

優しいvim入門

20卒の皆さんこんにちは

軽くてくらんちの説明を。。。。

隔週でお昼にしているゆる~い勉強会です

みなさんも是非いろいろお話ししてみてください！！！

whoami

whoami



whoami

- 日下部拓人
- id:t-kusakabe
- 19新卒
- SREの下っ端もどき
- vimが好き
- 一言 : SAA取ります

新卒なので技術のことわかりません。
マサカリは投げないでください。

今日はvimはいいぞ！という話をします。

＊使うことに専念します。

＊何か作ったりは特にしません。

ゴールはvim使ってみたいなと思っていただくこと！

お断り

本スライドは、圧倒的主観であり戦争の引き金になりたい訳ではございませんのであらかじめご了承ください。

ところでみなさん、エディタは何を使われていますか？

vscod?

|IDEA???

もっといいものを使いましょう！！！

vim is 何？

Vim Advent Calendar 2012 の記事

かなりすごいブログ

Macを購入したら絶対に導入したい！私が3年間で厳選した超オススメアプリ10選！



2013年10月10日

[はてブに追加](#)

[この記事をツイート](#)



[Feedlyで購読](#)

さて本日は、MacBook Airを2010年に買ってから、数々のフリーソフト・シェアウェアを使い続けてきた私が、MacBookを買ったら「絶対に」導入したい、超厳選アプリをランキング形式でご紹介したいと思います！今回は無料で使えるものばかり集めてみたので、シェアウェアに関しては後日またご紹介したいと思います

(^ ^)

それでは、ご覧ください！

↓ ↓ ↓

第10位



MacVim

まずは定番、Vimですね。知る人ぞ知るテキストエディタです。テキストエディタの記事では無いので詳しい説明は避けますが、Vimは「モード」という概念とキーの組み合わせによるテキストの操作方法が特徴的なエディタで、日本でも根強いファンは多いです。プログラマの方はぜひ導入すべきと言えるアプリでしょう。

ダウンロードはこちらから！



MacVim

MacといえばApple、AppleといえばiPod、iPodといえば音楽です。 そう、せっかくMacを買ったからには、音楽を楽しむなければいけません。 デフォルトでインストールされている「iTunes」でもよいのですが、ここはキーボードから小気味よく選曲などの操作をして音楽を聴きたいところです。 そんなあなたにオススメなのが、このMacVimというアプリ！なんと[プラグインの導入](#)によってiTunesの操作をキーボードで行うことができるようになるアプリなのです！MacVimには、おまけとしてテキストエディット！を出来る機能も付いているので、テキストをエディットすることもできてかなり便利ですね（かなり）。

ダウンロードはこちらから！



MacVim

さて、WindowsでもMacでも変わらない現代の必須ソフト、それは「ウェブブラウザ」ですね。WindowsだとInternetExplorer、通称IEが標準でインストールされていますが、IEは「ブラウザをダウンロードするための便利なソフト」と揶揄されています（苦笑）

Macに標準でインストールされているブラウザであるSafariはそれほど酷評されているわけではありませんが、やはり自分の手に馴染むブラウザを選択して使っていきたいですよね。

そこでMacVimというアプリです（＾＾）

w3mというやつをなんかインストールみたいなことして、MacVimにw3m.vimというプラグインを導入すれば、そう、ウェブブラウザです！

あとテキストをエディットすることもできます！

ちょっと飛んで

第3位



MacVim

第3位は「MacVim」です！世界にはEmacsとVim、2種類のエディタしかないことは有名ですが、これはVimの方です！Mac用のVimです。

さて、この第3位のMacVimというアプリなのですが、第4位のMacVimと、どちらを3位にしようか、最後まで迷って悩みました！（笑）（＾＾；）それくらい、MacVimとMacVim、どちらもすごくいいアプリなんです！



MacVim

さて、惜しくも1位を逃すこととなった第2位は、「MacVim」です！

「Vim」、中には初めて耳にした方も居るかもしれませんね。一般的に、PCを使う場合はその用途の9割がテキストエディットで残り4割がテキストエディットだと言われています。そう、つまり、Macを使うにあたってテキストエディット用ソフトウェア、通称テキストエディタ選びというのは非常に重要なのです。Macには多くのテキストエディタがありますが、あなたに合った最適なテキストエディタを選ぶ

ということは、あなたのPCライフそのものを最適化するということに他なりません。

そこで私がおすすめしたいのは、やはりMacVimです。最終的に他のエディタ、例えばEmacsなどを使っていくにしても、「効率的にテキストを編集すること」を突き詰め、テキストエディタとしての新しいパラダイムを実現したとも言えるVim(Vi)を経験しておくことは非常に大切でしょう。



MacVim

MacVimで
す！！！！！！！！！！！！！！！！！！！！

つまり。。。。

vim is 最強

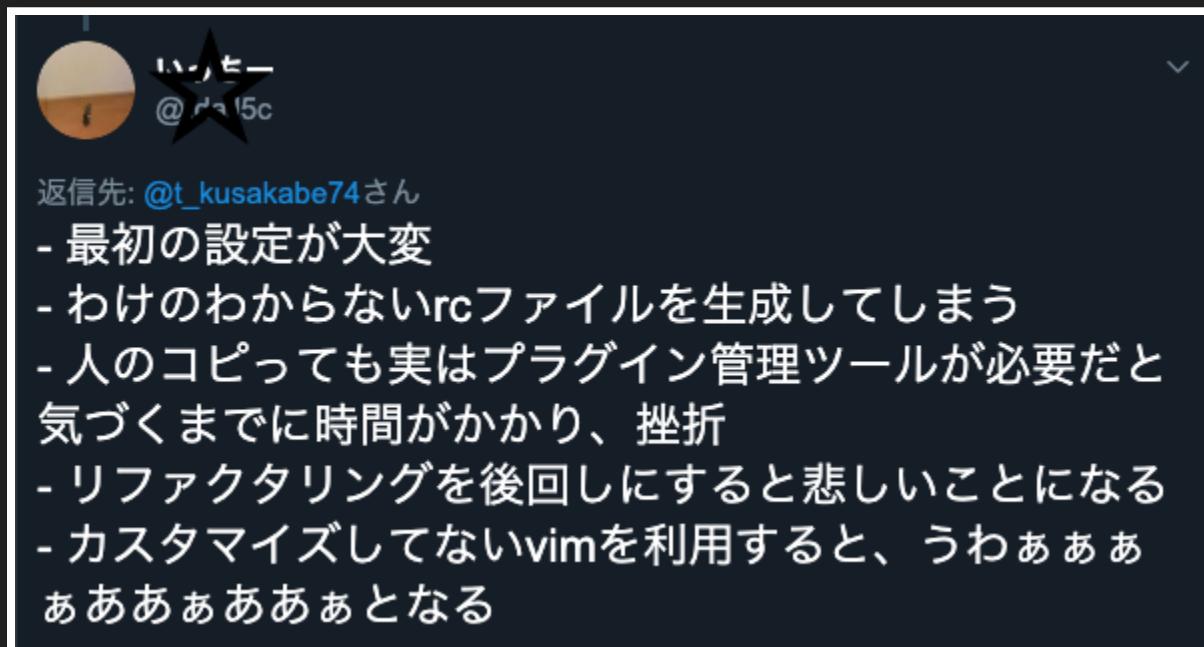
vimで出来ないことはない。

とはいえ

やっぱり使い始めるまでに色々障壁がある。

今日はその障壁を突破していきましょう。

悩める先輩A



返信先: [@t_kusakabe74](#)さん

- 最初の設定が大変
- わけのわからないrcファイルを生成してしまう
- 人のkopipっても実はプラグイン管理ツールが必要だと気づくまでに時間がかかり、挫折
- リファクタリングを後回しにすると悲しいことになる
- カスタマイズしてないvimを利用すると、うわあああああああとなる

1つずつ見ていく。

最初の設定が大変

dotfileを作ろう！！！

dotfileとは。。。

コンピュータ

ドットファイル【どっとふあいる】



Unix系のOSにおいて設定ファイルなどの名前を.から始まる名前にし、隠しファイルとしたもの。

目次

たとえば.htaccess、.bashrcなど。同様にディレクトリも.から始まる名前にすることで隠すことができる。

ドットファイルとは

つまり.vimrcを作つてdotfileに含めておけば大変なのは
最初の一回だけ！

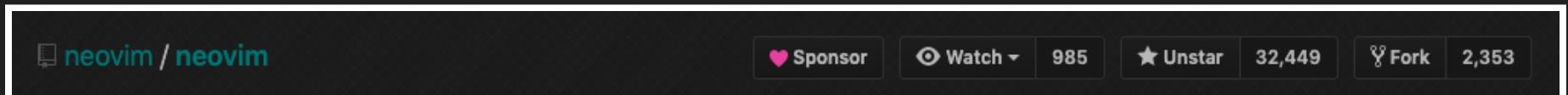
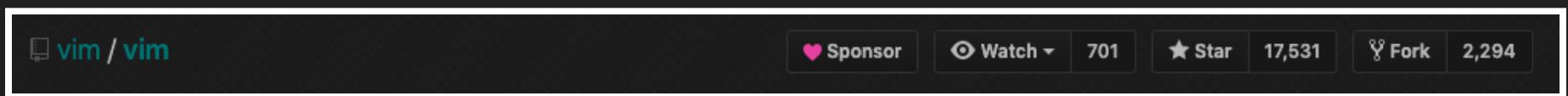
頑張って書きましょう。

わけのわからないrcファイルを生成してしまう

neovimを使おう！

実はvimといってもたくさん種類がある。

本家vimよりstarも多い



そしてneovimは.vimrcではなく、init.vim。

つまり、わけのわからないrcファイルが生成されない！！！ｗ

人のコピっても実はプラグイン管理ツールが必要だと気づくまでに時間がかかり、挫折

今日この勉強会で解決しましょう**💪**

リファクタリングを後回しにすると悲しいことになる

定期的に見直す時間は必要。。。。

zc-vimを作りませんか？？？

カスタマイズしていないvimを利用すると、うわああああ
あああああとなる

わかる。

なのでみんなでvimを使いましょう！！！

vim選び編

いろんなvimがある。

macvim



- いわゆるgui vim
- cliが苦菜人には嬉しいかもしれない
- OS Xならbrew installで入る

onivim

The screenshot shows the GitHub page for the Oni 1 repository. At the top is the Oni logo, a circular icon with a green-to-blue gradient and a white circular arrow containing a downward-pointing triangle. To the right of the logo is the word "ONI" in a large, bold, dark font. Below the logo and title is the subtitle "Modern Modal Editing". A horizontal line separates the header from the body. In the body, there is a note: "NOTE: We're now shifting our focus to building Oni 2" followed by two small worker emoji icons. Another horizontal line follows. At the bottom, there are three green buttons: "build passing", "build passing" (with a GitHub icon), and "codecov". Below these are two more green buttons: "chat 94 online" and "downloads 134k".

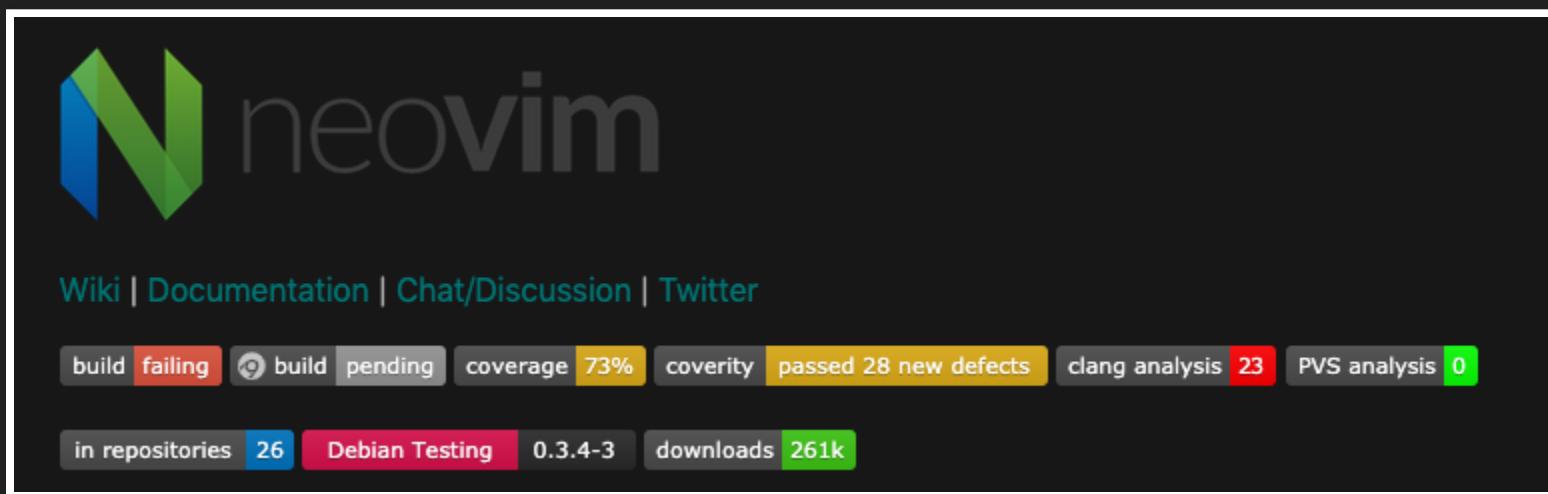
NOTE: We're now shifting our focus to building Oni 2 🚧 🚧

build passing build passing codecov

chat 94 online downloads 134k

- vscode likeなvim
- neovimをguiにしたもの
- vscode愛用者には嬉しいかもしれない

neovim



- 平たく言うとvimでは成し得なかつことをするために生まれたもの
- 非同期処理をやりやすくしたかったりする

僕はneovimを使っているので、 neovimの想定で話をして
いきます。 (以降nvim)

※ 基本的にはそんなに違いはありません。

vim操作編

きっとみんなさんが一番挫折するポイント。

とはいって普通に使う分にはそんなに覚えることはない！

vimにはいろいろModeがある

i と esc で入力とコマンドModeを切り替える

移動は h, j, k, l

保存は :w

これだけ覚えておけば今日からでもvimで作業ができます！

vim plugin編

vimのいいところはカスタマイズ出来るところ。

まずはpluginを管理しやすい基盤を作る。

- Vundle
- vim-plug
- NeoBundle
- dein.vim

今回はvim-plugを使います。

たったこれだけ！

あとは好きなpluginを書いていくだけです。

ちなみに僕の構成はこんな感じです。

```
.  
├── colors  
│   └── iceberg.vim  
├── ftplugin  
│   ├── dart.vim  
│   ├── go.vim  
│   ├── php.vim  
│   ├── pug.vim  
│   ├── ruby.vim  
│   └── yaml.vim  
└── init.vim  
└── userautoload  
    ├── basic.vim  
    ├── plug.vim  
    └── statusline.vim
```

おすすめplugin編

vimの使いやすさはカスタマイズ度合いで決まる。

便利なpluginをフル活用しましょう！

fzf



command-line fuzzy finder

- golang^製
- cliでインクリメンタルな曖昧検索ができる
- pecoみたいなやつ

これをvimと組み合わせると。。。。

```
337
340 >>>
341
342 便利なpluginをフル活用しましょう！
343
344 >>>
345
346 
347
348 >>>
349
350 - golang製
351 - cliでインクリメンタルな曖昧検索ができる
352 - pecoみたいなやつ
353
354 >>>
355
356 これをvimと組み合わせると。。。
357
358 >>>
359
360 
361
362
xx NERD_tree_1 - 60% 15:1 TERMINAL master | slide.md 98% 359:1
node_modules/cli/cli.js
node_modules/kew/kew.js
plugin/zoom-js/zoom.js
plugin/notes/notes.js
test/test-markdown.js
test/qunit-1.12.0.js
lib/js/classList.js
lib/js/html5shiv.js
plugin/math/math.js
lib/js/head.min.js
js/reveal.min.js
test/test-pdf.js
package.json
js/reveal.js
test/test.js
Gruntfile.js
> bower.json
6670/8428
~/d/s/g/t/slide-techlunch-vim/js
> fzf
```

高速にfile検索が出来る！

vscode等の cmd + p 相当のことが高速にできる。

beffuerを使うより早いのでは。。。？

coc.nvim



Conquer of Completion

Make your vim/neovim as smart as VSCode.

license MIT bounties 400 build failing coverage 71% doc :h coc.txt chat on gitter

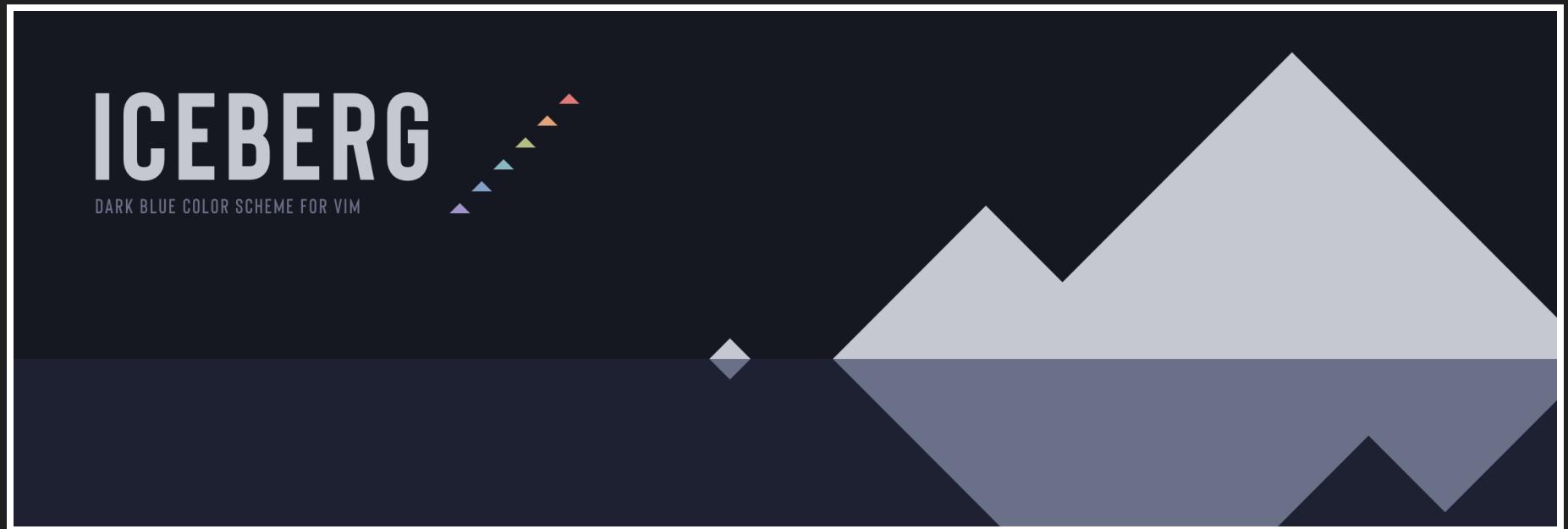
vimの補完は永遠の課題。。。。

coc.nvimはそこそこいい感じの解決策な気がしています。

Make your vim/neovim as smart as VSCode.

js周りは、まじでvscodeくらいの補完ができる

iceberg.vim



エディタにとって見た目はとても大事

5+ ./vimrc | ./vimrc

```

591 " camelcasemotion {{{
592 map <silent> b <Plug>CamelCaseMotion_b
593 map <silent> e <Plug>CamelCaseMotion_e
594 map <silent> w <Plug>CamelCaseMotion_w
595 " }}}
596
597 " ctrlp {{{
598 let g:ctrlp_cache_dir = s:env.path.data . '/ctrlp'
599 let g:ctrlp_clear_cache_on_exit = 0
600 let g:ctrlp_custom_ignore = {
601   'dir': '\v[^\(\)(node_modules|\.git)\$',
602   'xml': []
603 }
604 let g:ctrlp_map = ''
605 let g:ctrlp_match_window = 'bottom,order:ttb'
606 let g:ctrlp_prompt_mappings = {
607   'PrtBS()': ['<C-h>', '<BS>'],
608   'PrtClear()': ['<C-k>'],
609   'PrtCurLeft()': ['<C-b>', '<Left>'],
610   'PrtCurRight()': ['<C-f>', '<Right>'],
611   'PrtDelete()': ['<C-d>', '<Del>'],
612   'PrtHistory(-1)': [],
613   'PrtHistory(1)': [],
614   'PrtSelectMove("j")': ['<C-n>', '<Down>'],
615   'PrtSelectMove("k")': ['<C-p>', '<Up>'],
616   'ToggleByFname()': [],
617   'ToggleType(-1)': ['<C-Down>'],
618   'ToggleType(1)': ['<C-Up>'],
619 }
620 let g:ctrlp_tilde_homedir = 1
621 nnoremap <C-g> :CtrlPMixed<CR>
622 " }}}
623
624 " flow {{{
625 let g:flow#enable = 0
626 let g:flow#omnifunc = 0
627 " }}}
628
629 " go {{{
630 let g:go_version_warning = 0
631 " }}}
632
633 " javascript {{{
634 let g:javascript_plugin_flow = 1
635 " }}}
636
637 " mucomplete {{{
638 let g:mucomplete#enable_auto_at_startup = 1
639 let g:mucomplete#no_mappings = 1
640 " }}}
641
642 " shadeline {{{
643 let g:shadeline = {}
644 let g:shadeline.active = {
645   'left': [
646     'fname',
647     'flags',
648   ],
649   'right': [
650     '<',
651     ['ff', 'fenc', 'ft'],
652     'ruler',
653   ],
654 },
655 }
656
657 let g:shadeline.inactive = {
658   'left': [
659     'fname',
660     'flags',
661   ],
662 },
663
664 function! ShadelineItemGitBranch()
665   let name = exists('*fugitive#head')
666   if !name
667     return empty(name) ? '' : printf('(%s)', name)
668   endif
669 }
670
671 " waffle {{{
672 let g:waffle_auto_cd = 1
673 " }}}
674
675 " Disable unused plugins
676 let g:loaded_gzip = 1
677 let g:loaded_tar = 1
678 let g:loaded_tarPlugin = 1
679 let g:loaded_zip = 1
680 let g:loaded_zipPlugin = 1
681
682 " Disable unused kaoriya plugins
683 let g:plugin_dicwin_disable = 1
684
685 " Local settings {{{
686 if filereadable(s:env.path.local_vimrc)
687   execute 'source' . s:env.path.local_vimrc
688 endif
689
690 " cocopon.me public
691
692 49 40 41 42 43 44 45 46 47
693 39 *** *** *** *** *** *** *** *** *** ***
694 30 *** *** *** *** *** *** *** *** *** ***
695 1;30 31 *** *** *** *** *** *** *** *** ***
696 1;31 32 *** *** *** *** *** *** *** *** ***
697 1;32 33 *** *** *** *** *** *** *** *** ***
698 1;33 34 *** *** *** *** *** *** *** *** ***
699 1;34 35 *** *** *** *** *** *** *** *** ***
700 1;35 36 *** *** *** *** *** *** *** *** ***
701 1;36 37 *** *** *** *** *** *** *** *** ***
702 1;37 38 *** *** *** *** *** *** *** *** ***
703
704 cocopon%
705
706 zsh [running] [+]

```

icebergを使えばとりあえずそれっぽくなります。

この辺りは無限に話せるのでおすすめ、気になるplugin
があれば教えてください。

お知らせ

隔週でやる予定だったてくらんちが毎週になるみたい
です 😅

ルールとしては、発表者が2週間後の発表者を決める、
というのは変わっていないです。

次回は西口さんの怖いtwilio勉強会！！！

- 現状のTwilioのプロダクト(赤電話)に何がどう活かせるか
- 今後のTwilio以外も含めた、WebRTC界隈の流れの予想
- 海外カンファレンス参加で得られたものと感想

再来週は、草間さんの怖いkubernetes！！！

きっと恐ろしい話が聞けると思います👀

質問タイム