

■Git のダウンロード

1. Git で検索
2. Downloads をクリック
3. Windows をクリック
4. 64bit Git for Windowssetup をクリック

■GitHub アカウントの作成

1. github で検索
2. github へ登録
 - ①E-mail の入力
 - ②パスワードの入力
 - ③UserName の入力
3. ロボットでないことの確認を実施
4. メールアドレスに code が送信されるので確認して入力
5. いくつかの質問は適当に答えて、無料版を選択
6. dev リポジトリを作成
7. 右側「Setting」メニューをクリック
 - 左側から SSH and GPG Keys をクリック
 - 右上の NewSSHKey をクリック
8. C:\Users\ユーザ名\.ssh から、id_rsa.pub をコピーして貼り付ける
9. gitbash にて、「ssh -T [git@github.com](https://github.com)」を実行
 - 自分のユーザ名が出てくれば OK

■git の初期設定

★名前とメールアドレスを登録

1. git config --global user.name "Uryu Soichiro"
2. git config --global user.email "soichirouryu@di-system.co.jp"

★git リポジトリを作成

→リポジトリとは、ファイルやディレクトリの状態、ファイルの変更履歴を記録する場所
「git init」 コマンドでローカルリポジトリを作成

※「.git」ディレクトリが作成され、この中にバージョン管理のための各種情報が格納される。

3. BOX からダウンロードしたプロジェクトを C ドライブ直下に配置
4. gitbash から cd c:¥システム開発演習
5. 4. のフォルダ内に sample.txt を作成
6. git init で git リポジトリを作成
7. git add .
8. git commit
9. github でリポジトリを作成
10. git add .
11. git commit [init project]

■Git で使える Linux コマンド

cd…ディレクトリの変更を行う

ls…ディレクトリの内容を表示「-a」 オプションで隠しファイルを含めて表示

mkdir…新規ディレクトリの作成

rm…ファイルの削除

cp…ファイルのコピー

mv…ファイルの移動とリネーム

■Git での基本的なワークフロー

①ファイルの変更をステージングエリアへ追記する。 (「git add」 コマンド)

②ローカルリポジトリにコミットする (「git commit」 コマンド)

→コミットコメントを記述する。

エディタの使い方が特殊なので注意！

まず、半角英数字入力モードに切り替えて「i」を押す。

入力可能になるので、コミットコメントを記述する。

「Esq」キーを押し「:wq」と入力し、エンターキーを押す

③リモートリポジトリにプッシュする (「git push」 コマンド)

→GitHub で作ったリポジトリ内にコマンドがあるので、それを Git で実行

「git remote add origin [https://github.com/~/~](https://github.com/~/)」

→GitHub (リモートリポジトリ) にローカルリポジトリを表示

「git push -u origin master」

→ローカルリポジトリからリモートレポジトリに送信

■Git の基本的な使い方

★変更のコミット

①ファイルの変更をステージングエリアへ追記する。 (「git add」 コマンド)

※ステージングエリアとは、コミットするファイルを選択する場所のこと

コミット作業は1作業ずつ行うことが原則「git add ファイル名」

「git add .」で変更があるすべてのファイルをステージングエリアに追加可能

②ローカルリポジトリにコミットする (「git commit」 「git commit -v」 コマンド)

→変更メッセージをつけてローカルリポジトリに記録するのがコミット

コミットすることで、変更がリポジトリ内に時系列で記録される。

※コミットメッセージは分かりやすく！

何をどうして変更したかを分かりやすく表記する。

==おすすめ==

1行目：変更内容の要約

2行目：空行

3行目：変更理由

③「git log」を使用すると、コミットの履歴が確認できる。

★削除のコミット

- ① 適当にローカルリポジトリ内にファイルを作成 (sample2.txt)
- ② ステージングエリアに追加してコミット
- ③ `ls→rm sample2.txt →ls` で sample2.txt をローカルから削除
- ④ 削除をステージングエリアに追加するためには、「`git rm ファイル名`」を使う
- ⑤ `git commit`

★複数のファイルのコミット

- ① 「`git add .`」
- ② 「`git commit`」

★ローカルリポジトリの現在の状況を確認するコマンド

「`git status`」

…最後のコミットから現在までの変更分を表示できる (よく使う)

ステージ挙げていない変更 (赤字)

ステージに挙がっているがコミットされていない変更 (緑字)

★git の変更履歴の確認

「`git log`」…リポジトリにコミットされたログを確認する

j で下移動、k で上移動、q で終了

「`git log -n 3`」…最新の3コミット分だけを確認できる

「`git log --oneline`」一行で表示する

「`git log -p ファイル名`」ファイルの差分を表示する。

※ファイル名の指定はしなくてもよい

★ファイルの変更差分を確認

「`git diff`」ファイルの変更差分を確認できる

add や commit する前に確認しよう

「`git diff`」…ステージと変更ファイルの差分

「`git diff HEAD`」…ステージとコミットの差分