

GitHub のアカウント作成と Git CUI のセットアップ(Mac)

Hack U Project

目次

- Git / GitHubとは
- **事前準備 1** GitHub のセットアップ
- **事前準備 2** Git のセットアップ
- **事前準備 3** 演習準備
- **事前準備 4** GitHub への疎通確認

Git / GitHubとは

Git / GitHub とは

Git	バージョン管理※の方式
GitHub	Gitでバージョン管理をしやすくする Webサービス ブラウザから変更履歴やコードを共有 することが簡単にできる

※バージョン管理：いつ、誰が、どのファイルを
どう変更したかを記録すること

Git を使うには

主に以下の2つの方法がある

- コマンドラインでGitコマンドを使う
- GUIツールを使う

今回は コマンドラインで
Gitコマンド を使います

事前準備 1

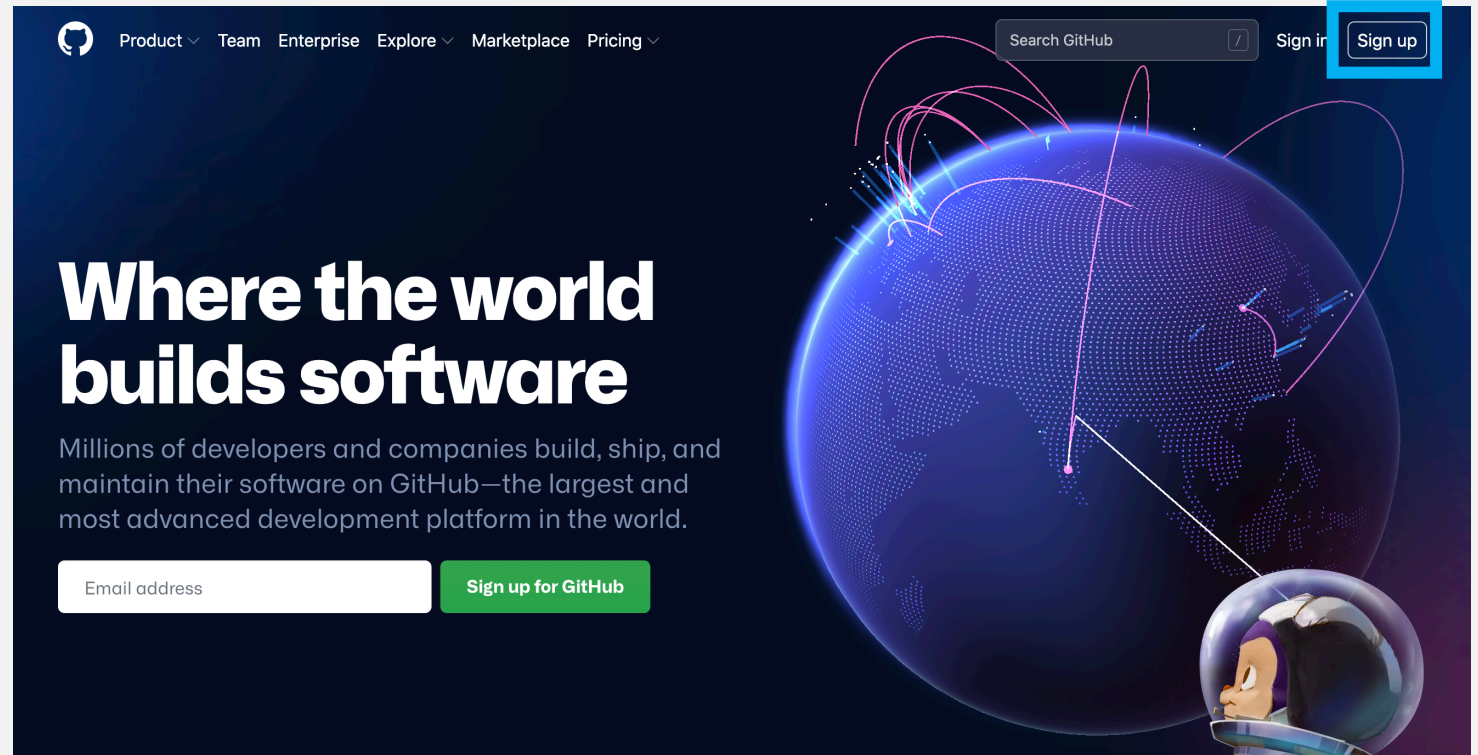
GitHub のセットアップ

GitHubのセットアップ

- GitHubを利用するためのアカウント作成を行います
- GitHub上でファイルを配置するための
リモートリポジトリを作成します

GitHub のアカウントを作る

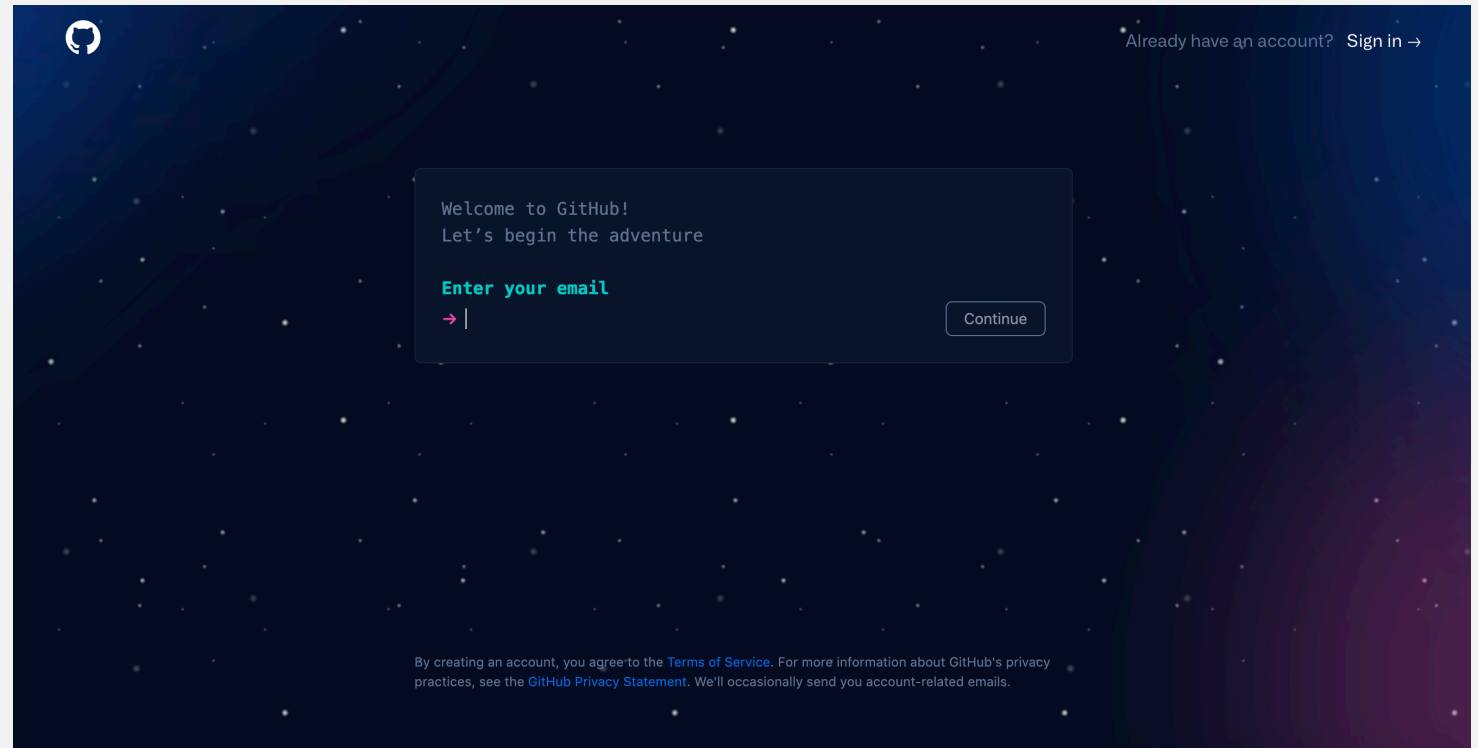
- <https://github.com/> にアクセスする
- 右上の「Sign up」をクリックする



GitHub のアカウントを作る

- 画面の案内に沿って必要事項を入力する
 - メールアドレス
 - パスワード
 - アカウント名 (username)

※アカウント名は公開
されます



GitHub のアカウントを作る

- アカウントの検証→「検証する」
- 「Create account」ボタンを押す
- メールアドレスに送信されたコードを入力
- Welcome to GitHub
(GitHubからのアンケートに答える)
- 「Free」プランを選択

GitHub のアカウントを確認する

- プロフィールページにアクセス
 - `https://github.com/アカウント名`

上記ページが表示されればアカウント作成完了です
アカウント作成されていない場合、404になります

リモートリポジトリを作る


- プロフィールページから「Create repository」をクリック

Owner には先程作った
アカウント名が入る

Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)


Owner * Repository name *

 アカウント名 /

Great repository names are short and memorable. Need inspiration? How about [scaling-eureka?](#)

Description (optional)

☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

☐ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer.

Add .gitignore: **None** ▼ | Add a license: **None** ▼ ⓘ

Create repository


「**github_ws**」という
Repository nameで
作りましょう

リモートリポジトリを作る

Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)


Owner * Repository name *

 アカウント名 /

Great repository names are short and memorable. Need inspiration? How about [scaling-eureka?](#)

Description (optional)

☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

☐ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer.

Add .gitignore: **None** ▼ Add a license: **None** ▼ [i](#)

Create repository

リポジトリを誰でも参照
可能にするかを設定する
(今回は **public**)

事前準備 2

Git CUI のセットアップ

Git CUIのセットアップ

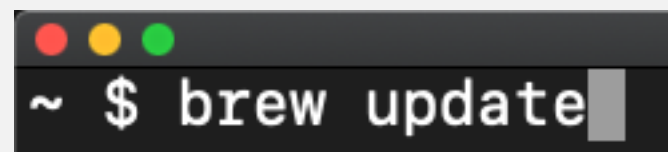
- ローカルのPCに Git をインストールします
- Git を使うための初期設定を行います

Git CUIのインストール

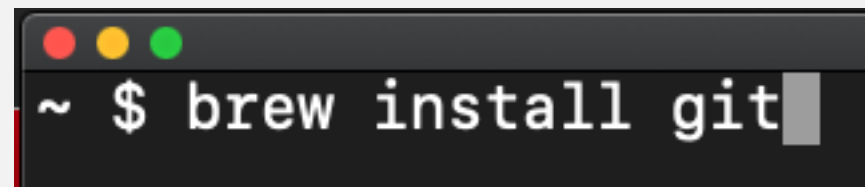
brewを使って最新のgitをインストールします

※brewを入れていない場合はインストールしてください

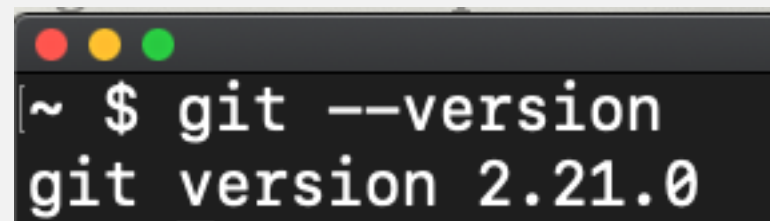
1. ターミナルを開く
2. `$ brew update` 実行
3. `$ brew install git` 実行
4. `$ git --version` で確認



```
~ $ brew update
```



```
~ $ brew install git
```



```
~ $ git --version  
git version 2.21.0
```

※git_versionはテキストと異なっても大丈夫です

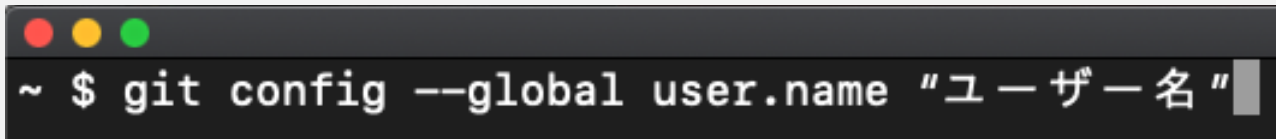
Git CUIの初期設定

ユーザー名とメールアドレスを登録する

※これはGit上での区別に使われるもののため、必ずしもGitHubと合わせる必要はありません

- `$ git config --global user.name "ユーザー名"`

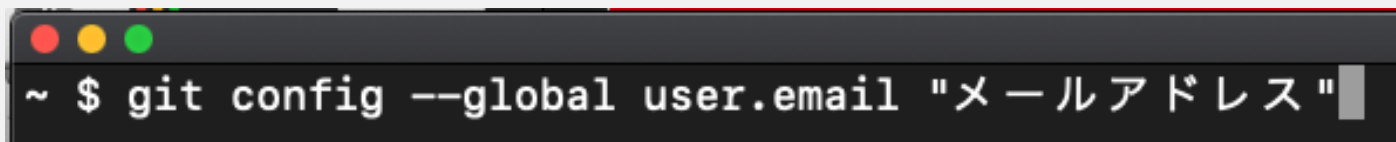
でユーザー名を登録



```
~ $ git config --global user.name "ユーザー名"
```

- `$ git config --global user.name "メールアドレス"`

でメールアドレスを登録



```
~ $ git config --global user.email "メールアドレス"
```

事前準備 3

演習準備

ローカルリポジトリの作成

- Git の機能の一つである **clone** を使って、GitHub で作成した**リモートリポジトリ**をローカルにコピーします
- 各Gitの機能やリポジトリについては演習の資料で解説します

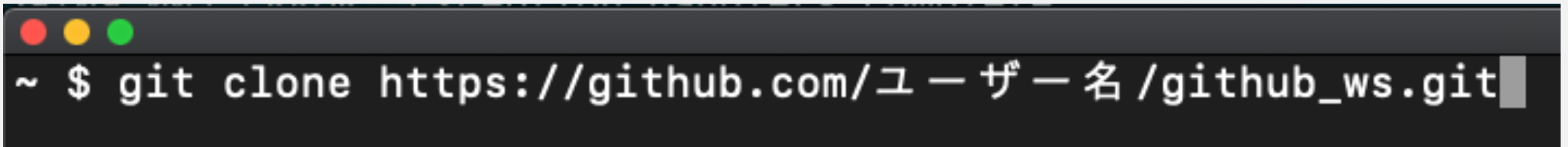
ローカルリポジトリの作成

1. 適当なディレクトリに移動する

※ 「git」 や 「workspace」 などのディレクトリ下に clone するとリポジトリが管理しやすいです

2. clone のコマンドを打つ

```
$ git clone https://github.com/ユーザー名/github_ws.git
```

A terminal window with a dark background and light gray text. The prompt is '~ \$' and the command being entered is 'git clone https://github.com/ユーザー名/github_ws.git'. The cursor is at the end of the command.

```
~ $ git clone https://github.com/ユーザー名/github_ws.git
```

ローカルリポジトリの作成

3. Finderでcloneしてきたディレクトリを開き、
空のディレクトリができていることを確認

※エディターなどを使っている場合はそれで確認してもOKです

※Finderで開くコマンド

```
$ open github_ws
```

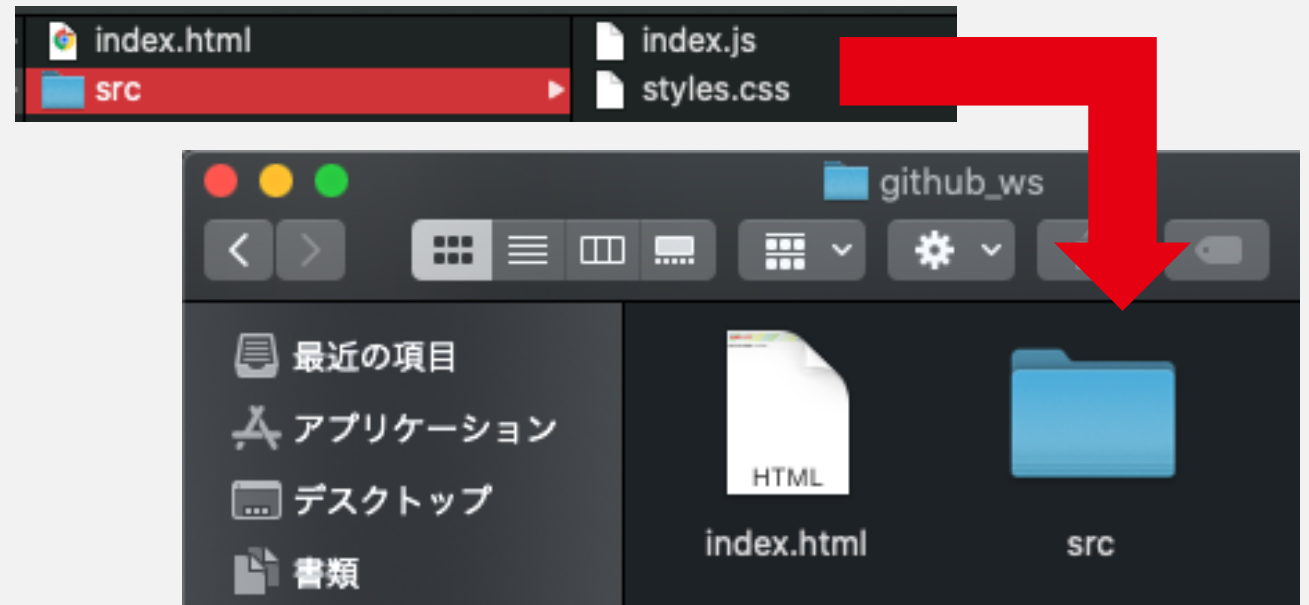


演習ファイルの準備

1. https://github.com/hackujp/github_tutorialを開く
2. リポジトリ内のCUI/hands-on.zipを開き、Downloadボタンよりダウンロード
3. Hands-on.zipを解凍

演習ファイルの準備

4. 先程開いたローカルリポジトリに展開したファイルを移動させる
5. 移動させる対象は以下の通り
 - Index.html
 - src



事前準備 4

GitHubへの疎通確認

GitHubへの疎通確認

- GitHub に演習ファイルをアップロードして疎通確認を行います
- Git の基本操作 `add/commit/push` を使用します

GitHubへの疎通確認 (add)

- `$ git status` で追加した演習ファイルが表示されていることを確認してください。

```
~/github/github_ws $ git status
On branch master
Your branch is up to date with 'origin/master'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)

    index.html
    src/

nothing added to commit but untracked files present (use "git add" to track)
```

GitHubへの疎通確認 (add)

- `$ git add .`（最後にピリオドが必要です）でaddされます。`$ git status` でnew fileとして表示されることを確認してください

```
~/github/github_ws $ git add .
~/github/github_ws $ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   index.html
    new file:   src/index.js
    new file:   src/styles.css
```

GitHubへの疎通確認 (commit)

- `$ git commit -m "<任意のメッセージ>"`
でコミットされます。
- 3つのファイルが表示されることを確認してください。

```
[~/github/github_ws $ git commit -m "first commit"]
[master d55fbdf] first commit
3 files changed, 50 insertions(+)
create mode 100644 index.html
create mode 100644 src/index.js
create mode 100644 src/styles.css
```

GitHubへの疎通確認 (push)

- `$ git push origin master` でpushします

※ ブランチがmainの場合はmaster -> main。

commit時の表示を確認してください

```
~/github/github_ws $ git push origin master
```

- Githubのユーザー名とパスワードを入力する必要があります。

GitHub に反映されているか確認

The screenshot shows the GitHub interface for a repository named 'github_ws' owned by 'ユーザー名' (username). The top navigation bar includes the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Explore. The repository header shows the name 'ユーザー名 / github_ws', a search bar, and buttons for Unwatch (1), Star (0), and Fork (0). Below the header, there are tabs for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, and Insights. The 'Code' tab is active, showing a file browser with a commit history table. The table lists a commit 'Add index.html & src' by 'ユーザー名' with hash '50cdc45' and timestamp '14 minutes ago', and 1 commit. Below the table, there are buttons for 'Go to file', 'Add file', and 'Code'. A blue box at the bottom prompts the user to 'Add a README'. On the right side, the 'About' section states 'No description, website, or topics provided.' and the 'Releases' section states 'No releases published' with a link to 'Create a new release'.

GitHub logo Search or jump to... Pull requests Issues Marketplace Explore

ユーザー名 / github_ws Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights

master 1 branch 0 tags Go to file Add file Code

ユーザー名	Add index.html & src	50cdc45	14 minutes ago	1 commits
src	Add index.html & src		14 minutes ago	
index.html	Add index.html & src		14 minutes ago	

Help people interested in this repository understand your project by adding a README. Add a README

About No description, website, or topics provided.

Releases No releases published Create a new release

Pushがうまくいかない時

- ネットワークの問題
 - 学校のWifiなどを使用していると、ネットワークの設定で制限がかけられ、Pushができない場合があります。
 - 別のネットワークやスマホのテザリングなどで再度試してみましょう

**事前準備は以上となります
お疲れさまでした**