

GitHubとは
Gitをオンラインで管理できるサービス
Gitとは
バージョン管理システム
ファイルに変更点を加えたときわざわざ名前を付けて保存する必要なし
(200417_document_new_txtみたいな)
複数人で共有ファイルを編集したとき先に編集した人のデータが消えたりしない
それをオンラインでできるのがGitHub

導入を始める前の予備知識

リポジトリ
ファイルやディレクトリの状態を記録する場所
状態というのは変更履歴など(図解)
個人で使用するローカルリポジトリと共有するリモートリポジトリがある(以下ローカル、リモート)

コミット
変更履歴の記録を行う(図解)

プッシュ
変更履歴をローカルからリモートにアップロードする(図解)
ローカルでの作業がリモートに反映される

クローン
リモートをローカルに複製する(図解)
リモートに影響を与えず作業できる

プル
リモートからローカルに変更履歴をダウンロードする(図解)
最新のファイルを取得する

クローンはリポジトリそのものを複製
プルは変更履歴だけダウンロード

ブランチ
Gitの機能の一つ
履歴の流れを分岐して記録する機能(図解)
『複数人で共有ファイルを編集したとき先に編集した人のデータが消えたりしない』
のはこの機能のおかげ

WebでGitHubの使い方について調べると必ずと言っていいほど「Gitを使って～」とある
ここでGitの問題点

コマンド操作
慣れるのに時間がかかる
視覚的にわかりにくい
なので

Gitは使わない(使いたかったら個人で頑張れ)
ロボコンするうえではおそらく必要ない
GitHubで十分
(企業とかはGitめっちゃ使ってるところもあるらしいから学んで損はない)

—GitHubを実際に使って見せたり—
そこでプルリクエストやマージなども教える
プルリクエスト

プッシュ時などにレビューを挟むことでバグが起きにくくなったりする
マージ
プルリクエストの許可
(厳密には違うけどここでは説明しない)

GitHub Desktopの紹介

利点

- 好きなエディターで編集できる
- クローンなどの操作が簡単
(ロボコンでは必要ないだろうけど画像の差分表示可能)

--これも実際に使って見せたり--

fetchとか教えたり

fetch

pullの前段階的なもの

リモトリポジトリの内容を確認するだけでローカルに反映されない