

OSSコントリビュートデビュー・俺の話を聞いてくれ

MSD/SDU 井上

はじめに

OSSに初コントリビュートをした感想と簡単な流れについて記述する。

お断り

- 本LTはエンジニアであるスピーカーが自慢をたれ流す時間がほとんどです。
- 質問は思いつかないかもなので実況チャンネルには感想とか昨日の夕食とかツッコミとかwelcomeです。
- gitの用語がわからなかったら後で連絡してください。丁寧にお答えします。

Agenda

- OSSとは
- OSSコントリビュートとは
- 具体的に何やったん？
- 実際にコントリビュートしたものたち
- 感想
- 最後に

OSSとは

オープンソースソフトウェアのこと

ソースコードの改変、再配布を自由に行なって良いもの

PHP, Java, Rubyなど超有名プログラミング言語もOSS

OSSコントリビュートとは

文字通りにOSSの開発の手助けをすること

例えば

- バグの修正
- 新機能の追加
- ソースコードのリファクタリング
- ドキュメントの修正（Typo、翻訳）

具体的に何やったん？

色々頑張ったけど要約すると以下が効果あった

- やり方をググる
- コントリビュートできるまとめページをdigった
- すぐにリポジトリをforkして変更作業着手

やり方をググる

OSS コミット やり方 で検索してみた

- コントリビュートできそうなOSSを探す = issueを報告しているOSSを探す
 - [good first issue](#)
 - [Github Help Wanted](#)
 - [GitHub](#)

good first issue とは

GitHubのissueのラベルのうち `good first issue` が含まれるものをまとめているページ

*good first issueは 初めてのコントリビュートに向いている Issue につけられる ラベル

issueの選び方

もちろん何でもコントリビュートできる訳ではない
ケイパや興味で選ぶ必要がある

僕の属性

- 得意なのはGo, TypeScript, DevOps(GitHub Actions)
- 興味があるのはテスト、スクレイピング、ORM、HTTPフレームワーク

変更作業

私がやった流れは以下

1. とりあえず `Can I take this?` と投稿する
2. コントリビューートのルールを確認する (Readmeを読む)
3. 間髪を入れずリポジトリをfork（自分のアカウント配下にコピー）する
4. ブランチを切って変更作業をする
5. リモートリポジトリにpushしてコントリビューート先にプルリクエストを出す

実際にコントリビュートしたもののたち

Goのテストコードの修正

<https://github.com/MontFerret/ferret/pull/781>

<https://github.com/gavv/httpexpect/pull/347>

テストコードの軽微な修正

GitHub Actions Warningの修正

<https://github.com/fairhive-labs/go-pixelart/pull/16>

実は10月にプロジェクトで行なった作業と全く同じ

不要な引数の削除

<https://github.com/apache/camel-k/pull/4155>

未使用の関数の引数を削除するだけ

なんとapacheのリポジトリにコントリビュート

感想

- Thank you というコメントが嬉しい
- 初めてで緊張したが楽しかった
- Thank you というコメントが嬉しい
- 会ったことない人とのものづくりできるの楽しい
- Thank you というコメントが嬉しい

最後に

知らない人に自分のやったことを認めてもらうのは嬉しい
やろうぜOSSコントリビュート