# Git/GitHub 演習(個人演習)

産業技術大学院大学 中鉢欣秀

2016年度

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認 プランチの使い方

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

(参考)より進んだ 使い方

へ~ファイルの削除と名前 の変更

- へ〜 操作を取り消すコマンド 作業の一度中断と再開 その他知っておくと良いコ

### 目次

- Git 入門
  - Git の操作方法と初期設定
  - Git のリポジトリ

  - ブランチの使い方
  - コンフリクト
- - GitHub とは
  - リモートリポジトリ

  - GitHub flow
  - コンフリクトについて

Git/GitHub 演習 (個人演習)

産業技術大学院大 中鉢欣秀

GitHub 入門

(参考)より准んだ 使い方

### Git チートシート(カンニング表)

- 主な Git コマンドの一覧表
  - Git チートシート(日本語版)
- 必要に応じて印刷しておくとよい

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法

変更履歴の作成 履歴の確認 プランチの使い方

コンフリクト

GitHub 入門

SitHub とは リモートリポジトリ SitHub flow

(参考)より進んだ

使い方 ファイルの削除と名

操作を取り消する

その他知っておくと マンド

### Git コマンドの実行確認

- 端末を操作して Git コマンドを起動してみよう。
- 次のとおり操作することで Git のバー ジョン番号が確認できる。

git --version

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ

コミットの作成方法 変更履歴の作成 履歴の確認 ブランチの使い方

「ランチの使い方 Iンフリクト

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

(参考)より進んだ

スマップ ファイルの削除と: の変更

操作を取り消すコ作業の一度中断と

)他知っておくと良 バ

### 名前とメールアドレスの登録

- ●(まだなら)git に名前とメールアドレ スを登録しておく
- 次のコマンドの \$NAME と \$EMAIL を 各自の名前とメールアドレスに置き換 えて実行せよ
  - 名前はローマ字で設定すること

```
git config --global user.name

$NAME
git config --global user.email
$EMAIL
```

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認 プランチの使い方

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

(参考) より進んだ 使い方

|更い/月 |ファイルの削除と |の変更

の変更 桑作を取り消すコマ 作業の一度中断と評

の一段中断と特無 也知っておくと良 ド

### その他の設定

• 次のとおり、設定を行っておく

```
git config --global color.ui
auto
git config --global
push.default simple
git config --global core.editor
emacs
```

- 1 行目: 色付きで表示を見やすく
- 2 行目: push する方法(詳細省略)
- 3 行目: vim を使う場合は emacs ではなく vim を

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow コンフリクトについて

(参考)より進んだ 使い方

マイルの削除と名前の変更

作を取り消すコマンド 業の一度中断と再開 の他知っておくと良いコ

### 設定の確認方法

• ここまでの設定を確認する

```
git config -l
```



産業技術大学院大学 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ

Git のリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認

コンフリクト GitHub 入門

SitHub とは リモートリポジトリ SitHub flow

(参考)より進んだ 使い方

スマイルの削除と D変更

マスス 操作を取り消すコマンド 作業の一度中断と再開 その他知っておくと良いコ

### プロジェクト用のディレクトリ

- リポジトリとはプロジェクトでソース コードなどを配置するディレクトリ
  - このディレクトリを「ワーキング ディレクトリ(ないしはワーキン グツリー)」とも言う
- Git によりバージョン管理ができる
  - ファイルに対する編集作業の内容 が追跡され、記録される
- 将来的に GitHub と連携させることで共 同作業ができるようになる

#### Git/GitHub 演習 (個人演習)

産業技術大学院大学 学 中鉢欣秀

Git 入門

Git の操作方法と初期 Git のリポジトリ

コミットの作成方法 変更履歴の作成 履歴の確認 ブランチの使い方

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

(参考)より進んだ 使い方

こいソJ ファイルの削除と: D変更

の変更 条作を取り消すコマンド 作業の一度中断と再開 その他知っておくと良い

### Git リポジトリを利用するには

- リポジトリを利用する方法には主に2 種類ある
  - git init コマンドで初期化する方法
  - git clone コマンドで GitHub から入 手する方法
- ここでは,まず 1.について解説する

#### Git/GitHub 演習 (個人演習)

産業技術大学院大学 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ

コミットの作成方法 変更履歴の作成 夏歴の確認 ブランチの使い方

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

(参考)より進んだ

ファイルの削除と の変更

の変更 操作を取り消すコマンド 作業の一度中断と再開 その他知っておくと良い

# Git リポジトリの初期化方法

- my\_project ディレクトリを作成し、Git リポジトリとして初期化するコマンド は次のとおり
  - 1 ~ 2 行目:ディレクトリを作成 して移動
  - 3行目:ディレクトリをリポジト リとして初期化
- mkdir ~/my\_project cd ~/my\_project git init
  - 以降の作業は作成した my\_project ディレクトリで行うこと

(個人演習) 産業技術大学院大 学 中鉢欣秀

Git/GitHub 演習

Git 入門 Git の操作方法と初期設定

Gitの操作方法と初期設定 Gitのリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認 ブランチの使い方

GitHub 入門
GitHub とは
リモートリポジトリ
GitHub flow
コンフリクトについて

(参考)より進んだ 使い方 ファイルの削除と名前 の変更 操作を取り消すコマンド

0/77

### リポジトリの状態を確認する方法

現在のリポジトリの状態を確認するコマンドは次のとおり

git status

- このコマンドは頻繁に使用する
- 何かうまく行かないことがあったら、 このコマンドで状態を確認する癖をつ けるとよい
  - 表示される内容の意味は徐々に覚えていけば良い

### Git/GitHub 演習 (個人演習)

産業技術大学院大学 学 中鉢欣秀

Git 入門 Git の操作方法と初期設定

Git のリポジトリ コミットの作成方法 変更履歴の作成 覆歴の確認 ブランチの使い方

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow コンフリクトについて

(参考)より進んだ使い方

ファイルの削除と名前 の変更 操作を取り消すコマンド 作業の一度中断と再開

### 「.git」ディレクトリを壊すべからず

- ティレクトリにリポジトリを作成する と「.git」という隠しディレクトリが できる
  - ls -a で確認できるが…
- このディレクトリは <u>絶対に手動で変更</u> してはならない
  - もし削除したら Git とは無関係の 単なるディレクトリになる

#### Git/GitHub 演習 (個人演習)

産業技術大学院大学 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認

ブランチの使い方 コンフリクト - \_\_

GitHub 入門 GitHub とは リモートリポジトリ

コンフリクトに

(参考)より進んだ 使い方

へ v /J ファイルの削除と の変更

D変更 操作を取り消すコマント 作業の一度中断と再開

### コミットについて

- Git の用語における「コミット」とは、 「ひとかたまりの作業」をいう
  - 新しい機能を追加した,バグを直 した,ドキュメントの内容を更新 した,など
- Git は作業の履歴を、コミットを単位と して管理する
  - コミットは次々にリポジトリに追加されていき、これらを記録することでバーションの管理ができる(古いバージョンに戻る、過去の変更内容を確認する、など)

■ コミットには、作業の内容を説明する

Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門 Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法

プランチの使い方 コンフリクト

GitHub 入門
GitHub とは
リモートリポジトリ
GitHub flow
コンフリクトについて
(参考)より進んだ

(学場) まり 進んだ 使い方 ファイルの削除と名前 の変更 操作を取り消すコマンド

の一度中断と再開 の他知っておくと良いコ バド

13/77

### README ファイルの作成

my\_project リポジトリに README ファイルを作成してみよう

echo "My README file." > README

プロジェクトには <u>必ず README ファ</u> イルを用意 しておくこと

### Git/GitHub 演習 (個人演習) 産業技術大学院大

学中鉢欣秀

Git 入門 Git の操作方法と初期設定

Git のリポジトリ コミットの作成方法 変更履歴の作成

プランチの使い方 コンフリクト

GitHub 入門

Jモートリポジトリ SitHub flow コンフリクトについて

(参考)より進んだ

~ ・ / ・ ファイルの削除と名 の変更

の変更 操作を取り消すコマン 作業の一度中断と再開

> 他知っておくと良い ド

### リポジトリの状態の確認

● git status で現在のリポジトリの状態を 確認する

git status

未追跡のファイル(Untracked files:)の 欄に作成した README ファイルが (赤色で)表示される

#### Git/GitHub 演習 (個人演習)

産業技術大学院大学 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定

Git のリポジトリ コミットの作成方法

履歴の確認 ブランチの使い方 コンフリクト

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

コンフリクト

(参考) より進んだ 使い方

-ファイルの削除と4 D**変更** 

の変更 操作を取り消すコマン 作業の一度中断と再開

### 変更内容のステージング

- コミットの一つ手前にステージングと いう段階がある
  - コミットしたい変更はステージン グしておく
    - 逆に言えば、変更をコミットするためには、 ステージングしておかなくてはならない
  - 新しいファイルをステージングす ると、これ以降、git がそのファイ ルの変更を追跡するようになる
    - これをトラッキングという

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 中鉢欣秀

Git 入門

コミットの作成方法

GitHub 入門

(参考)より准んだ

使い方

### ステージングの実行

作成した README ファイルをステージングするには、次のコマンドを打つ

git add .

- 「git add」の「.(ピリオド)」を忘れな いように
  - ピリオドは、リポジトリにおける すべての変更を意味する。
  - 複数のファイルを変更した場合には、ファイル名を指定して部分的にステージングすることもできる…
    - が、このやりかたは好ましくない
    - 一度に複数の変更を行うのではなく,一つ

Git/GitHub 演習 (個人演習)

産業技術大学院大学 学 中鉢欣秀

Git 入門

Git 八十 Git の操作方法と初期設定

Gitのリボジトリコミットの作成方法 変更履歴の作成 履歴の確認

ョンフリクト GitHub 入門

GitHub 人口 GitHub とは リモートリポジトリ GitHub flow コンフリクトについて

(参考) より進んだ 使い方 ファイルの削除と名前 の変更

操作を取り消すコマンド 作業の一度中断と再開 その他知っておくと良いコ

# ステージング後のリポジトリへの状態

再度, git status コマンドで状態を確認 しよう

```
git status
```

● コミットされる変更(Changes to be committed:)の欄に、README ファイルが(緑色で)表示されれば正しい結果である



産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法

変更機能の作成 関歴の確認 プランチの使い方

コンフリクト GitHub 入門

GitHub 人門
GitHub とは
リモートリポジトリ
GitHub flow
コンフリクトについて

(参考)より進んだ使い方ファイルの削除と名前の変更

操作を取り消すコマンド 作業の一度中断と再開 その他知っておくと良いコ マンド

# ステージングされた内容をコミット する

- ステージング段階にある変更内容をコ ミットする
- コミットにはその内容を示すメッセー ジ文をつける
- ●「First commit」というメッセージをつけ て新しいコミットを作成する

commit -m 'First commit'

●「-m」オプションはそれに続く文字 列をメッセージとして付与するこ とを指示するもの

(参考)より進んだ 使い方

Git/GitHub 演習 (個人演習) 産業技術大学院大

中鉢欣秀

Git 入門 コミットの作成方法

GitHub 入門

### コミット後の状態の確認

- コミットが正常に行われたことを確 認する
  - ここでも git status コマンドか活 躍する

### status

- 「nothing to commit, . . . 」との表示から コミットすべきものがない(=過去の 変更はコミットされた)ことがわかる
- この表示がでたら(無事コミットでき たので)一安心してよい

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 中鉢欣秀

Git 入門

コミットの作成方法

GitHub 入門

(参考)より准んだ

使い方

### 更なるコミットを作成する

- リポジトリで変更作業を行い、新しい コミットを追加する
  - README ファイルに新しい行を追 加する
- 次の \$NAME をあなたの名前に変更して実行しなさい

echo \$NAME >> README

既存のファイルへの追加なので「>>」を用いていることに注意

Git/GitHub 演習 (個人油習) 産業技術大学院大 中鉢欣秀 Git 入門 GitHub 入門 (参考)より進んだ 使い方

### 変更後の状態の確認

• リポジトリの状態をここでも確認する

git status

コミットのためにステージされていない変更(Changes not staged for commit:)の欄に、変更された(modified)ファイルとして README が表示される

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

it の操作方法と初期設定 it のリポジトリ

更履歴の作成 歴の確認

コンフリクト

GitHub 入門

リモートリポジトリ GitHub flow

(参考)より進んだ 使い方

-7ァイルの削除と: )変更

変更 対作を取り消すコマンド ○業の一度中断と再開 ○の他知っておくと良いこ

### 差分の確認

▶ トラックされているファイルの変更箇 所を確認する

```
git diff
```

- 頭に「+」のある(緑色で表示された) 行が新たに追加された内容を示す
  - 削除した場合は「-」がつく

#### Git/GitHub 演習 (個人演習)

産業技術大学院大学 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ

**変更履歴の作成** 履歴の確認

コンフリクト Nations 7 日日

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

(参考)より進んだ 使い方

> 'ァイルの削除と名 |変更

の変更 操作を取り消すコ<sup>-</sup>

業の一度中断と再開 の他知っておくと良 ンパ

### 新たな差分をステージングする

作成した差分をコミットできるように するために、ステージング段階に 上げる

git add .

● git status を行い,README ファイルが「Changed to be commited:」の欄に(緑色で)表示されていることを確認する

#### Git/GitHub 演習 (個人演習)

産業技術大学院大学 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ

変更履歴の作 履歴の確認

プランチの使い コンフリクト

GitHub 入門

AITHUD 人門 GitHub とは

リモートリポジトリ GitHub flow コンフリクトについ

(参考) より進んだ 使い方

ファイルの削除と の変更

○変更 ೬作を取り消すコマンド ○業の一度中断と再開

## ステージングされた新しい差分のコ ミット

● 変更内容を示すメッセージとともにコミットする

```
git commit -m 'Add my name'
```

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git ハーコ Git の操作方法と初期設定 Git のリポジトリ

変更履歴の作成履歴の確認

コンフリクト GitHub 入門

tHub 入門 tHub とは

Hub とは Eートリポジトリ Hub flow シフリクトについて

(参考)より進んだ使い方ファイルの削除と名前の変更

変更 4作を取り消すコマンド 業の一度中断と再開

### バージョン履歴の確認

- これまでの変更作業の履歴を確認
  - 2つのコミットが存在する

```
git log
```

- 各コミットごとに表示される内容
  - コミットの ID (commit に続く英文 字と数字の列)
  - Author と Date
  - コミットメッセージ

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

#### Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成

#### 覆歴の確認

ブランチの使い コンフリクト

### GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

#### (参考)より進んだ 使い方

ステイルの削除と名前 D変更

を作を取り消すコマンド 業の一度中断と再開 ∵の他知っておくと良い

### 2つのコミットの比較

- 異なる2つのコミットの変更差分は次のコマンドで確認できる
  - コミットの ID は log で確認できる (コマンドで ID を指定する場合は, 概ね先頭 4 文字を入力し後は省略 してよい)

```
git diff $COMMIT_ID_1

→ $COMMIT_ID_2
```

●(参考)後で説明するブランチはコミットのエイリアスなので、ブランチごとの比較もできる

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門
Git の操作方法と初期設定
Git のリポジトリ
コミットの作成方法
変更履歴の作成
履歴の確認

コップリット GitHub 入門

GitHub とは リモートリポジトリ GitHub flow コンフリクトについて

(参考) より進んだ 使い方 ファイルの削除と名前 の変更

)変更 上作を取り消すコマンド 三業の一度中断と再開 ・の他知っておくと良いコ

### コミットの情報確認

● 次のコマンドでコミットで行った変更 内容が確認できる

git show \$COMMIT\_ID

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法

履歴の確認

コンフリクト

GitHub 入門

altHub 人門 GitHubとは

リモートリポジトリ GitHub flow コンフリクトについて

(参考)より進んだ 使い方

ファイルの削除と の変更

の変更 操作を取り消すコ 作業の一度中転と

> その他知っておくと マンド

### ブランチとは

- 「ひとまとまりの作業」を行う場所
- ソースコードなどの編集作業を始める際には必ず新しいブランチを作成する
- Git の内部的にはあるコミットに対するエイリアス(alias)である

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

#### Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成

#### **ブランチの使い方** コンフリクト

GitHub 入門

リモートリポジトリ GitHub flow

#### (参考)より進んだ 使い方

-ファイルの削除と4 の変更

操作を取り消すコマ 作業の一度中断と再

の他知っておくと

### master は大事なブランチ

- Git リポジトリの初期化後,最初のコミットを行うと master ブランチができる
- 非常に重要なブランチであり、ここで 直接編集作業を行ってはならない
  - ただし、本演習や、個人で Git を利用する場合はこの限りではない

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成

**ブランチの使い方** コンフリクト

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

(参考)より進んだ 使い方

ファイルの削除と: の変更

操作を取り消すコマンド 作業の一度中断と再開 その他知っておくと良い

### ブランチの作成と移動

- 新しいブランチ「new\_branch」を作成して、なおかつ、そのブランチに移動する
  - ●「-b」オプションで新規作成
  - オプションがなければ単なる移動 (後述)

git checkout -b new\_branch

◆ 本来,ブランチには 「これから行う作業の内容」が分かる名前を付ける

Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成

**ブランチの使い方** コンフリクト

GitHub 入門 GitHub とは リモートリポジトリ

リモートリポジトリ GitHub flow コンフリクトについ

(参考)より進んだ 使い方

欠 ♥ \* /J ファイルの削除と の変更 \*& /r \* ™ ! / \*\* \* \* \* \*

・へへ 操作を取り消すコマン :業の一度中断と再開

### ブランチの確認

- ブランチの一覧と現在のブランチを確認する
  - もともとある master と、新しく作 成した new\_branch が表示される

git branch -vv

- ブランチに紐づくコミットの ID が同じ ことも確認
- git status の一行目にも現在のブランチ が表示される

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成

プランチの使い方

Call. L 7 88

GitHub 入門 GitHub とは リモートリポジトリ

GitHub flow コンフリクトに

(参考)より進んだ 使い方

ファイルの削除と の変更 場件を取り当まっ

作を取り消すコマンド 業の一度中断と再開 の他知っておくと良い:

### ブランチでのコミット作成

README に現在の日時を追加

```
date >> README
2 git add
3 | qit commit -m 'Add date'
```

- 新しいコミットが追加できたことを git log で確認
- git branch -vv でコミットの ID が変化し たことも確認

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 中鉢欣秀

### Git 入門

ブランチの使い方

GitHub 入門

(参考)より進んだ

### 使い方

### ブランチの移動

- new\_branch ブランチでコミットした内容を master に反映させる
  - まずは master に移動する

git checkout master

- git status, git branch -vv で現在のブラ ンチを確認すること
- この段階では、README ファイルに行った変更が <u>反映されてない</u> ことを確認すること

#### Git/GitHub 演習 (個人演習)

産業技術大学院大学 学 中鉢欣秀

#### Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認

### プランチの使い方

コンフリクト

### GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

#### (参考) より進んだ 使い方

ァイルの削除と名前 変更 作を取り消すコマンド 季の一度中断と再開

34/77

### 変更を master にマージ

● new\_branch で行ったコミットを master に反映させる

git merge new\_branch

- README に更新が反映されたことを確認
- git branch -vv により両ブランチのコミット ID が同じになったことも確認
- git log も確認しておきたい

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成

ブランチの使い方

Call. E 7 88

GitHub 入門 GitHub とは

リモートリポジトリ GitHub flow

コンフリクトに

(参考) より進んだ 使い方

ファイルの削除と の変更

作を取り消すコマンド 業の一度中断と再開 の他知っておくと良い

### マージ済みブランチの削除

マージしたブランチはもはや不要なの で削除して良い

git branch -d new\_branch

● git branch -vv コマンドで削除を確認

#### Git/GitHub 演習 (個人演習)

産業技術大学院大 中鉢欣秀

Git 入門

ブランチの使い方

GitHub 入門

(参考)より准んだ 使い方

## コンフリクトとは

- ファイルの同じ箇所を,異なる内容に 編集すると発生する
- Git はどちらの内容が正しいのかわからない
- 次のシナリオに従い、コンフリクトを 発生させてみよう

### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

### Git 入門

Gitの操作方法と初期設定 Gitのリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認 ブランチの使い方 コンプリクト

## GitHub 入門

AitHub 人門 GitHub とは リモートリポジトリ GitHub flow

## (参考)より進んだ 使い方

ファイルの削除と名前 の変更 操作を取り消すコマン

#TFを取り消りコマント 作業の一度中断と再開 その他知っておくと良い

## コンフリクトのシナリオ

- ●「のび太」の作業
  - nobita ブランチを作成する
  - README ファイルの <u>一行目</u> を 「Nobita's README.」に変更する
  - 変更を add して commit する
- ここで一度, master ブランチにもどる
  - README がもとのままだあること を確認
- ●「しずか」の作業
  - shizuka ブランチを作成する

▲ 変更を add I. て commit する

● README ファイルの <u>一行目</u> を 「Shizuka's file.」に変更する Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門
Git の操作方法と初期設定
Git の場件方法と初期設定
Git のリポジトリ
コミットの作成方法
変更履歴の作成
履歴の確認
プランチの使い方

コンフリクト

GitHub 入門

GitHub とは

リモートリポジトリ

GitHub flow

(参考)より進んだ使い方ファイルの削除と名前の変更 操作を取り消すコマンド

38/77

# マージとコンフリクト発生

- master ブランチに移動する
- まず, nobita ブランチをマージ
  - 問題なくマージできる
- 次に, shizuka ブランチをマージ
  - ここでコンフリクトが発生する

## Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

### Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 腹歴の確認 プンチの使い方 コンフリクト

## GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

### (参考)より進んだ 使い方

ファイルの削除と名前 の変更 操作を取り消すコマンド 作業の一度中断と再開

# コンフリクト時のメッセージ

merge に失敗するとようなメッセージ が出る(長いので改行を加えた)

Auto-merging README
CONFLICT (content):
Merge conflict in README
Automatic merge failed; fix
conflicts
and then commit the result.

また、git status すると Unmerged pathes: の欄に、「both modified: README」と表示される

## Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門
Git の操作方法と初期設定
Git のリポジトリ
コミットの作成方法
変更履歴の作成
履歴の確認
プランチの使い方
コンプリクト

GitHub 入門
GitHub とは
リモートリポジトリ
GitHub flow
コンフリクトについて

(参考) より進んだ 使い方 ファイルの削除と名前 の変更 操作を取り消すコマンド

> )一度中断と再開 3知っておくと良いコ :

# README ファイルの内容

● README を開くとコンフリクトが起き た箇所がわかる

======= の上がマージ前の master ブランチ,下がマージしようとした shizuka ブランチの内容

## Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門
Git の操作方法と初期設定
Git のリポジトリ
コミットの作成方法
変更履歴の作成
履歴の確認
プランチの使い方
コンフリクト

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow コンフリクトについて

(参考)より進んだ使い方ファイルの削除と名前の変更

操作を取り消すコー 作業の一度中断と その他知っておく。

41/77

## コンフリクトの解消

● テキストエディタで修正し、手動でコンフリクトを解消する

Nobita & Shizuka's READMEfile. (以下略)

## Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Sitのリポジトリコミットの作成方法 変更履歴の作成

夏歴の確認 プランチの使い方

コンフリクト

GitHub 入門

GitHub 人門
GitHub とは
リモートリポジトリ
GitHub flow

(参考)より進んだ 使い方

ファイルの削除との変更

操作を取り消すコマ 作業の一度中断と再

# 解消した結果をコミットする

- その後はいつもどおり、add して commit すれば作業が継続できる
  - マージ済みの master から新しくブランチを作成すること
- なお、テキストエディタを用い、手動で正しくコンフリクトを解消する前でもcommitできてしまうので、この点には注意する

### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

### Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認 ブランチの使い方

## コンフリクト

GitHub 入門 GitHub とは リモートリポジトリ GitHub flow

## (参考) より進んだ 使い方

マイルの削除と名 ファイルの削除と名 D変更 B作を取り消すコマ

e作を取り消すコマンド ■業の一度中断と再開 ・の他知っておくと良い

# 目次

- - Git の操作方法と初期設定
  - Git のリポジトリ

  - ブランチの使い方
  - コンフリクト
- GitHub 入門
  - GitHub とは
  - リモートリポジトリ
  - GitHub flow
  - コンフリクトについて

Git/GitHub 演習 (個人演習)

産業技術大学院大 中鉢欣秀

Git 入門

(参考)より准んだ

使い方

44/77

# GitHub でソーシャルコーディング

- ソーシャルコーディングのためのクラウド環境
  - GitHub
  - GitHub Japan
- GitHub が提供する主な機能
  - GitHub flow による協同開発
  - Pull requests
  - Issue / Wiki

### Git/GitHub 演習 (個人演習)

産業技術大学院大学 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成

変更履歴の行成 履歴の確認 ブランチの使い方

コンフリクト ..... 7 BB

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

(参考)より進んだ 使い方

スマップ ファイルの削除と の変更

----操作を取り消すコマント 作業の一度中断と再開 その他知っておくと良い

# GitHub アカウントの作成

- ●(まだなら)次の URL の指示に従い GitHub アカウントを作成
  - Signing up for a new GitHub account
     User Documentation
- アカウントの種類

GitHub Help

- 無料版で作成する場合「Join GitHub for Free」を選択する
- 学生の場合「Student Developer Pack」にアップグレードすること もできる
- その後、確認メールが届くので、必要 に応じて残りの手順を実施せよ

Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門
Git の操作方法と初期設定
Git の場件方法と初期設定
Git のリポジトリ
コミットの作成方法
変更履歴の作成
履歴の確認

GitHub 入門 GitHub とは

リモートリポジトリ GitHub flow コンフリクトについて

(参考)より進んだ 使い方 ファイルの削除と名前 の変要 操作を取り消すコマンド 作業の一度中断と再層

## SSH による GitHub アクセス

- GitHub へのアクセスは SSH を用いた 公開鍵暗号方式の認証を用いる
  - SSH 公開鍵の設定を行えば以降の パスワード認証が不要になる
- (まだなら) SSH を生成して GitHub に 登録しなさい
  - 鍵を生成するとき「passphrases」 が聞かれるが、この演習では何も 入力しなくてよい
  - Generating an SSH key User Documentation

## Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

### Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認

## GitHub 入門

GitHub とは リモートリポジトリ GitHub flow コンフリクトについて

## (参考)より進んだ 使い方

ファイルの削除と: D変更 B.*tr* t 取り※ナコ・

作を取り消すコマンド 業の一度中断と再開 の他知っておくと良い:

# リモート VS ローカルリポジトリ

- ローカルリポジトリ
  - git init コマンドを用いて作成した リポジトリを「ローカルリポジト リ」という
- リモートリポジトリ
  - ●「リモートリポジトリ」とは,サーバ上にあるリポジトリであり,ローカルのリポジトリと連携させることができる
- リモートリポジトリの利点
  - ネットワークを経由してどこから でも利用することができる

■ 複数人のチームで協同作業をする

Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門
Git の操作方法と初期設定
Git の場件方法と初期設定
Git のリボジトリ
コミットの作成方法
変更履歴の作成
履歴の確認
プランチの使い方

GitHub 入門
GitHub とは
リモートリポジトリ
GitHub flow
コンプリクトについて

(参考) より進んだ 使い方 ファイルの削除と名前 の変更 操作を取り消すコマンド

48/77

# リモートリポジトリの作成

- リモートリポジトリを GitHub で作 成する
  - 名前は「our\_project」とする
- 次の手順で作成する
  - Creating a new repository User Documentation
- README とライセンスを追加すること
  - Initialize this repository with a README」にチェックを入れる
  - ●「Add a license:」から「MIT License」を選ぶ

## Git/GitHub 演習 (個人演習)

産業技術大学院大学院大学 学 中鉢欣秀

Git 入門
Git の操作方法と初期設定
Git のリポジトリ
コミットの作成方法
変更履歴の作成

ブランチの使い方 コンフリクト GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

(参考)より進んだ

! い.力 \*ァイルの削除と名 )変更

作を取り消すコマンド 業の一度中断と再開 の他知っておくと良いコ

## GitHub floww による開発の流れ

- GitHub flow
  - Understanding the GitHub Flow GitHub Guides



中鉢欣秀

Git 入門

GitHub 入門

GitHub flow

(参考)より准んだ 使い方

# 1: リモートリポジトリをローカルに 複製

- リモートにあるリポジトリをローカル に複製することを clone という
  - Cloning a repository User Documentation
- 下記の「\$GITHUB\_URL」の部分を GitHub の our\_project リポジトリ URL にして実行
  - URL は「git@...」で始まる SSH
    - 接続用のものを用いる
      - リポジトリの URL はブラウザ用の URL と は異なるので注意!

2|git clone \$GITHUB\_URL

産業技術大学院大学 中本欣秀

Git/GitHub 演習 (個人演習)

の操作方法と初期設定 のリポジトリ ミットの作成方法 更履歴の作成 歴の確認 ランチの使い方 ンフリクト

GitHub 入門
GitHub とは
リモートリポジトリ
GitHub flow
コンフリクトについて

(参考)より進んだ 使い方 ファイルの削除と名前 の変更 操作を取り消すコマンド

# 2: master から作業用ブランチを作成

- 作業用のブランチを作成して移動する ブランチの名前は「greeting」 とする
- git checkout -b greeting

## Git/GitHub 演習 (個人演習) 産業技術大学院大

中鉢欣秀

Git 入門

GitHub 入門

GitHub flow

(参考)より准んだ 使い方

# 3: ブランチで編集作業を行う

● ここでは, hello.txt という名前のファイ ルを作成する

| echo 'Hello GitHub' > hello.txt

Git/GitHub 演習 (個人演習) 産業技術大学院大 中鉢欣秀

Git 入門

GitHub 入門

GitHub flow

(参考)より准んだ

使い方

# 4: ブランチでコミットを作成

● 変更した内容をステージングしてから コミットする

```
git add .
2 git commit -m 'Create
  → hello.txt!
```

- この編集、add、commit の作業は作業 が一区切りつくまで何回も繰り返して よい…
  - が,こまめに push するのが良いと される

## Git/GitHub 演習 (個人油習)

産業技術大学院大 中鉢欣秀

Git 入門

GitHub 入門

(参考)より進んだ 使い方

# 5: ブランチをリモートに送る

- ブランチで作成したコミットをリモー トに送る
  - 下記の origin はリポジトリの URL の別名として自動で設定されているもの
  - greeting は作業しているブランチ名

git push -u origin greeting

## Git/GitHub 演習 (個人演習) 産業技術大学院大 学 中鉢欣秀 Git 入門 Ga の操作方法と初期設定 Ga のリポジトリ

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認 ブランチの使い方 コンフリクト

GitHub 入門 GitHub とは リモートリポジトリ

GitHub flow コンフリクトについて

(参考) より進んだ 使い方 ファイルの削除と名前

ファイルの削除と名前の変更 操作を取り消すコマンド 作業の一度中断と再開

# 6. GitHub でプルリクエストを送る

- ブランチが GitHub に登録されたことを 確認し、Pull request を作成する
- 手順は次のとおり
  - Using pull requests User Documentation の前半
  - Creating a pull request User Documentation

### Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 度歴の確認 プランチの使い方

GitHub 入門

GitHub とは リモートリポジトリ

GitHub flow コンフリクトについて

(参考)より進んだ

こいソリ ファイルの削除と名 D変更

作を取り消すコマンド 業の一度中断と再開 の他知っておくと良い

# 7. GitHub でレビュー(+自動テスト)

- プルリクエストを用いたレビューの方 法は下記参照
  - Using pull requests User Documentation の後半
- 人手によるレビューの他、自動的なテ ストも行うのが望ましい(説明は省略)

## Git/GitHub 演習 (個人演習)

産業技術大学院大 中鉢欣秀

Git 入門

GitHub 入門

GitHub flow

(参考)より准んだ 使い方

# 8. GitHub でプルリクエストをマージ

- Pull request のレビューが済んだらマー ジする
  - Merging a pull request User **Documentation**
- マージが完了したら、ローカル・リモ ート共に、マージ済みのブランチは削 除してよい

### Git/GitHub 演習 (個人演習)

産業技術大学院大 中鉢欣秀

Git 入門

GitHub 入門

GitHub flow

(参考)より准んだ 使い方

# 9. ローカルの master を最新版にする

- GitHub で行ったマージをローカルに反映させる
  - master ブランチに移動して git pull
  - 不要になった作業用ブランチ は削除

```
git checkout master
git pull
git branch -d greeting
```

## Git/GitHub 演習 (個人演習) 産業技術大学院大

果技術人学院人 学 中鉢欣秀

Git 入門 Git の操作方法と初期設 Git のリポジトリ

コミットの作成: 変更履歴の作成 履歴の確認

関歴の確認 ブランチの使い方 コンフリクト

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

コンフリクトについて

(参考)より進んだ 使い方

: V ・ /J ? ァイルの削除と: )変更 &作を取り消ます:

作を取り消すコマン 業の一度中断と再開 の他知っておくと良

# GitHub Flow に習熟するには?

ここで手順 2: (52節) に戻り、一連の作業を複数回(5回以上!)繰り返すこと体に叩き込む!

Git/GitHub 演習 (個人演習) 産業技術大学院大 学 中鉢欣秀

Git 入門
Git の操作方法と初期設定
Git のリポジトリ
コミットの作成方法
変更履歴の作成
履歴の確認

GitHub 入門
GitHub とは
リモートリポジトリ
GitHub flow
コンフリクトについて

(参考)より進んだ使い方ファイルの削除と名前の変更 操作を取り消すコマンド作業の一度中職と再開 その他知っておくと良いコ

# GitHub flow におけるコンフリクトについて

- コンフリクトとは?
  - コンフリクトは、コードの同じ箇 所を複数の人が別々に編集する と発生
- コンフリクトが起きると?
  - GitHub に提出した Pull requests が 自動的にマージできない

## Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門 Git の操作方法と

コミットの作成方 変更履歴の作成 履歴の確認

夏歴の確認 ブランチの使い方 ヨンフリクト

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

コンフリクトについて

(参考)より進んだ 使い方

! い **力** ' ァイルの削除と名前 |変更

の変更 操作を取り消すコマ

業の一度中断と再開 の他知っておくと良

# コンフリクトへの基本的な対処法

- 初心者は、演習の最初の方では「他人 と同じファイルを編集しない」ことに して、操作になれる
  - 上達したら積極的にコンフリクト を起こしてみて、その解決方法 を学ぶ
- コミットはできるだけ細かく作成する と良い
  - その分,他の人とかち合う可能性 が減る

## Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

### Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認 プランチの使い方

## GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

## ヨンフリクトについて

(参考)より進んだ 使い方 ファイルの削除と名前 の変更 操作を取り消すコマンド

口っておくと良い

# GitHub でのコンフリクトの解消方法

- new\_feature ブランチで作業中であり、 最新の更新は commit 済とする
- 解消するための操作は次のとおり
  - 1 行目~ 2 行目:master を最新版にする. 2 行目: リモートの master を shizuka にマージ. 3 行目: コンフリクトを解消する. 4~6 行目: このブランチを再度 push
- これにより、プルリクエストがマージ 可能になれば成功

Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門
Git 入門
Git の操作方法と初期設定
Git のリポジトリ
コミットの作成方法
変更関歴の作成
履歴の確認
プラナの使い方
コンフリクト

コンノリクト

GitHub 入門

GitHub とは
リモートリポジトリ

GitHub flow
コンフリクトについて

(参考)より進んだ使い方

・ 方 イルの削除と名前 更 を取り消すコマンド

git checkout master git pull

# 目次

- Git 入門
  - Git の操作方法と初期設定
  - Git のリポジトリ
  - コミットの作成方法
  - 変更履歴の作成
  - ●履歴の確認
  - ブランチの使い方
  - コンフリクト
- GitHub 入門
  - GitHub とは
  - GILHUD C VA
  - リモートリポジトリ
  - GitHub flow
  - コンフリクトについて

Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法

> 夏歴の確認 ブランチの使い方 コンフリクト

GitHub 入門 GitHub とは

GitHub flow コンフリクトについて (参考)より准んだ

使い方 ファイルの削除と名前

ファイルの削除と名前 の変更 操作を取り消すコマンド

作を取り消すコマンド 業の一度中断と再開 の他知っておくと良い

64/77

## Git に無視させたいファイル

- ツールが生成する中間ファイルなど、 Git で管理させたくないファイルは予め 「.gitignore」ファイルに記述しておく
- なお,「.gitignore」ファイル自体は Git がトラッキングするファイルに含める
- .qitiqnore の書き方については各自で 調べよ

### Git/GitHub 演習 (個人演習)

産業技術大学院大 中鉢欣秀

### Git 入門

## GitHub 入門

## (参考)より准んだ

ファイルの削除と名前

# Git が追跡するファイルの削除と名前 の変更

- Git が追跡しているファイルであって も、シェルの rm コマンドや mv コマン ドで削除や名前の変更をしてよい
- 「git add .」コマンドを実行すると、Git は削除や名前の変更も自動的に検 知する
  - ●「git rm」や「git mv」は使わなく てよい

## Git/GitHub 演習 (個人演習)

産業技術大学院大 中鉢欣秀

Git 入門

GitHub 入門

(参考)より准んだ 使い方

ファイルの削除と名前

# Git で行った操作の取り消し

- まちがって
  - ファイルをステージングさせた!
  - ステージングをコミットした!
- などの場合、操作を取り消すことが できる
  - 特定のファイルの変更の取り消し
  - 特定のコミットの取り消し

## Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 環際の確認

を交換症のFF級 関歴の確認 ブランチの使い方

コンフリクト

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

コンフリクトに

(参考)より進んだ 使い方

\*ァイルの削除と名 )変更

操作を取り消す

業の一度中断と再開 の他知っておくと良

# HEAD によるコミットの指定

- 特定のコミットの ID を指定する方法に 「HEAD」を使った相対指定がある
  - show コマンドで確認しながら用い ると良い(下記はサンプル)

```
qit
      show
           HEAD~1
     show HEAD~1^2
2 ait
```

Git/GitHub 演習 (個人演習) 産業技術大学院大

中鉢欣秀

Git 入門

GitHub 入門

(参考)より進んだ 使い方

# ステージング/コミットの修正

## ファイルのステージングを取り消す

git reset \$FILE

\$COMMIT\_ID より後のコミットの取り消し (ローカルは保存)

git reset \$COMMIT\_ID

\$COMMIT\_ID より後のコミットの取り消し (ローカルの変更も破棄)

git reset --hard \$COMMIT\_ID

## Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

### Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認

## GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

## (参考)より進んだ 使い方

'ァイルの削除と |変更

## 操作を取り消すコマンド

の一度中断と再開 他知っておくと良いコ ド

# 誤って編集や削除や修正したファイ ルの回復

- file.txt を誤って編集や削除した場合
- add する前
  - ステージング領域からの取り出し

git checkout file.txt

- add した後
  - 直近のコミットからの取り出し

checkout HEAD file.txt

Git/GitHub 演習 (個人演習) 産業技術大学院大

中鉢欣秀

Git 入門

GitHub 入門

(参考)より進んだ 使い方

# push 済みのコミットの取り消し

最後に行ったコミットが理由でコンフリクトが発生したような場合、次の操作により、「取り消しコミット」を作成することができる

git revert HEAD

- 最後の作業が取り消されていることを 確認
- その後は、この取り消しコミットを push すると、リモートでの変更内容も 取り消される

## Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認

ランチの使い方 ンフリクト

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow コンフリクトについて

(参考) より進んだ 使い方

· 'ァイルの削除と名 '変更

操作を取り消する

F業の一度中断と再開 その他知っておくと良い アンド

# やりかけの作業の stash

- あるブランチで作業中に他のブランチ に一時的に移動したいことがある
  - 作業の途中で master ブランチを最 新にする、など
- このような場合、git stash コマンドが 活用できるので調べてみよ

## Git/GitHub 演習 (個人演習) 産業技術大学院大 学 中鉢欣秀

Git 入門
Git の操作方法と初期設定
Git のリポジトリ
コミットの作成方法
変更環際の作成

を 履歴の確認 ブランチの使い方 コンフリクト

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow コンフリクトについ<sup>っ</sup>

(参考) より進んだ 使い方

ファイルの削除と名 の変更

操作を取り消する

作業の一度中断と再開 その他知っておくと良 マンド

# チートシートにある残りのコマンド

以下、Git チートシートにあるコマンドで、ここまで取り上げなかったものを取り上げる

## Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認

GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

(参考)より進んだ 使い方

ファイルの削除と名 の変更

かるな 操作を取り消すコ 作業の一度中断と

その他知っておくと良いコマンド

# ステージングしたファイルの差分表 示

- git add でステージングすると git diff で 差分が表示されない
- この場合,次のコマンドで確認できる

git diff --staged

## Git/GitHub 演習 (個人演習) 産業技術大学院大

業技術大学院大 学 中鉢欣秀

Git 入門 Git の操作方法と初期設定 Git のリポジトリ

コミットの作成方法変更履歴の作成 履歴の確認

履歴の確認 ブランチの使い方 コンフリクト

GitHub 入門 GitHub とは

GitHub とは リモートリポジトリ GitHub flow コンフリクトについて

(参考) より進んだ 使い方 ファイルの削除と名前 の変更

ファイルの削除と名 D変更 操作を取り消すコマ

作業の一度中断と再開 その他知っておくと良いコ

# 特定のファイルにのみ関連する履歴 確認

- 将来,複数のファイルを履歴管理する ようになったら特定のファイルの履歴 のみ確認したい
- その場合,次のとおりにする

```
| git log --follow README
```

Git/GitHub 演習 (個人演習) 産業技術大学院大

業技術大学院大学院大学 学 中鉢欣秀

Git 入門
Git の操作方法と初期設定
Git のリポジトリ
コミットの作成方法
変更履歴の作成

を更履歴の作成 夏歴の確認 プランチの使い方 コンフリクト

コンフリクト GitHub 入門

itHub とは モートリポジトリ itHub flow

(参考)より進んだ使い方

-ファイルの削除と名 の変更 操作を取り消すコマ

操作を取り消すコマンド 作業の一度中断と再開 その他知っておくと良いコ

# リモートブランチの最新情報を取得

- git fetch はリモートにあるブランチの 最新情報をローカルに取ってくるコマンド
- 例えば、git pull は master ブランチで次の2つを実行することと同じ意味

```
git fetch origin
git merge origin/master
```

## Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

Git 入門

Git の操作方法と初期設定 Git のリボジトリ コミットの作成方法 変更履歴の作成 履歴の確認 プランチの使い方

> ョンフリクト itHub 入門

GitHub 入門 GitHub とは

リモートリポジトリ GitHub flow

(参考)より進んだ 使い方

: V・/J 7ァイルの削除と: )変更

の変更 操作を取り消すコ

作薬の一肢中断と再開 その他知っておくと良いコ マンド

## git Is-files

• 省略

## Git/GitHub 演習 (個人演習)

産業技術大学院大 学 中鉢欣秀

### Git 入門

Gitの操作方法と初期設定 Gitのリポジトリ コミットの作成方法 変更履歴の作成 履歴の確認 プランチの使い方

## GitHub 入門

GitHub とは リモートリポジトリ GitHub flow

コンフリクトにつ

## (参考)より進んだ 使い方

ファイルの削除と の変更

操作を取り消すコ

その他知っておくと良いコ マンド