

【Git】初期設定

前提知識

まずはGitの初期設定を行っていく前に必要な前提知識を付けていきましょう。

Gitを操作する際には、主に2つの選択肢があります。

1. CLI(Command Line Interface)を使った操作
2. GUI(Graphical User Interface)を使った操作

CLIとは？

CLIとは、コマンドラインツールとも呼ばれるもののことで、キーボードでコマンドを打つことによってファイルやフォルダの操作をするためのものです。

映画やドラマでハッカーがハッキングをする際に操作している黒い画面のことをイメージして貰えばいいと思います。
macOSの入ったPCでは、標準で「ターミナル.app(Terminal.app)」というアプリがインストールされていますが、これがCLI(コマンドラインツール)となります。



GUIとは？

GUIとは、グラフィカルユーザーインターフェースのことで、普段PCを使って使用している多くのツールはこれに該当します。

特徴としては、カーソルを使用してドラック&ドロップをしたり、Control+クリック(右クリック)でメニューを表示したりして操作ができるツールのことです。

これは幅広いツールが対象になりますが、例えば「[Google Chrome](#)」や「[Adobe XD](#)」などもGUIツールと言えます。

CLIと比べた場合に「ファイルやフォルダの操作をする」という目的においては、macOS標準の「ファインダー(Finder)」アプリなどが比較対象となります。

ファインダーもターミナルと同様にファイルとフォルダの操作が出来ますが、こちらの場合はカーソルやメニューを使用した操作をするといった違いがあります。



また、Gitの操作ができるGUIとしては「[GitHub Desktop](#)」や「[Sourcetree](#)」などのアプリケーションがあります。これはカーソルやメニューを使ってGitの操作をすることができるツールです。



Desktop



Sourcetree

Gitの操作はCLIとGUIどちらで行うべき？

Gitを使用する場合は、基本的にCLI(ターミナルなど)とGUI(Github Desktopなど)のどちらでも操作することができるため、ど

ちらを使っても構いません。

実際の開発現場では、(著者の体感として)CLIとGUIを使用している人の割合は半々の様に思います。
(※統計的な数値を集計した訳ではなく、あくまで個人的主観です。)

ですが、以降の記事ではCLIを使って操作することを前提としているため、そちらだけご注意ください。

Gitの初期設定

Gitのリポジトリを作成

Gitの世界では、フォルダのことを「リポジトリ(Repository)」と呼びます。(※1)
これはGitを使って管理する対象ファイルやフォルダのことを示しており、Gitを使って管理するソースコードは全てこのリポジトリ配下(フォルダの中)に格納するのがルールとなっています。

そのため、Gitを学ぶためにはまずリポジトリを作成する必要があります。
そこで、macOSに標準でインストールされている「ターミナル」を使って、リポジトリの作成を行いましょう。
(※以下CLIコマンドの使い方も含めて解説していきます。)

まずはターミナルを立ちげて、下記のコマンドを入力して実行してみます。

command

\$ ls

そうすると、以下の様な出力になるかと思います。

output

Applications Documents Library Music Pictures
Desktop Downloads Movies Public

この「ls」コマンドは、現在のフォルダ配下にあるファイルやフォルダの一覧を表示するためのコマンドです。
では「現在のフォルダ」がどこなのかも確認するために、今度は下記のコマンドを実行してみましょう。

command

\$ pwd

output

User/yamadaku

そうすると、上記のようにUser/yamatakuと出力されました。

これはこのPCでログインしている「yamataku」というユーザーのフォルダにいる、ということを意味しています。
この「yamataku」の部分は、実際にはログインしているユーザー名になるはずです。

つまり、今現在立っている`User/yamataku`というディレクトリ(=フォルダの階層)の配下には、`Applications`や`Documents`というフォルダが存在しているということが分かりました。

フォルダのディレクトリツリー(階層図)をテキストで書くと、以下の様なイメージになります。

```
directory
└── User
    ├── Yamataku
    │   ├── Applications
    │   ├── Documents
    │   ├── Library
    │   ├── Music
    │   ├── Pictures
    │   ├── Desktop
    │   ├── Downloads
    │   ├── Movies
    │   └── Public
    └── Public
```

ではディレクトリが分かったところで、さっそくGit学習用のフォルダ(リポジトリ)を作成していきましょう。
新しいフォルダを作成する場合は、`mkdir`コマンドを使用します。

```
command
$ mkdir Git
```

この時、`mkdir`の後に半角スペースを空けてフォルダ名を書くことで、その場所にフォルダが作成させます。
試しにもう一度フォルダ内の状況を確認してみましょう。

```
output
Applications Documents Git Movies Public
Desktop Downloads Library Music Pictures
```

そうすると`Git`というフォルダが正常作成されていることが確認できました。
CLIツールでは、この様にコマンドを使ってファイルやフォルダの操作を行っていくのが基本となります。

リポジトリに移動してGitを初期化

では次に今作成したGitフォルダへ移動をしてGitの初期化を行っていきます。

まずディレクトリ(=フォルダの階層)を移動する際には、「`cd`」というコマンドを使用します。

```
command
$ cd Git
```

上記の様に`cd`の後に半角スペースを空けてフォルダ名を書くことで、その対象フォルダのディレクトリへ移動することができます。

このコマンドを実行した後、現在のディレクトリを確認してみましょう。

```
command
```

```
$ pwd
```

```
output
```

```
/Users/yamataku/Git
```

すると、上記の様にGitフォルダへ移動できた事が分かりました。
ではさっそくここでGitの初期化を行いましょう。

```
command
```

```
$ git init
```

```
output
```

```
Initialized empty Git repository in /Users/yamataku/Git/.git/
```

Gitの初期化コマンドはgitの後に半角スペースを空けてinitとすることで、「Git」の「initialize」(初期化)ができます。
出力結果にも正常にリポジトリが初期化されたことが書かれていますね。

Gitのコマンドを使用する場合は、必ず最初にgitと書いた後に半角スペースを空けて任意のコマンドを入力する事がルールとなっています。

ここでもう一度現在のフォルダ状況の確認を行いましょう。

```
command
```

```
$ ls -a
```

```
output
```

```
.  ..  .git
```

この時、lsコマンドの後に-aという項目を追加しましたが、これはコマンドの「オプション」のことを意味しています。
この-aというオプションは、隠しフォルダや隠しファイルを表示するためのオプションです。

隠しフォルダとは、通常であれば表示や閲覧をしない想定 of ファイルやフォルダのことですが、Gitではこの隠しファイルと隠しフォルダの中にバージョン管理をするためのソースコードが格納されています。

普段の開発ではあまり意識することはありませんが、何かの設定をしたりする場合に使用することがあるため、頭の片隅に入れておいてください。

Gitで使用するアカウントの設定

今回はもう一つ初期設定としてやっておくべきことがあります。
それはGitで使用するユーザーアカウントの設定です。

Gitの世界では、「誰が」どんな作業をいつ行ったのかの記録(ログ)をとっているため、その記録の中に自分の情報を登録できるようにする必要があります。

Gitで使用するアカウント設定をする場合は、`git config`というコマンドを使用しましょう。

command

```
$ git config --global user.name 'YOUR NAME'
```

この時、コマンドの後に`--global`を追加しています。

このオプションは、現在PCにログインしているユーザーアカウントに対する設定を行う。という意味合いを持ちますが、最初
はあまり意識しすぎなくても大丈夫です。

このオプションに続いて、`user.name 'YOUR NAME'`と書いていますが、これはシングルクォーテーション(”)の中の文字列を
ユーザー名として登録する。という意味になります。

(※`YOUR NAME`の文字列は自分の名前で書き換えてくださいね。名前は英数字で表現することが一般的です。)

次にメールアドレスの登録もおきましょう。

command

```
$ git config --global user.email 'yourmail@example.com'
```

メールアドレスを登録する場合は、この様に`--global`オプションの後に続いて`user.email 'yourmail@example.com'`と
書けばOKです。

(※`'yourmail@example.com'`は実際のメールアドレスに書き換えてください。)

ついでにターミナル上でGitコマンドに色を付ける設定もおきましょう。

command

```
$ git config --global color.ui true
```

この`color.ui`に対して`true`を指定することで、ターミナル上のコマンドを色を付けることが出来ます。

これでGitで使用するアカウント設定も完了したので、Gitを使う準備が整いました。

注釈

- ※1: 厳密にはリポジトリ≒作業場所なので、ファイルの集合体を示すフォルダと異なる意味合いも持つので異なる概念もあります。

← 教材一覧へ戻る