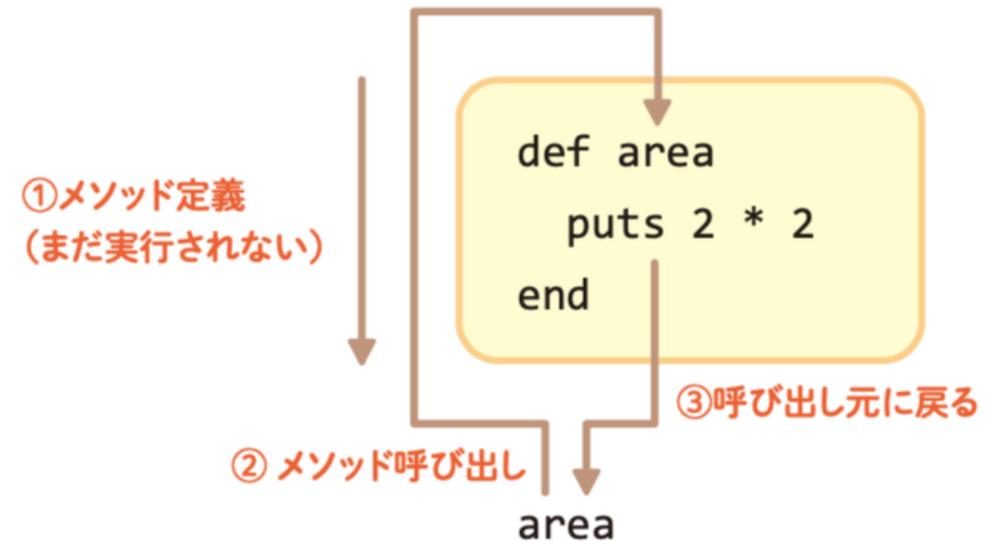
では、さきほど定義した `area` メソッドを実行してみましょう。メソッドを実行することを「呼び出す」といいます。

● `def3.rb`

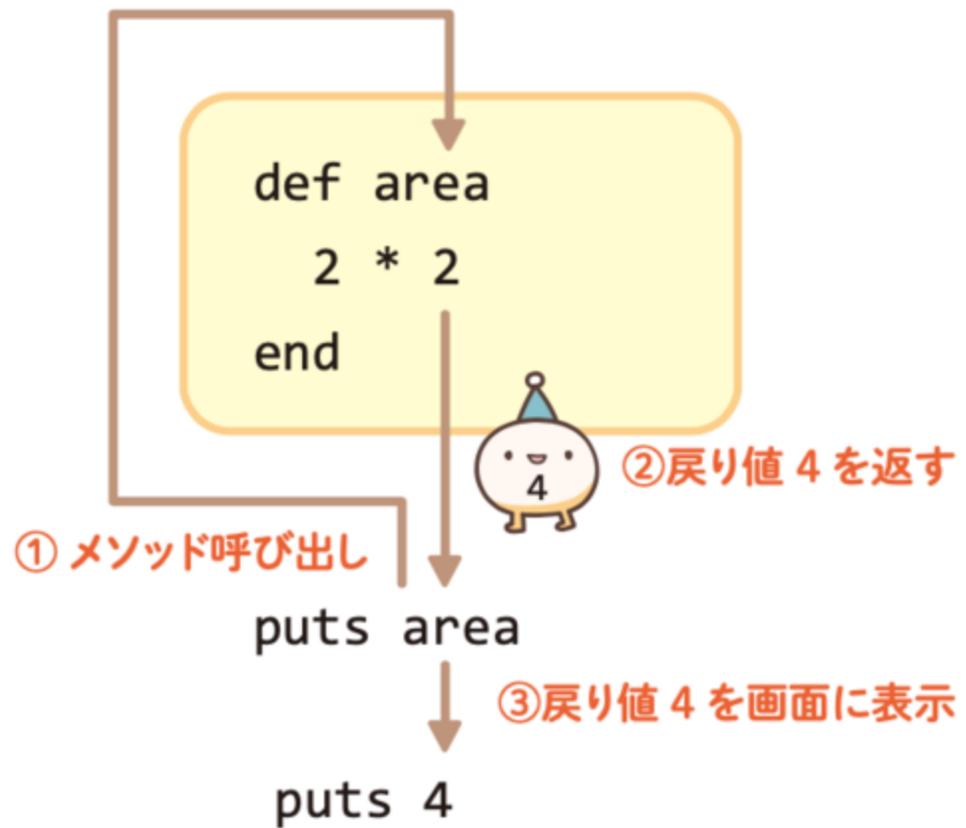
```
1: def area
2:   puts 2 * 2
3: end
4:
5: area ———— 定義したareaメソッドを呼び出し
```

```
ruby def3.rb ↵
4
```

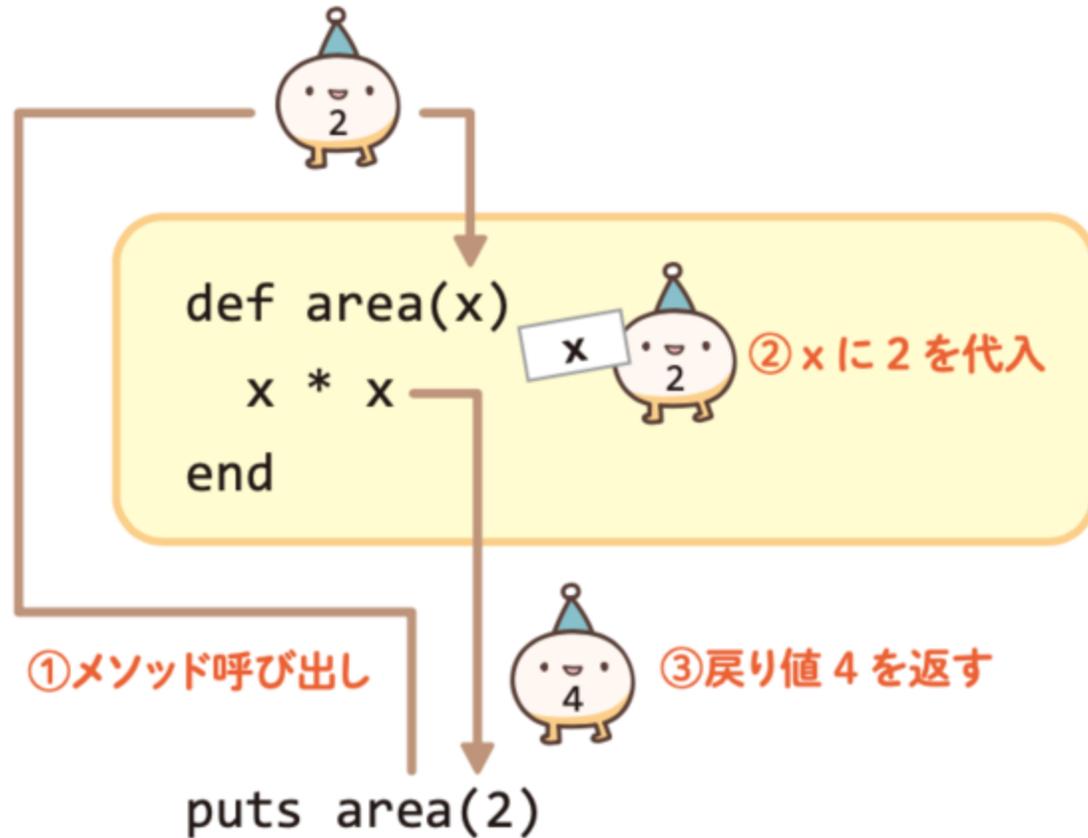
定義したメソッドは、メソッド名を書くことで呼び出す(実行する)ことができます。`area`と書くとメソッドが呼び出され、メソッド中の処理を実行し、`2 * 2`を計算して画面に表示します。



引数を渡さない、戻り値を返す



引数を渡す、戻り値を返す



- 引数を受け取る変数の説明が必要
- 前手順の戻り値の説明は、変数がない方がシンプル

わかりやすい説明は順番が重要

- 文章力、国語力はある一定レベルで成果物への分かりやすさ寄与が飽和する感じがする
- 順番の方が分かりやすさ寄与が大きいのでは仮説
- いろんな入門書の流れを読んでなぜその順になっているかを考えるのはたのしかった
 - 「Java言語プログラミングレッスン」：結城浩
 - 「たのしいRuby」：高橋征義, 後藤裕蔵
 - 「かんたんRuby」：すがわらまさのり
 - 「初めてのプログラミング」：Chris Pine
 - 「_whyの(感動的)Rubyガイド」：_why

http://www.aoky.net/articles/why_poignant_guide_to_ruby/

Ruby超入門とパーフェクトRailsは時間をつかっているところが違う

- 超入門はプロットを書き上げたあとに文章書きながら順番を入れ替えてサンプルコードを変えてという書き直し作業が大部分
- パRailsはプロットを書くためにRailsのコードを読んで書いて動かす時間が大部分
- 提供する価値が違う

ひとやすみ

- ここで半分
- 本の話で話したいことは全部話せたか？



ガートネットテック373

紙版
PDF版
Kindle版

パーフェクト
Ruby on Rails

技術評論社サイト/
書店/Amazon で発売中

紙版
PDF版
Kindle版

ゼロからわかる
Ruby
超入門

技術評論社サイト/
書店/Amazon で発売中

紙版
PDF版
EPUB版

Railsの
教科書

達人出版会/
BOOTH で発売中

PDF版

RubyとRailsの
学習ガイド

BOOTH で発売中

最近お気に入りの考え 「インセンティブ設計」

- incentive: 報酬、動機
- 良いインセンティブ設計をしておけばあとは放っておけばいい感じになるやる
- この分野でおもしろかった本
- 「市場を創る - バザールからネット取引まで」
 - <https://www.nttpub.co.jp/search/books/detail/100001751>
 - 市場経済の仕組みの解説本
- 「その問題、経済学で解決できます。」
 - <https://www.amazon.co.jp/dp/B00NF9PB20/>
 - 行動経済学の本
 - 人がインセンティブにどう応答したかの多数の実験結果

教材づくりはたいへんだ問題

- 「プログラマーはプログラムを書いた方が儲かる問題」
- 教材作成（書籍執筆ほか）よりもプログラムを書いた方が儲かる問題
- もしも逆転させられたら教材を爆発的に増やせる（かもしれない）
- しかしできていない

事例: 執筆スポンサー制度

- 私が2019年にやってみた取り組み
- 20万円1口で私の執筆を支援するスポンサーを募集
- 返礼品「1日稼働券」  1日稼働して、1日執筆日に充てさせてもらう
- 5社さんからお声かけいただいた
- フィヨルドさん、gifteeさんがスポンサーになってくれた
- スポンサーよりも仕事の形の方がよかるうということで仕事になった会社さんも
- 私としては総じてプラスだった
- ただ拡大するための次の手が思いつかない
- 世の中の教材を増やせる仕組みを作れたらいいと思う

ポエム

- 自分で書いてBOOTHで売るといい(宣伝)
 - BOOTH 手数料5.6%+22円
 - 組版システムReVIEWありがたや
 - twitterなどで宣伝しまくる
- 小さく書いて小さく売ろう
 - 2年間書き続けるとつらい
- 技術書典が素晴らしい
 - お祭りインセンティブ
 - 技術書典があると本が増える
 - RubyKaigiがあるとコードが増える
 - withコロナ時代のソリューションの模索

教育はたいへんだ問題

- 「プログラマーはプログラムを書いた方が儲かる問題」
- 教育よりもプログラムを書いた方が儲かる問題
- もしも逆転させられたらRubyコミュニティへ迎え入れる人を爆発的に増やせる（かもしれない）
- しかしできていない

事例: フィヨルドブートキャンプ

- <https://bootcamp.fjord.jp>
- ⚠️ 私は顧問ですが、今日の話は中の人公式な意見ではないです
- 現役エンジニアたちがメンターとして提出物などをレビューする
- 就職先企業のエンジニアたちもslackなどに登場する
- 企業研修としても利用する企業もある（別契約）
- ペパボさんのDAIMYO Engineer Collegeとも教材交換や留学制度などでコラボしている
- 受講生と留学生と関連エンジニアたちのコミュニティができています

事例: フィヨルドブートキャンプ

- 受講生が増えてきている（最近100人超えました）
- 事業継続可能なプログラミングスクールになれるかもしれない
- とても良い事業なので応援したい応援している
- 受講生が持続可能である状態をどう設計するか（後述）
- 事業継続可能な収益をどう設計するか（後述）

受講生の経済的な持続可能化

- 受講生は仕事をやめてきている人もいる
- 学習時間800時間ほどやってアルバイトができるようになれば持続可能になる
- アルバイトが潤沢にあるのか問題
 - 受講生へ向けたアルバイトを提供できる企業お待ちしております
- アルバイトできるようになるまでどうするのか問題
 - フィヨルド紹介企業就職で6ヶ月分キャッシュバック
 - 後払い奨学金みたいな感じか
 - これを受講生収益にしているモデルは興味深い
- 他の奨学金制度が作れるとより良いインセンティブ設計ができる

フィヨルドブートキャンプ事業としての持続可能化

- 受講生からの収益 月額: 29,800円
- 受講生が増えていけば収益も増えて経営が軌道に乗る
- リモートメインなのでオフィスは広くなくても大丈夫
 - (withコロナ時代になりその傾向が加速した)
- メンターがいれば受講生を増やせる (たぶん)
- その他の方法の収益をインセンティブ設計できるか？

フィヨルドブートキャンプでは副業メンターを募集する場合があります (宣伝)

- メンター（日報と提出物レビュー係）募集中（現在は推薦制）
- 副業アルバイトメンター制度、月10時間から
- 「プログラマーはプログラムを書いた方が儲かる問題」を回避する一時的な策
- 現在のところまだフルタイムで人を雇える経営システムが作れてない
- 一方でフルタイムでメンターの仕事をしてもらおうと、その人の現場力が失われてメンター価値が下がる問題もある
- フルタイム雇用で週2フリーランス、週3メンターとかがいいのかも？

ポエム

- 教材、教育による受益者群は？
 - Rubyをはじめた人たち（受講生）？
 - 教育側の人たち（スクール、執筆者）？
 - Rubyで開発している企業たち？
 - コミュニティの人たち？
 - そのほかの人たち（Webサービス利用者たち、行政の就業支援をする人たち）？
- 各者の利益は何か？
 - 今の世界ははどんなインセンティブ設計になっているのか？
 - それよりも良いインセンティブ設計はあるのか？
 - 局所的でも、今よりも良い設計ができれば、世界を良くできるはず

フィヨルドブーツキャンプ = 『おーい、磯野、野球やるうぜ』

「僕らはプログラミングとプログラミングをする人が好きなのであって教えること自体が好きじゃなくない。

もちろんプログラミングをする人とプログラミングを通じたコミュニケーションである『教えること』も嫌いじゃないが、とにかくやりたいのはプログラミングなのだ。

野球が好きで『おーい、磯野、野球やるうぜ』と誘うのだが、野球のルールもバットの振り方も知らないというのではメンツにならない。

だから教えるという感じだ。」

komagataのブログより

<https://docs.komagata.org/5588>

RubyistとRubyを書くことのいい

- 現在の私の気持ちをまとめると次のような感じになるようだ
- Rubyを書くことのいい
- Rubyistと会うことのいい
- 私の日常や旅にRubyistがいない機会の方が少ない
- Rubyコミュニティが長くつづいていて嬉しい
- Rubyコミュニティでこれからも多くの人とのいい時間を過ごしたい
- そのためにコードや本や場所をつくっていききたい
- 「持続可能なたのしさ」のインセンティブ設計を作っていきたい

おわり

フィヨルドブーツキャンプについてもっと知りたい方は

大江戸Ruby会議08 町田さんの発表資料

<https://speakerdeck.com/machida/puroguramingusukuruwoshi-metajing-wei-tokorekarafalsehua>

角谷さんのフィヨルドブーツキャンプでの講演資料

<https://speakerdeck.com/kakutani/fjorb-boot-camp-as-a-gate>