

GitHubの講習を始めます  
まずGitHubとは何なのか  
Gitをオンラインで管理できるサービス  
じゃあGitてなに

Gitというのはバージョン管理システムのこと

従来、開発などで以前の作業データを残しておきたい、つまりバージョンを管理したい、  
というときはこのように先頭に日付を記入したファイルをたくさん作ったりしてた  
ほかにも共有フォルダにある同じファイルを複数人が同時に触るとき、先に変更した人の  
データがほかの人のデータで上書きされたりしてしまうことがありました  
それらの問題を解決するのがバージョン管理システムです  
つまりGitHubはそれをオンラインで行える便利なサービスです

ではGitHub導入前に知っておいてほしい事前知識の説明をしていきます

まずリポジトリとは

ファイルやディレクトリの状態を記録する場所です。状態というのはファイルの変更履歴  
のことです。

例えば。このそれぞれの四角をファイル。まずabcと書かれたファイルがある。次にabcを  
消す。そしてedfと書き込む。するとリポジトリには「abcが消されedfと書いた」という  
状態が保存されてる

ローカルリポジトリ、リモートリポジトリというのはそのリポジトリがどこにあるのかで  
区別される。ファイルを共有する場所、GitHubならオンラインにあるリポジトリがリモ  
ートリポジトリ。個人の開発環境に置かれるのがローカルリポジトリ

次コミットとは

変更履歴の記録のこと。リポジトリにデータを保存することをコミットという、というだ  
けです。ちなみにリポジトリにはこんな風に今までのコミットの歴史がすべて記録されま  
す。こうすることで以前の開発記録を簡単に閲覧することができます

次にプッシュ

変更履歴をリモートリポジトリにアップロードすること。ローカルで作業した内容をプッ  
シュすることでリモートリポジトリに反映されます。その日の作業が終わったときなんか  
にしたりする

次クローン

リモートリポジトリをローカルリポジトリに複製することです。これを行うことでローカ  
ルで作業できるようになる。新たな開発に着手する時等に行う

次プル

リモートからローカルに変更履歴をダウンロードすること。リモートから最新の変更履歴  
をダウンロードする。

ここで注意してほしいのは、クローンはリポジトリそのものを持ってくるけどプルは自分  
のリポジトリに足りてない変更履歴だけを持ってくる

最後にブランチ

履歴の流れを分岐して記録するGitの機能。こんな風に作業するフィールドごとにブラン  
チを分けることでお互い影響せずに作業することができる。バージョン管理の説明の時に  
言った'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'  
のはこれのおかげです。

これで事前知識は終わり

じゃあいざGitHubを導入しよう！となって使い方などを調べているとほぼ必ず「GitBash  
をつかってー（または単にGitを使って）」とあります

でもGitBashロボコンで使う上では重大な欠点があります。それは、まず一つコマンド操  
作です。コマンド覚えないういけません。なので慣れるのに時間がかかります。人による  
だろうけど僕は時間かかってます。今勉強中です。そして視覚的にわかりにくい。こんな  
文字が無機質に並んでるだけの画面と対峙する必要があります。これは起動してプルして  
る時の画像ですね。ロボコンしてる人ってGitHub使えとは言ってもメインでやらない

けないのは開発なので、こういうのはめんどくさがってやらないと思うんですよ。  
なのでGitBash使いません。使いたい人は個人で頑張ってください。現状ロボコンするう  
えでは必要ないです。企業とかでは使ってるところもあるっぽいので、頑張って損はない  
と思いますが

じゃ実際にGitHub使っていきます。GitHubアカウント取得できてる人は見ながら一緒にや  
ってみるといい。アカウントの作り方から説明するのはやりません。どうしても出来なか  
ったらわかる人に後で聞いてください。  
で、またここで問題があります。GitBashを使わずGitHubだけで作業した場合好きなエデ  
ィターで編集ができません。実際にやって見せて。本来Gitを使えばそんなことはなく、  
好きなエディターで書ける。じゃあどうするか。GitHubDesktop使います。これの利点は

あ  
あ

あ  
があります。で、今日はロボコンで使う上での最低限のことしか教えないんで、いきなり  
こっち使っていきます。  
リポジトリ作る。  
ブランチ作る  
ファイル作ったり  
プル陸マージふえ t c hとか  
まあなんかいろいろやって見せる

end