J23375

細井 琉吾

GitHubについて

1. GitHubとは

GitHubとは、Gitで管理されたファイルをネット上にアップロードして取り扱うことができるプラットフォームのことです。GitHubを使えば、Gitで管理しているコードを複数人で管理・運用するのがとても楽になります。

1. GitHubの初期設定

GitHubを使う前に、まずGitを公式サイトからインストールしておきます。インストールができたら、GitHub公式サイトにアクセスしてアカウントを作成します。

アカウントの作成が完了したら、二要素認証を設定しておきましょう。右上のアイコンをクリックし、Settingsをクリックします。その後、左側にあるPassword and authenticationをクリックして二要素認証の設定を行います。スマホに任意の二要素認証用のアプリをダウンロードし、GitHubと連携させ、二要素認証の設定を完了させます。

次に、Personal access tokenを作成します。先ほどと同じように右上のアイコンをクリックし、Settingsをクリックします。その後、左側の下の方にあるDeveloper settingsをクリックします。そして、左側にあるPersonal access tokensを開き、Token(classic)をクリックします。Generate new tokenもclassicを選択し、任意の名前と期限を入力します。Select scopesはrepoにチェックを入れ、tokenの作成を完了させます。

以上で初期設定は完了です。

3.　 GitHubでのプルからコミットまでの流れ

まず、リモートリポジトリの最新の状態をローカルに反映させるためにプルを行います。[git pull origin ブランチ名]と入力します。



次に別のブランチで作業を行うために、新しいブランチを作成しましょう。

[git checkout -b 新しいブランチ名]



そして、任意のエディタを使用し、ソースコードやファイルを追加・変更します。編集が完了したら、Gitの管理下に追加するためにステージングを行います。全ての変更を対象にする場合は[git add .]、特定のファイルのみの場合は[git add ファイル名]。



最後に、変更されたものを１つのコミットとして保存します。

[git commit -m “変更した内容の説明”]



以上がプルからコミットまでの一連の流れです。

4．コンフリクト

他のブランチで変更した結果を取り込む際、他の人と同じファイルを修正していた場合に互いの変更がぶつかり合い、処理を実行できない場合があります。これをコンフリクトといいます。コンフリクトが発生する原因はいくつかあります。

１つ目は、同じ行の変更をした場合です。複数のブランチで同じファイルの同じ行を異なる内容で変更した場合に発生します。

２つ目は、同じファイルの変更をした場合です。同じファイルが異なるブランチで変更された場合でも、変更箇所が異なればマージすることができるのですが、変更内容に依存関係があったり、同じ部分で変更が重なる場合に発生します。

３つ目は、ファイルの追加、削除、変更が競合した場合です。一方のブランチでファイルを削除したのに、もう一方でそのファイルに変更を加えていた場合や、一方のブランチでファイルを追加したのに、もう一方でそのファイルを削除した場合などに発生します。

４つ目は、同じファイルを移動、または名前を変更した場合です。同じファイルが異なるブランチで移動、または名前を変更されて、両方の変更がマージされた場合に発生します。

コンフリクトを解消するには、コンフリクトが発生しているファイルを確認し、競合している部分を手動で変更する必要があります。どの変更を採用するか、あるいは両方を組み合わせるかを判断して修正しましょう。

5. まとめ

　GitHubは複数人でのコードの管理がとても楽になるプラットフォームです。正しい使い方を覚えて賢く使いましょう。

参考文献

PythonプログラミングVTuber サプー.” 【GitHub入門】初心者向け！GitHubでチーム開発するための基本操作を解説！”.

<https://www.youtube.com/watch?v=Dz95iUNt-fg&t=1s>

@taishi\_will\_be.”【超初心者向け】Github基礎のpush/pullについて解説”.Qiita.com.

<https://qiita.com/taishi_will_be/items/13f00ddf40311337583b>

OJK.” VSCodeでGithub（の超基本） Chapter 04プル→コミット→プッシュの基本的な流れ”.Zenn.

<https://zenn.dev/ojk/books/github-vscode/viewer/pull-push>

@tgc\_nuco in 株式会社Nuco.”【徹底解説】これを読めばもう怖くないGitコンフリクト”.Qiita.

<https://qiita.com/tgc_nuco/items/c65f36e15243535c095b>