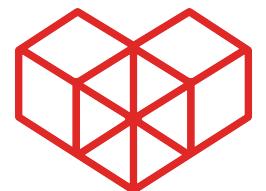


Git, GitHub とは？



kintone
developer program



コンテンツの目的

SC本部メンバーが。。。。

- 1 GitとGitHubについて理解できた**
- 2 GitHubでコードを管理できるようになった**



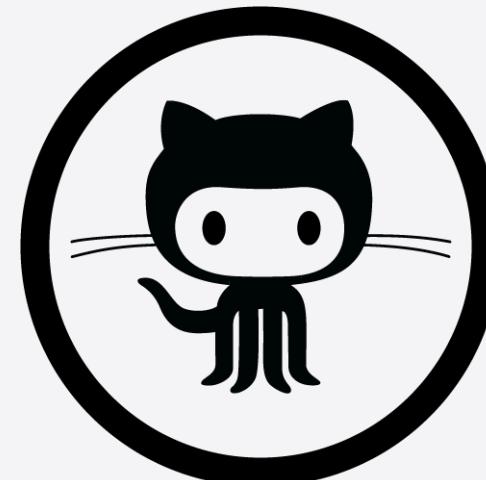
Overview

Gitの基本



GitHub Icon

GitHubの基本



GitHub Icon

ハンズオン

- Gitをインストールする。
- GitHubアカウントを作成する
- リポジトリを作成する

Git とは...



- Git とは、ソフトウェア開発のための**バージョン管理**を提供するソフトウェアです。
- Linuxカーネル開発を管理するツールとして2005年に開始。
- 無料で使えるオープンソースの分散バージョン管理システム。

バージョン管理とは？



ゲームのセーブポイント



履歴

時間とともに加えられていく変更を記録することができます。



チームワーク

他のシステムを使う開発者と共同作業をすることができます。



バックアップ

後で特定バージョンを呼び出すことができるようになります。

バージョン管理の例

Google Docs

← Today, 13:39

100% Total: 1 edit

What is Git & GitHub?

- First, What is Version Control?**
 - Similar to a game's checkpoint system. You save your current so if you were to miss up in the next level, you do not start all back to where you last saved your spot.
 - Version Control keeps track of all the changes you have made time.
 - Allows for multiple people working on the same project at the same time.
 - Allows for easy file recovery
- Examples of Version Control?
 - Google Docs
 - Dropbox
- What is Git?
 - Git is the actual software that provides the Version Control

Version history

Only show named versions

TODAY

- 10 April, 13:39 Current version Genji Fujimori
- 10 April, 09:49 Genji Fujimori

WEDNESDAY

- 8 April, 17:37 Genji Fujimori

TUESDAY

- 7 April, 18:36 Genji Fujimori
- 7 April, 15:02 Genji Fujimori

LAST WEEK

Kintone

Space: US Kintone Developer Program (KDP) App: KDP Swag View: In stock Record: Popsockets

Item name: Popsockets

Photo: 

Link: <https://www.ipromo.com/promo-products/20072/PopSockets.html>

Notes: Despite the white face, surprisingly has not gotten dirty!

Brought to Le Wagon: 23 Tokyo Stock as of Jan 31 = 13

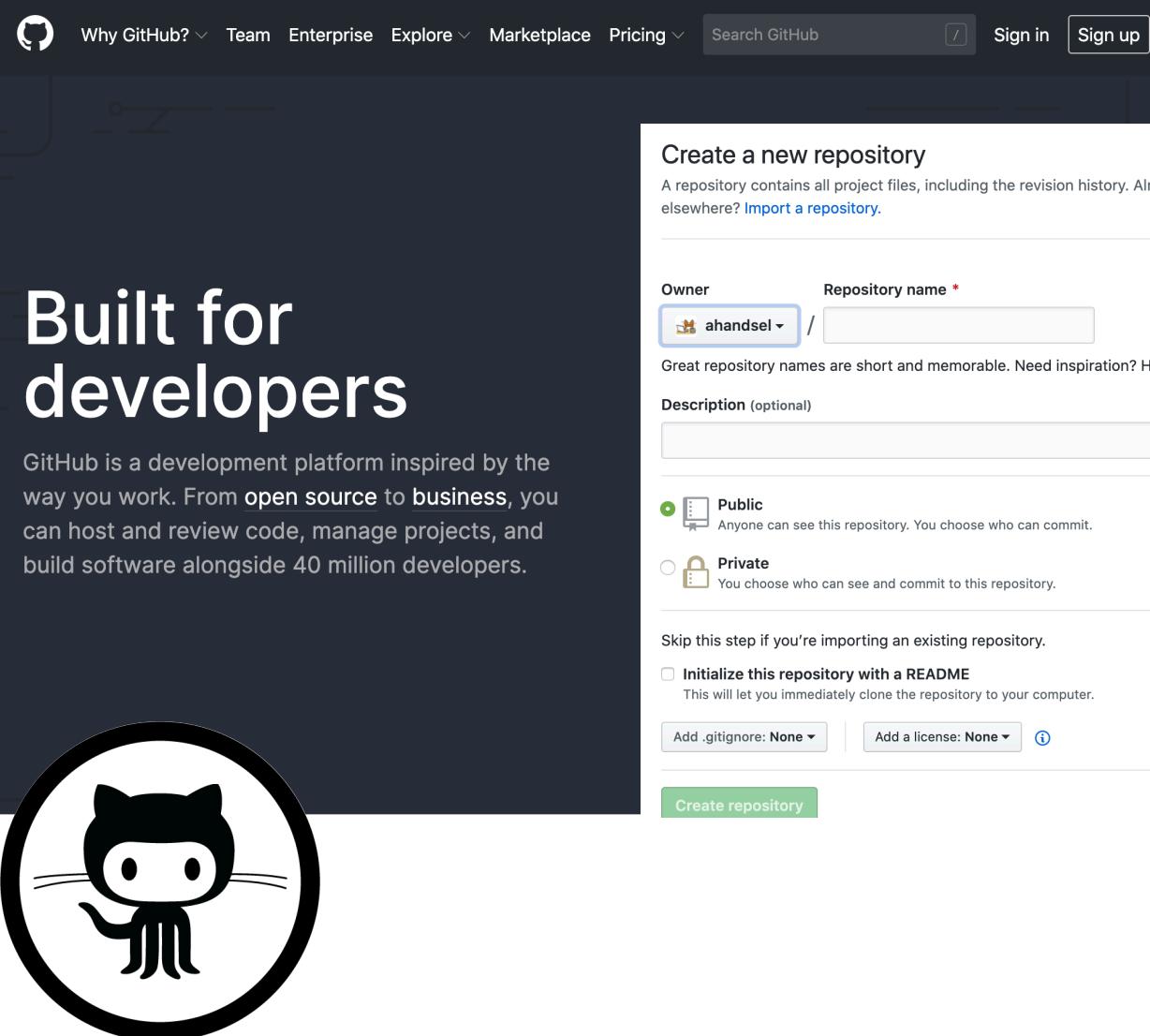
History

Show Latest Contents

24: Genji Fujimori Feb 27, 2020 7:30
Total Stock: Changed from "52" to "43"
After: Entered "1"
Given Out: Changed from "-10" to "-9"
Stock: Changed from "10" to "1"
[Show Update Details](#)
[Restore Previous Version](#)

23: Genji Fujimori Feb 22, 2020 6:40
Deleted or added one or more rows in a table.
Event Date: Specified "Feb 21, 2020"
Event Name: Entered "FEMhack"
Before: Entered "10"
After: Updated the value.
Given Out: Entered "-10"
Date: Specified "Feb 21, 2020"
Place: Selected SF Current Stock
Stock: Entered "10"
[Show | Update Details](#)

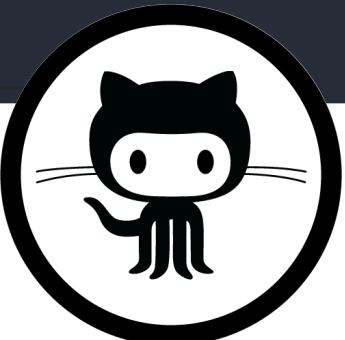
GitHubとは...



The screenshot shows the GitHub homepage. At the top, there's a navigation bar with links for "Why GitHub?", "Team", "Enterprise", "Explore", "Marketplace", "Pricing", "Sign in", and "Sign up". Below the navigation is a search bar labeled "Search GitHub". The main content area features a large "Create a new repository" form. It includes fields for "Owner" (set to "ahansel"), "Repository name" (a required field), and "Description (optional)". There are two radio button options for visibility: "Public" (selected) and "Private". Below these are sections for "Skip this step if you're importing an existing repository" and "Initialize this repository with a README". At the bottom of the form is a green "Create repository" button.

Built for developers

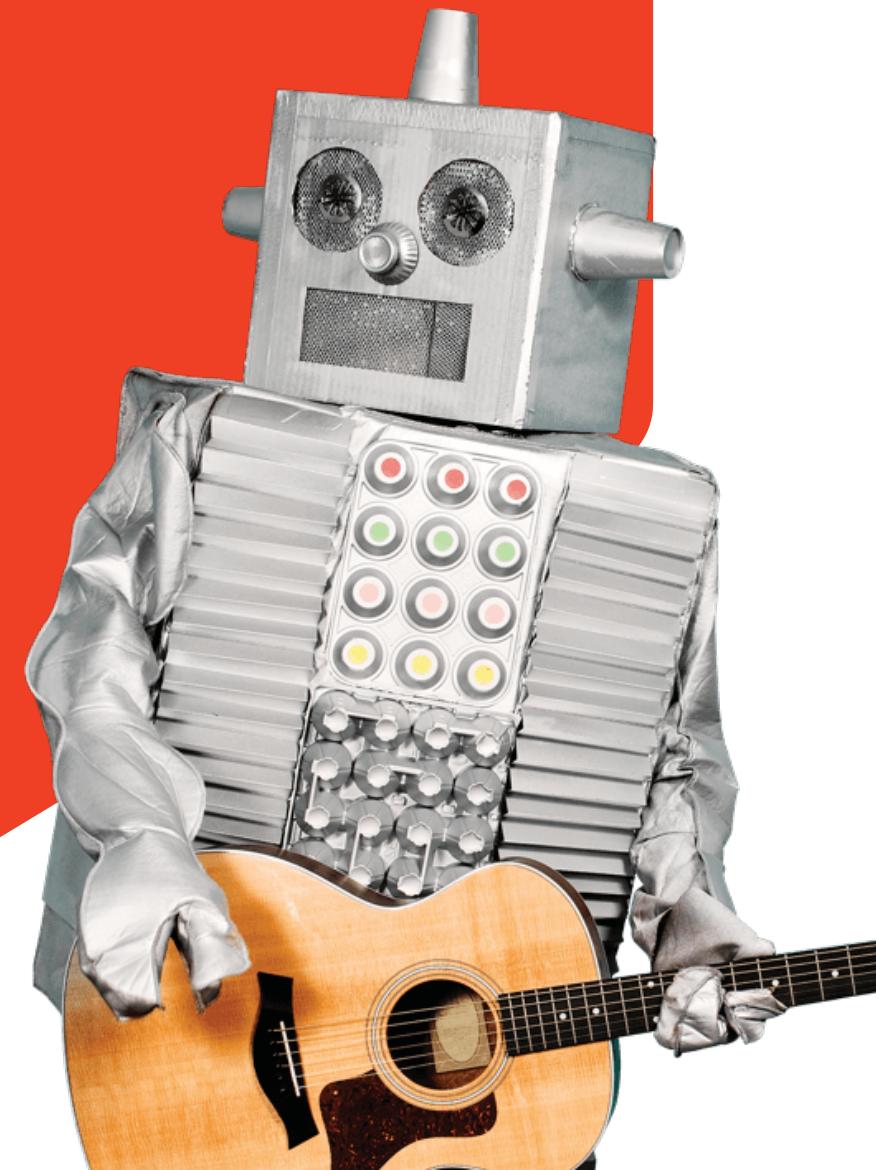
GitHub is a development platform inspired by the way you work. From **open source** to **business**, you can host and review code, manage projects, and build software alongside 40 million developers.



- GitHub.comとは共同開発プラットフォームです
- コードを見たり見せたりできる場所です。
 - Google Docs とちょっと似ていて、いろんな人たちがコードを同時に見たり編集できます。
 - "Remote repositories" 「リモートリポジトリ」と聞いたら、GitHubだなっと考えてください。
- 2008年に設立され、現在はマイクロソフトの子会社です。

使い始める

Hands-on A





Hands-On A - Overview

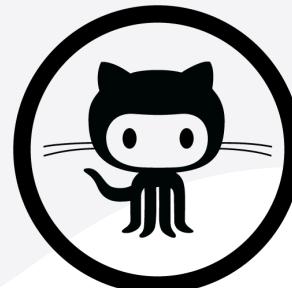
ローカルでの設定

- フォルダを作成する。
- フォルダをgitで管理するように構成する。



GitHubの設定

- GitHubで「フォルダ」のようなものを作成する。
- Repository（リポジトリ）と呼ばれています。



ローカルとGitHubを接続する

- 接続するように2つを構成する。
- ローカル側でファイルを作成してgitコマンドを実行すると、ファイルはGitHub.comに表示されます。

ローカルGitリポジトリを作成する

```
$ cd Documents  
$ mkdir Kintone_Dojo  
$ cd Kintone_Dojo  
$ git init  
Initialized empty Git  
repository in  
/Users/g001494/Documents/Kin  
töne_Dojo/.git/
```

- Documentディレクトリに移動し、**kintone_dojo**という名前のディレクトリを作成します
 - \$ pwd ~/Documents/Kintone_Dojo
- Git initコマンドでgitリポジトリを初期化します。
- リポジトリは、Repoと短縮されて呼ばれることもあります

README.md ファイルの追加

```
$ touch README.md  
$ vi README.md
```

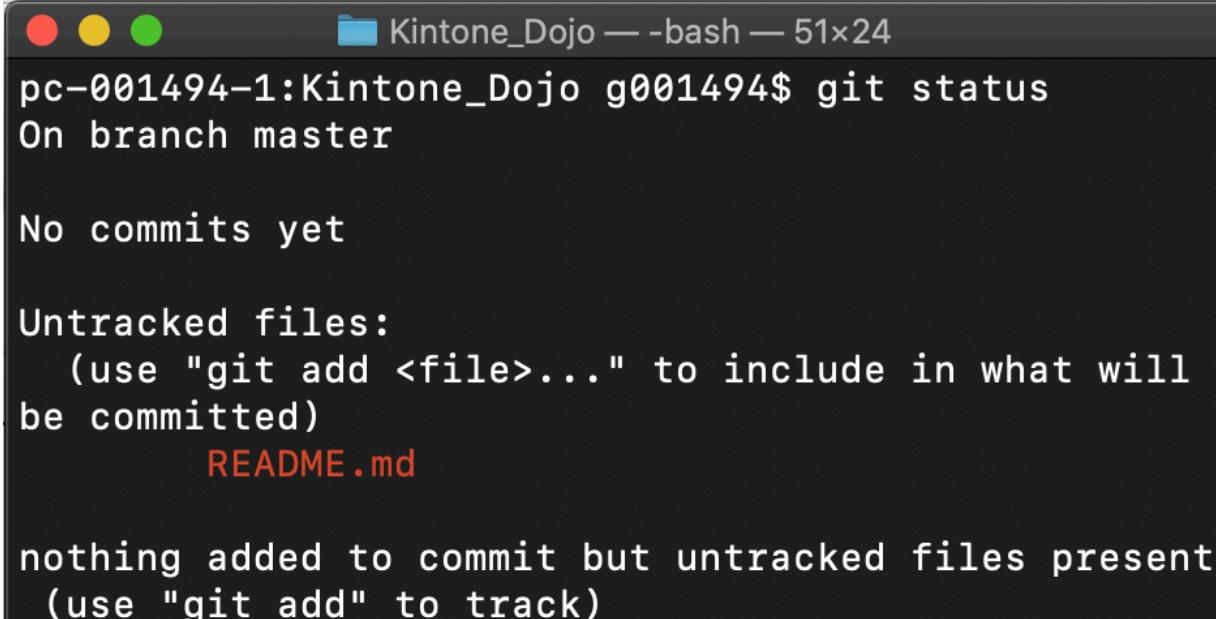
...

```
# Kintone Dojo Repo  
このフォルダはKintone Dojoの課題  
で使用します。
```

- README.md ファイルを作成します
- README.md ファイルにリポジトリの説明を追加します。
- README ファイルは、ソフトウェアや git リポジトリの目的や使用方法を説明するために使用されます。

Gitのステータスを確認する

```
$ git status
```



```
pc-001494-1:Kintone_Dojo g001494$ git status
On branch master

No commits yet

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

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

git status コマンド

- ・作業ディレクトリとステージングエリアを表示します
- ・**Changes to be committed:** どのファイルに変更が加えられているのか確認できます
- ・**Untracked files:** どのファイルはgitで追跡していないのかを確認できます

画像を確認すると、README.mdを追跡する必要があることがわかります。

ステージングエリアにファイルを追加

```
$ git add README.md  
$ git status  
On branch master  
No commits yet  
Changes to be committed:  
  (use "git rm --cached  
<file>..." to unstage)  
    new file: README.md
```

git add <file/folder>

- ファイル/フォルダーをステージングエリアに追加するコマンド

現在README.mdはステージングエリアに存在します

Gitリポジトリにファイルを追加

```
$ git commit -m "README file  
created"
```

```
$ git status
```

```
[pc-001494-1:Kintone_Dojo g001494$ git commit -m ]  
"First version of README file"  
[master (root-commit) d775d42] First version of  
README file  
 1 file changed, 7 insertions(+)  
 create mode 100644 README.md  
[pc-001494-1:Kintone_Dojo g001494$ git status ]  
On branch master  
nothing to commit, working tree clean
```

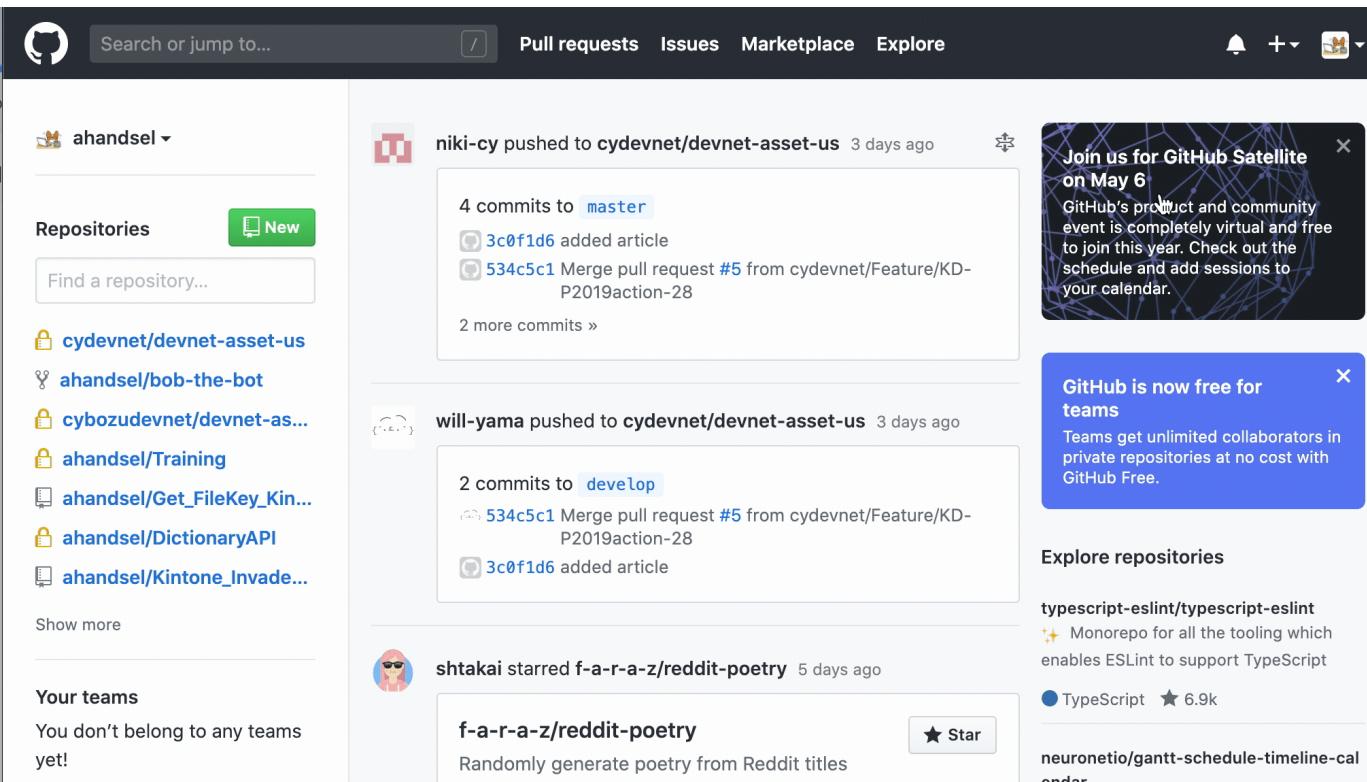
git commit -m "message"

- ファイル/フォルダをリポジトリに追加するコマンド

README.mdがリポジトリに追加されました!!

- **README.mdがmaster**ブランチに追加されたことが分かります

GitHub リポジトリを作成する



リポジトリを作成する

- <https://github.com/new>
- **kintone_dojo**という名前のリポジトリを作成します
- READMEでリポジトリを初期化しないでください

Local Git → GitHub

```
$ git remote add origin  
https://github.com/ahandsel/kintone_dojo.git  
  
$ git push -u origin master  
Username for 'github.com': ahandsel  
Password for 'ahandsel@github.com': <Token_Here>  
Counting objects: 100% (3/3), done.  
Delta compression using up to 4 threads  
Compressing objects: 100% (2/2), done.  
Writing objects: 100% (3/3), 434 bytes | 86.00 KiB/s, done.  
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0  
To https://github.com/ahandsel/kintone_dojo.git  
 * [new branch]      master -> master  
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

Local Git → GitHub

repositoryをpushしましょう！

- GitHubの[Clone or download]ボタンをクリックし、HTTPSリンクをコピーしてURLを取得します

git remote add origin <link>

- ローカルリポジトリをGitHubのリモートリポジトリに接続します

git remote

- リモートリポジトリを管理するコマンドです

<https://help.github.com/en/github/using-git/adding-a-remote>
<https://git-scm.com/docs/git-remote>
<https://git-scm.com/docs/git-push>



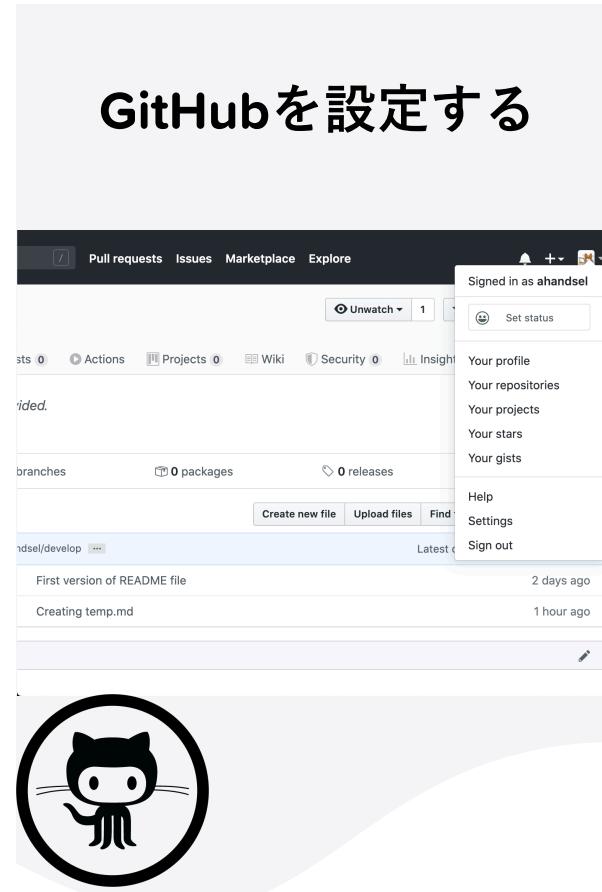
Hands-on Aが完了しました!

gitをインストールする

```
$ git config  
  --global  
  user.name  
  user.email  
  --list
```



GitHubを設定する

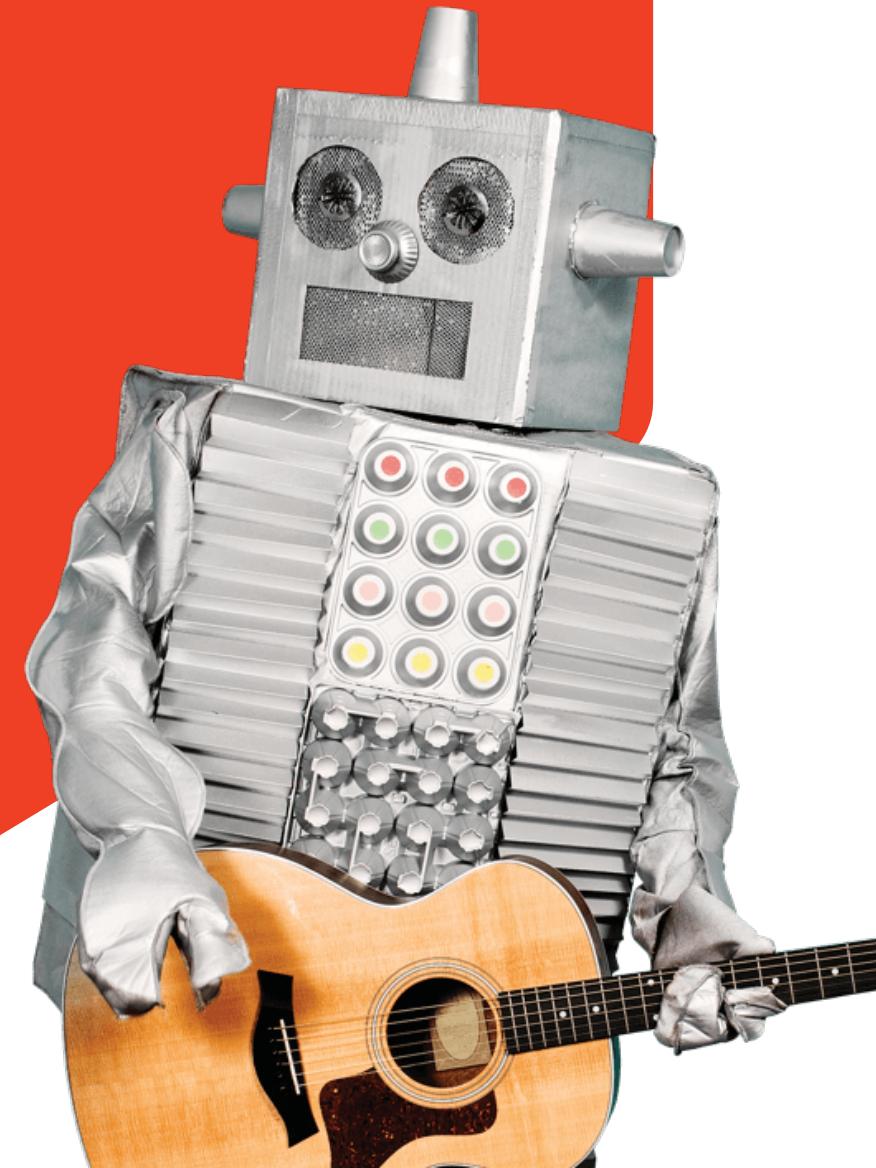


ローカルリポジトリを作成してプッシュする

```
$ git init  
$ git status  
$ git commit -m
```

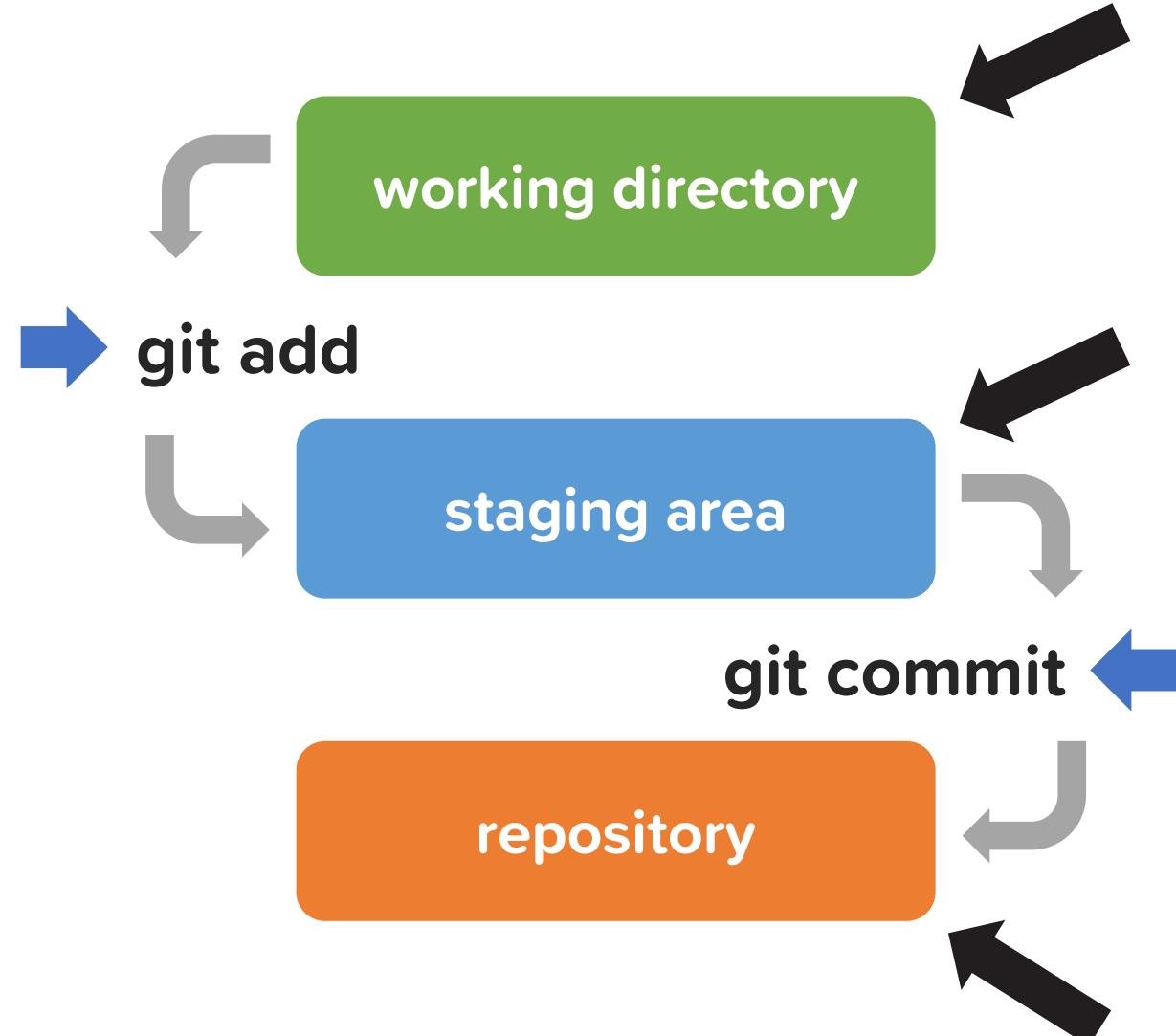
Git, GitHubの解説

ハンズオンAの概要



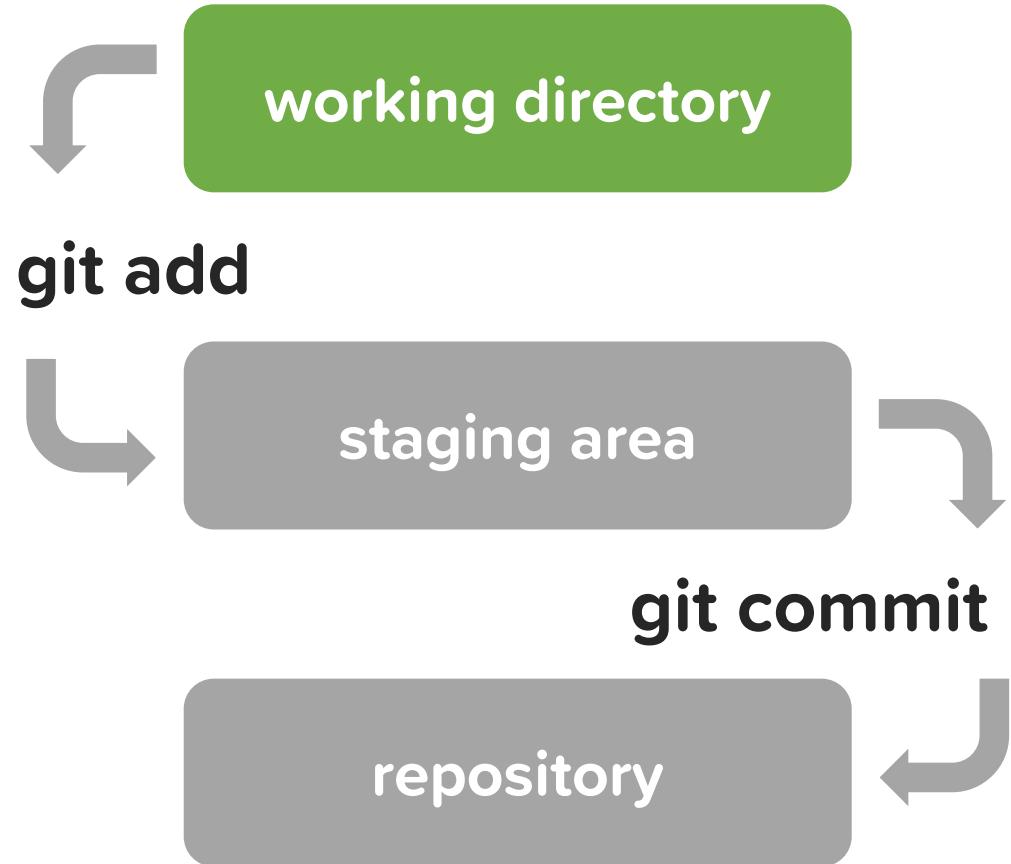
Gitでの変更の保存方法

スペース間でアイテムを転送する2つのコマンドがあります



ファイル、フォルダの変更が保存される3つのスペースがあります

Gitでの変更の保存方法



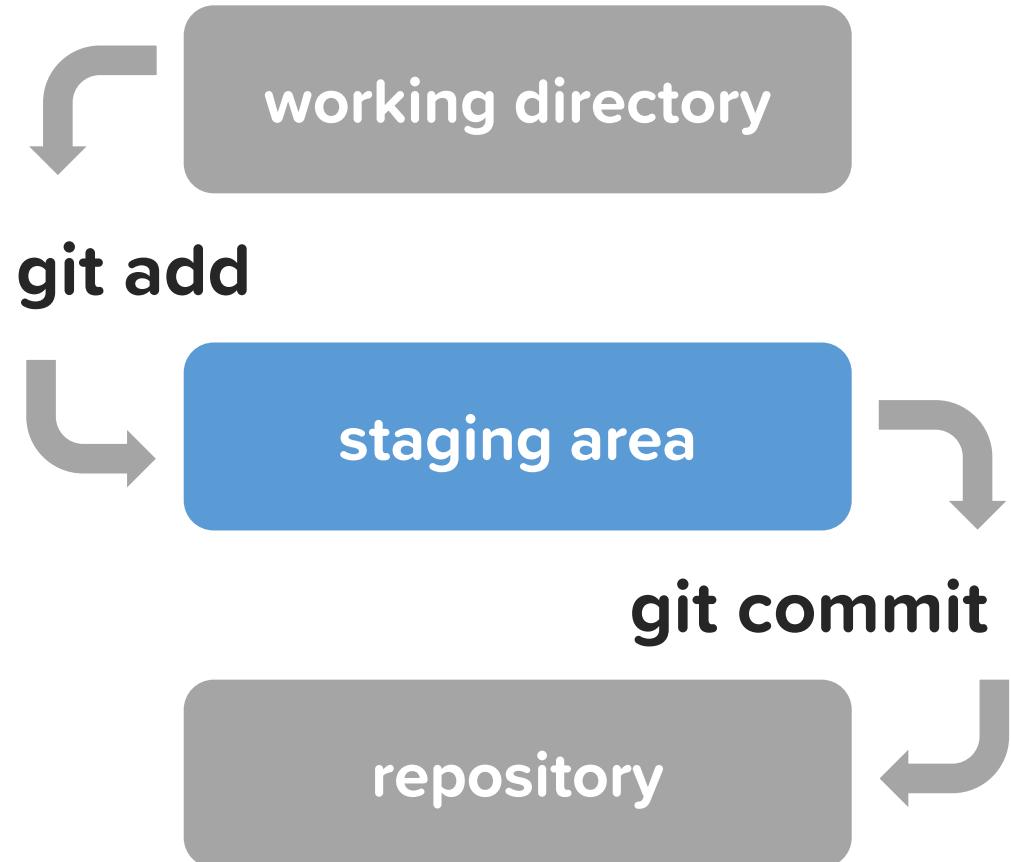
working directory

- ・作業中のローカルディレクトリ
- ・すべての変更は最初に作業ディレクトリーで行われます。

git add

- ・作業ディレクトリからステージングエリアへ追加するコマンド

Gitでの変更の保存方法



staging area

- ・作業ディレクトリとリポジトリの間のバッファ用スペース
- ・以前は「インデックス」と呼ばれていました。
- ・ある特定の変更のみを追加し、まとめてリポジトリに追加する準備を行うことができます

ステージングエリアが必要な理由

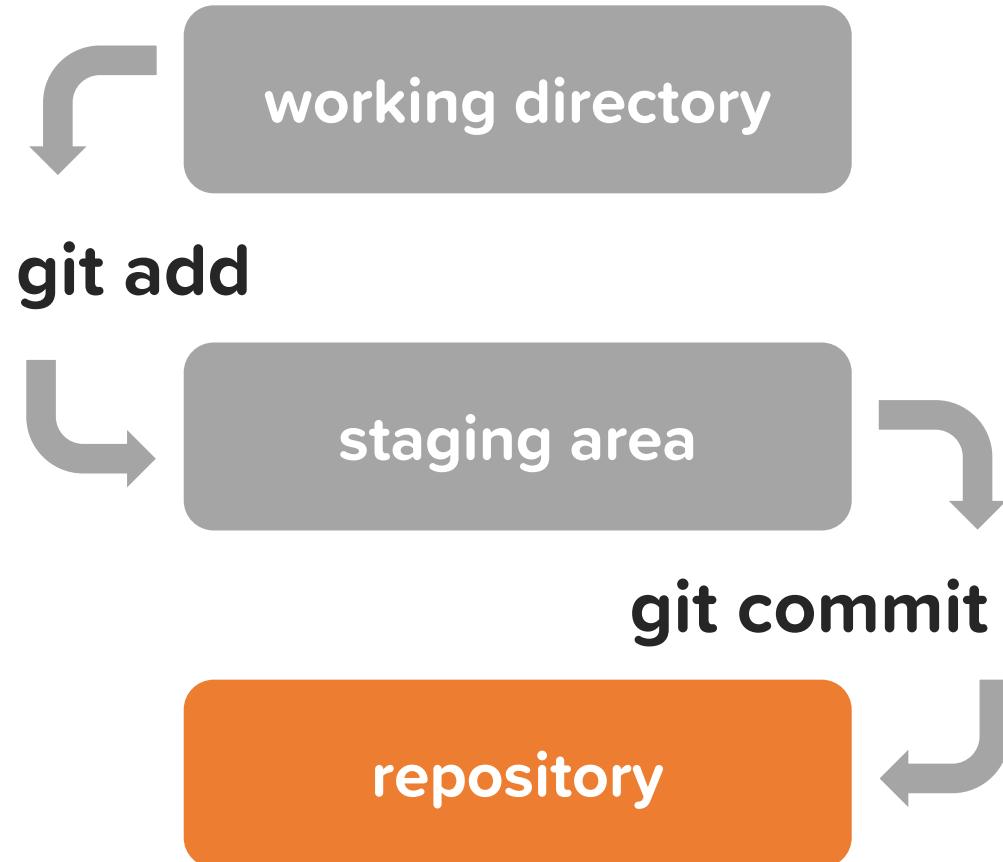
- ファイルをステージングする=コミットのためのファイルを準備する
- あなたが音楽を作っていると想像してください
- あなたは様々なメッセージを含められた曲を書いています。
 - 怒りの歌から愛の歌まですべて。
- すべての音楽をランダムにアップロードしますか？
- いいえ、テーマを付けたアルバムを作りましょう。

ステージングエリアが必要な理由

口マンチックなアルバムを作成するには：

- ラブソングだけを `git add` します。
- すべてのラブソングを追加するまでの間決められたラブソングは `Staging Area` で保存されています。
- アルバム内の必要なすべての曲を `Staging Area` に保存したら、コミットする時間です。
- `git commit -m` する時、"Love Song" ってアルバムのタイトルをコメント追加してコミットします。

Gitで変更を保存する



git commit

- リポジトリへ変更を記録するコマンド
- 変更を保存したいファイルがステージングエリアに配置されたら、**git commit** コマンドを使用します
- ボスを倒してゲームの進行状況を保存したいときに使用するイメージです
- コミットごとにどんな変更を記録するのかを把握するためのコメントを残します。



Gitで変更を保存する

```
g001494@pc-001494-1 Kintone_Dojo % pwd  
/Users/g001494/Documents/Kintone_Dojo  
g001494@pc-001494-1 Kintone_Dojo % ls -la  
total 8  
drwxr-xr-x  4 g001494  staff  128 Jun  3 13:29 .  
drwx-----@ 18 g001494  staff  576 Jun  3 13:25 ..  
drwxr-xr-x 10 g001494  staff  320 Jun  3 13:51 .git  
-rw-r--r--  1 g001494  staff   24 Jun  3 13:34 README.md  
g001494@pc-001494-1 Kintone_Dojo % cd .git  
g001494@pc-001494-1 .git % ls  
HEAD          description      index      objects  
config         hooks           info       refs  
g001494@pc-001494-1 .git %
```

repository

- Gitリポジトリは、プロジェクト内の「.git」フォルダで管理されています。
- リポジトリは、プロジェクトの変更を追跡できます。

remote repository (GitHub)

- GitHubのサーバー上のリポジトリであり、コードを他のユーザーが確認できるようにします

リモートリポジトリの操作

git remote add origin <link>

- ローカルマシンにリモートリポジトリのクローンを作成すると、gitによって**alias**が作成されます。
 - originはリモートリポジトリのURLのニックネームのようなものです
 - 最も一般的な**alias**は「origin」と呼ばれます。
-
- 次のコマンドはどちらも同じ内容を実行します
 - git push -u https://github.com/ahandsel/demo.git master
 - git remote add ALIAS https://github.com/ahandsel/demo.git
 - git push -u ALIAS master

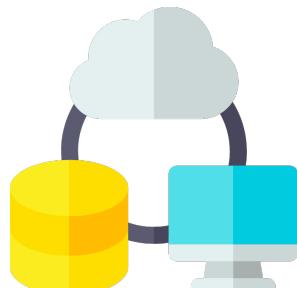
<https://git-scm.com/book/en/v2/Git-Basics-Working-with-Remotes>
<https://git-scm.com/docs/git-remote>

リモートリポジトリの操作

```
$ git remote --verbose
origin https://github.com/user/repo (fetch)
origin https://github.com /user/repo (push)
```

git remote

- ローカルとリモートのリポジトリ間の接続を管理します



git remote --verbose

- Gitが保存しているURLと、そのリモートリポジトリへの読み書き時に使用できるエイリアス（ニックネーム）を一覧表示します。

git push?

git push <remote> <branch>

- ローカルリポジトリをリモートリポジトリにアップロードするコマンドです
- ローカルリポジトリからリモートリポジトリにコミットをエクスポートします
- git fetchの対応（ローカルリポジトリへのインポートコミット）

注：プッシュすると、変更が上書きされる可能性があります。



git push?

git push コマンドは、最後のプッシュまたはクローン以降に変更されていないリモートリポジトリに対してのみ機能します

- つまり、リモートリポジトリを変更した人は他にいません
- あなたと別のクローンが同時にあなたの前に上流にプッシュした場合、そのプッシュは正しく拒否されます。
- 最初に彼らの仕事をフェッチしてプッシュに含める必要があります



git push?

ゲームの例-どうぶつの森

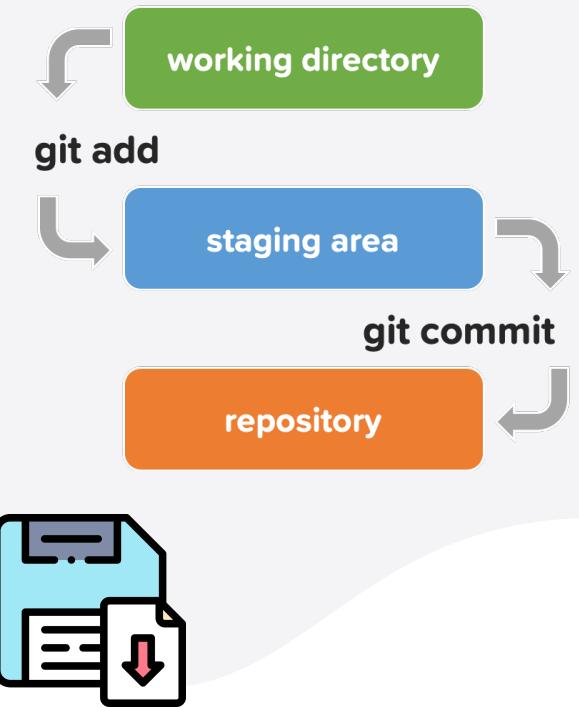
- Genji は島にテントを追加しました。
- GenjiがゲームステータスをGitHubにアップロードしました。
- 宗平も島に家を追加しました。
- しかし、GitHubにアップロードする前に、
 - GitHubから最新のゲームステータスを取得する必要があります (Genjiがテントを追加したため)





ハンズオン A レビュー

Gitでの保存



git remote

ローカルとリモートのリポジトリ間の接続を管理するコマンド

git push

ローカルリポジトリをリモートリポジトリにアップロードするコマンド。