

Git & Github

1: Introduction

2: How to use GitKraken

3: Team Development with Github flow



Daisky Nakamura



過去に戻りたい！と思った事ありませんか？



git

できるんです。そう、Gitならね



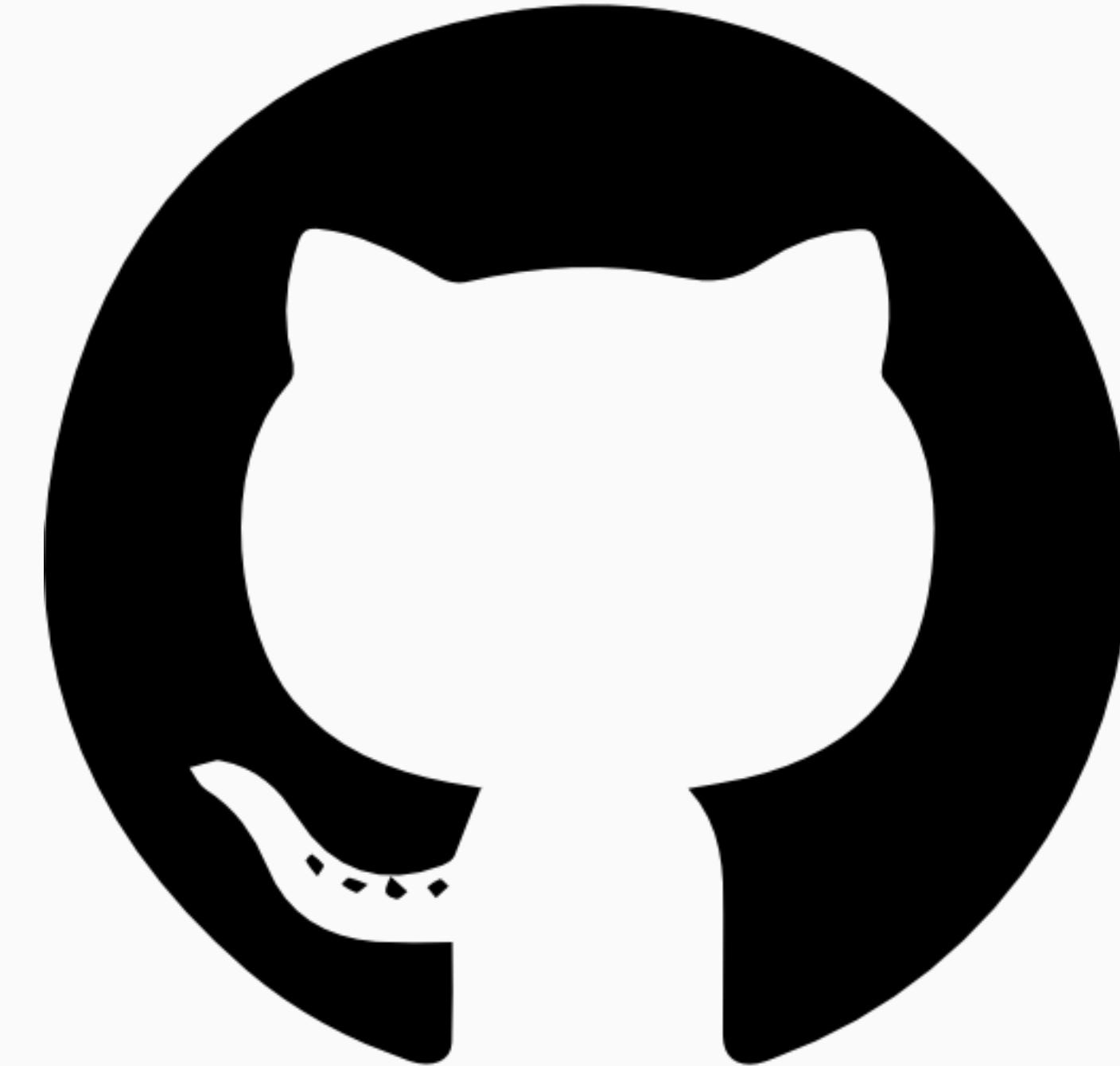
でもGit難しそう



簡単なんです。そう、GitKrakenならね



チーム開発ってどうやって進めるんだ？



Githubがあれば大丈夫

Agenda

- 1: Gitとは
- 2: 1人でGitを使う方法
- 3: チームでGitを使う方法

Flow

1: Gitとは

Study

2: 1人でGitを使う方法

Practice

3: チームでGitを使う方法

Study

Practice

Goal



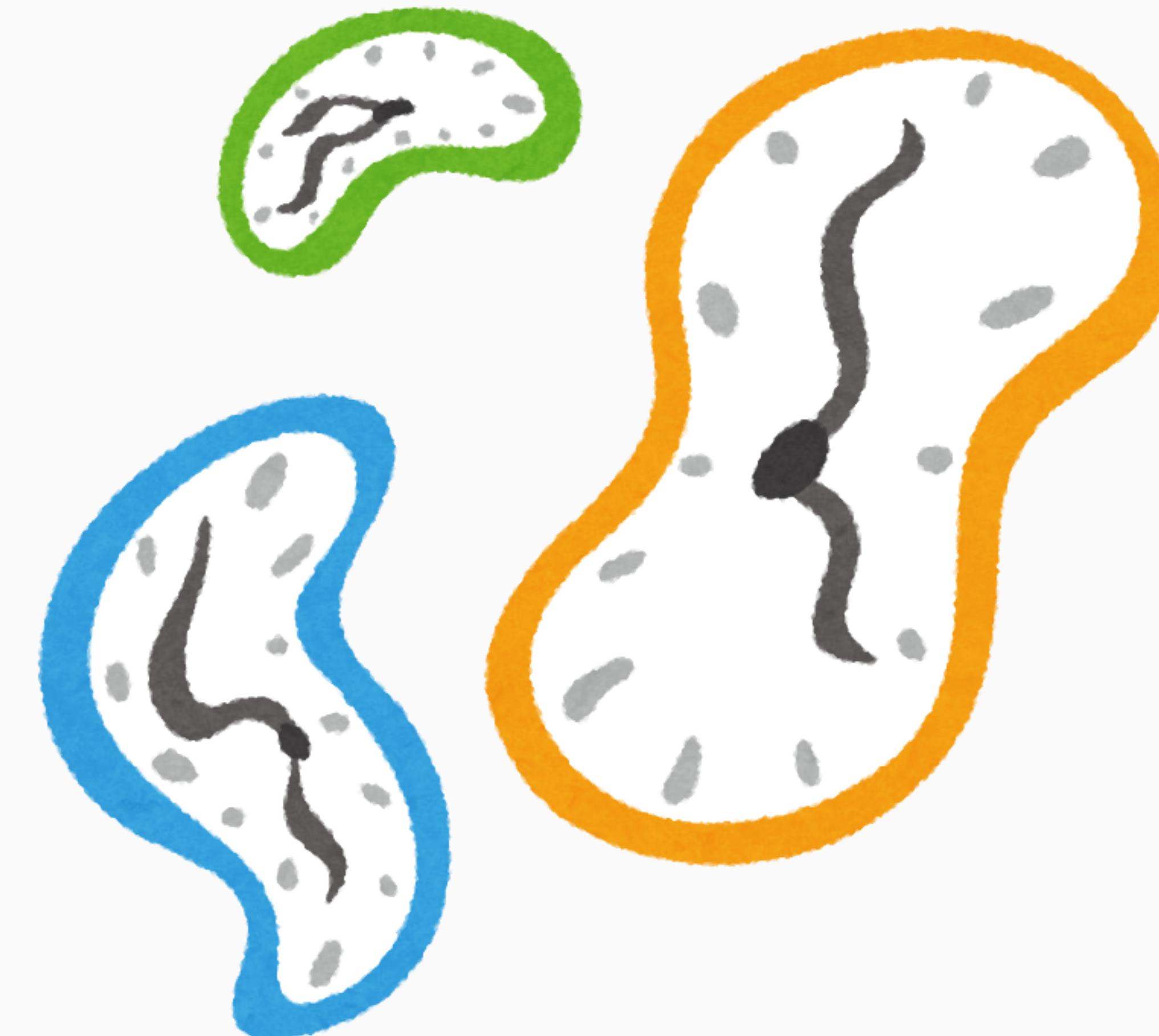
Git&Githubでスムーズなチーム開発を

Git とは

Intoroduction

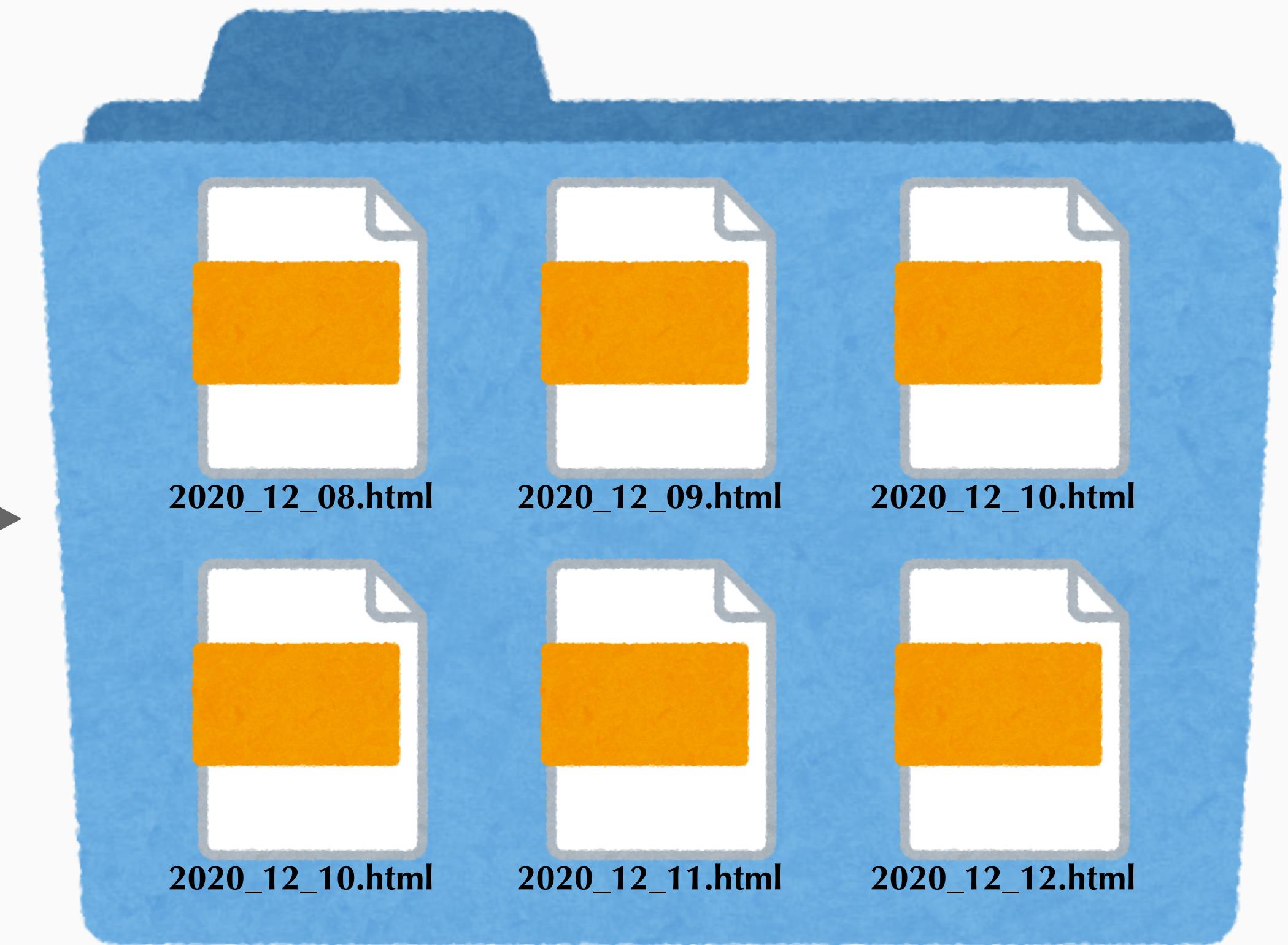
Gitとは

Study



あの時に戻りたい！を叶えるタイムマシン

Gitがなかった時



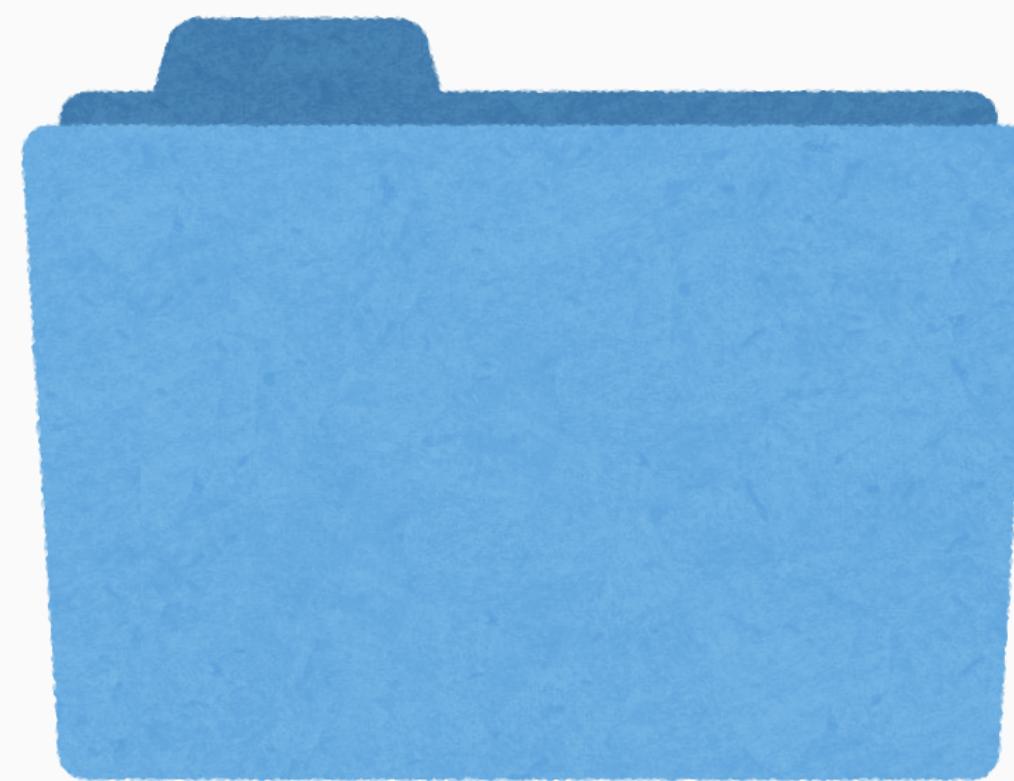
バージョン違いのファイルが溢れてた

その結果



どのファイルが何の作業か解らない

Gitがある時

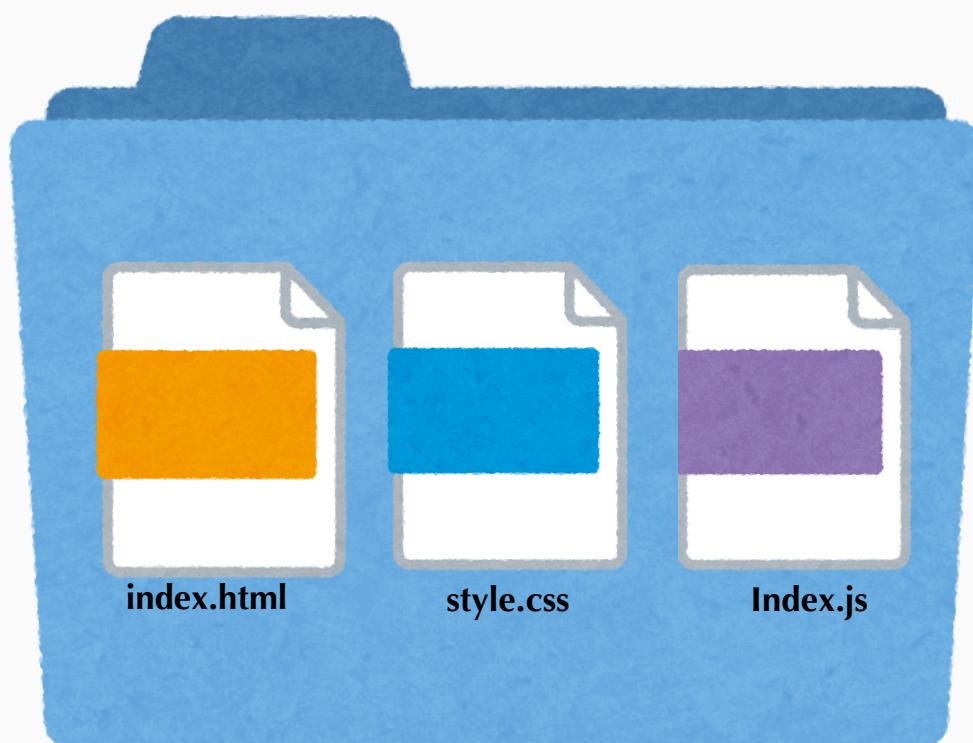


- JSを作成
- CSSを作成
- HTMLを作成

作業の履歴を保存する事ができる

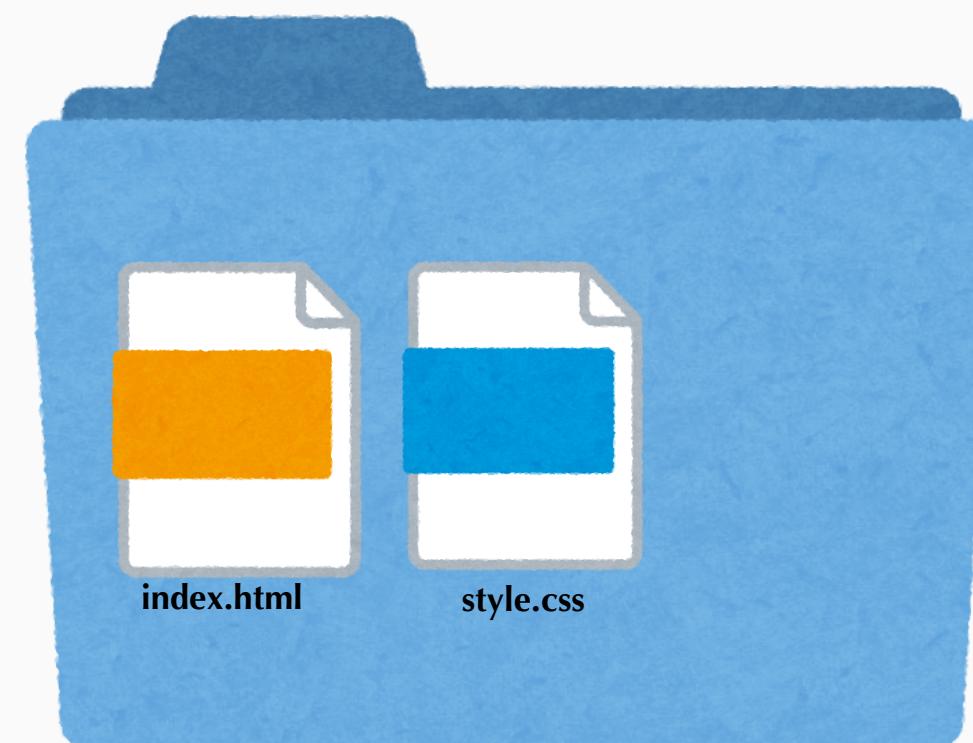
履歴があると何が良いか

Before

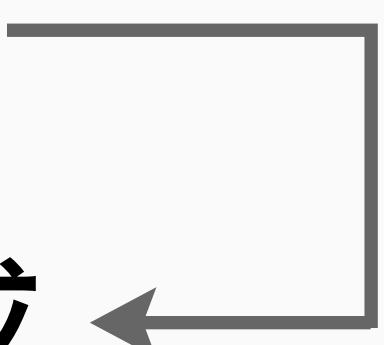


- JSを作成
- CSSを作成
- HTMLを作成

After

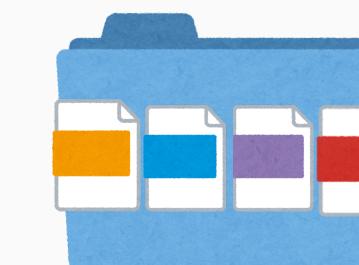
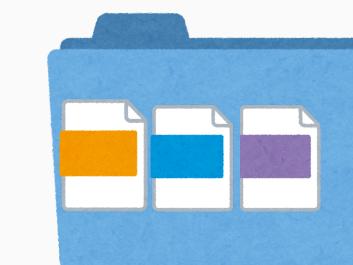
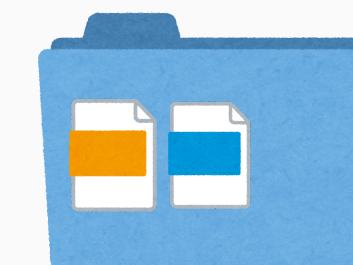


- JSを作成
- CSSを作成
- HTMLを作成



過去の状態に戻れる

過去に戻るために必要な物



- ・ HTMLを作成

- ・ CSSを作成

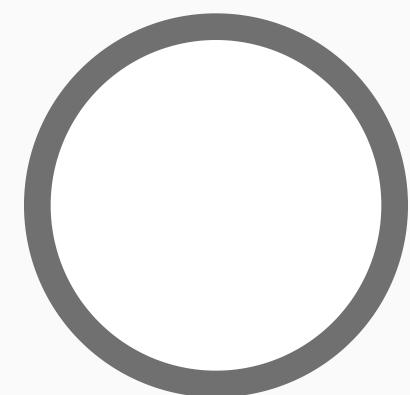
- ・ JSを作成

- ・ PHPを作成

チェックポイント

重要な用語①

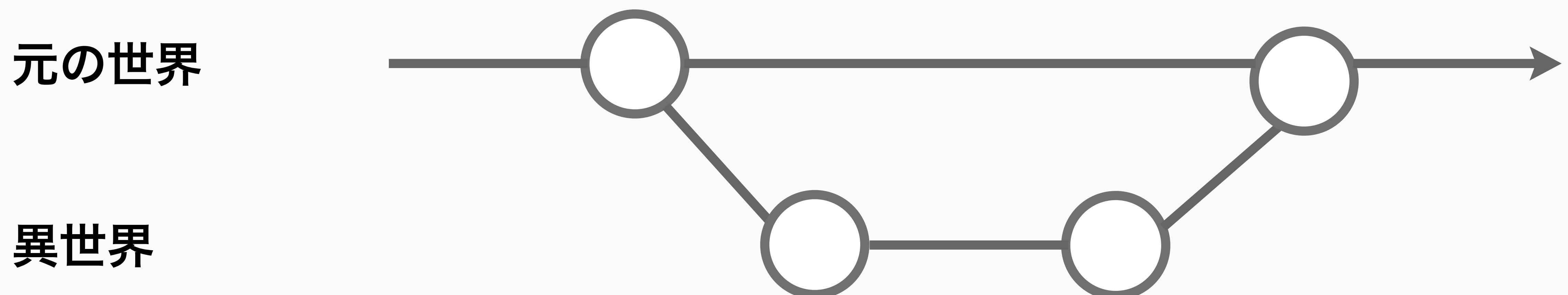
Study



Commit

チェックポイントをコミットと呼ぶ

さらに便利な仕組み



コミットからパラレルワールドへ行ける

いつ異世界へ行くべきか

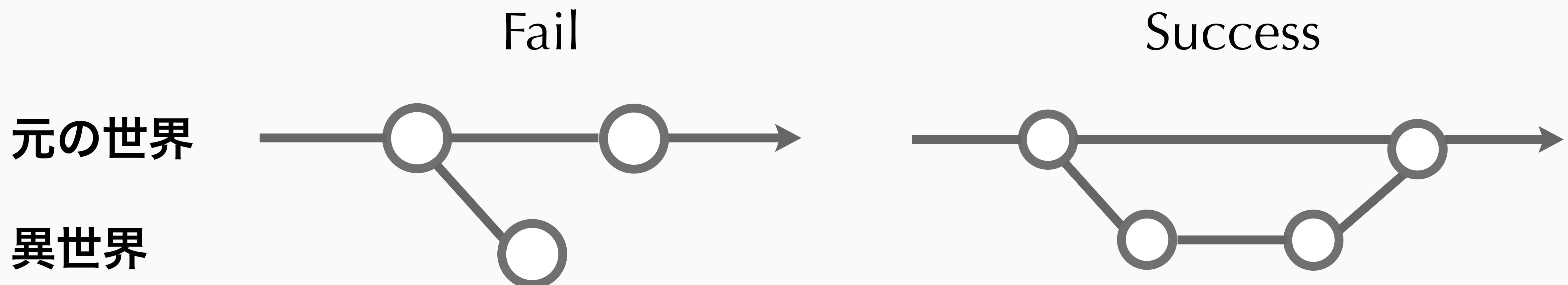
Study



新しい機能を思いついた時

失敗したら異世界は壊せば良い

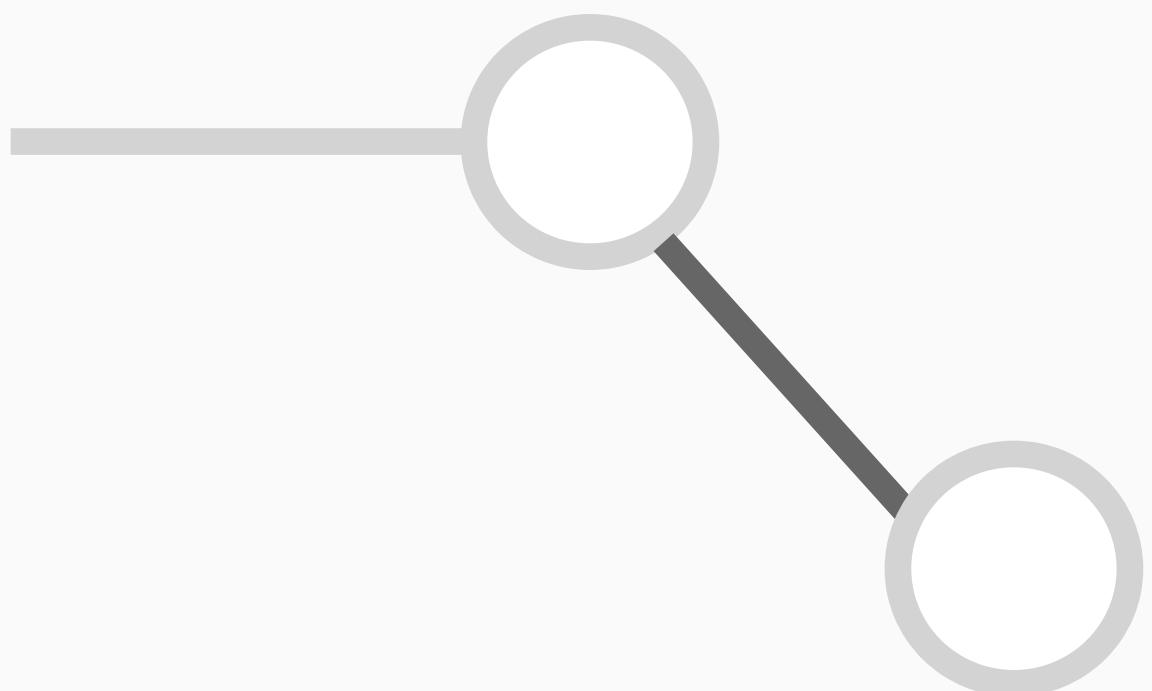
Study



成功したら元の世界に戻す

重要な用語②

Study

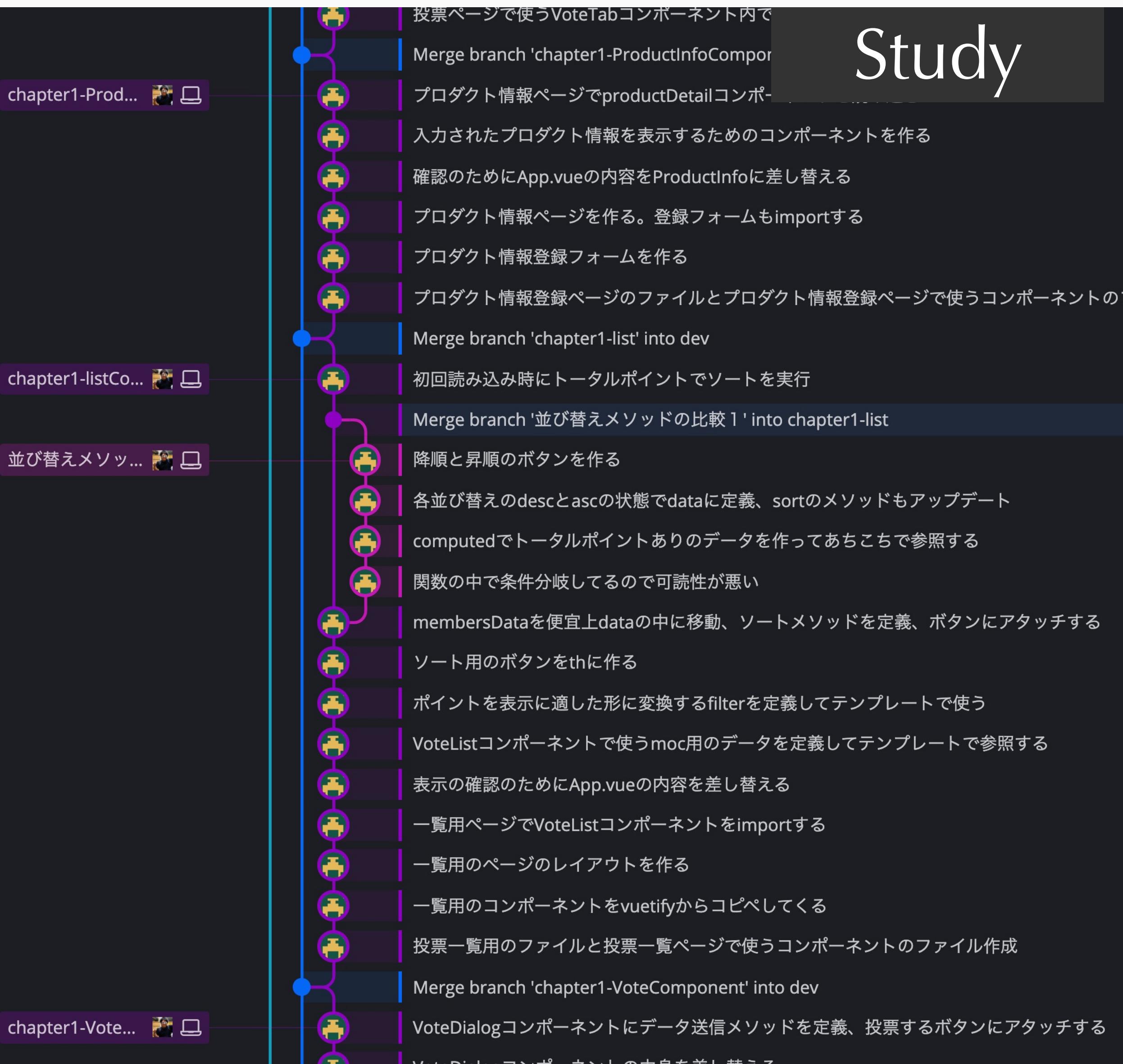


Branch

異世界をブランチと呼ぶ

ブランチの例

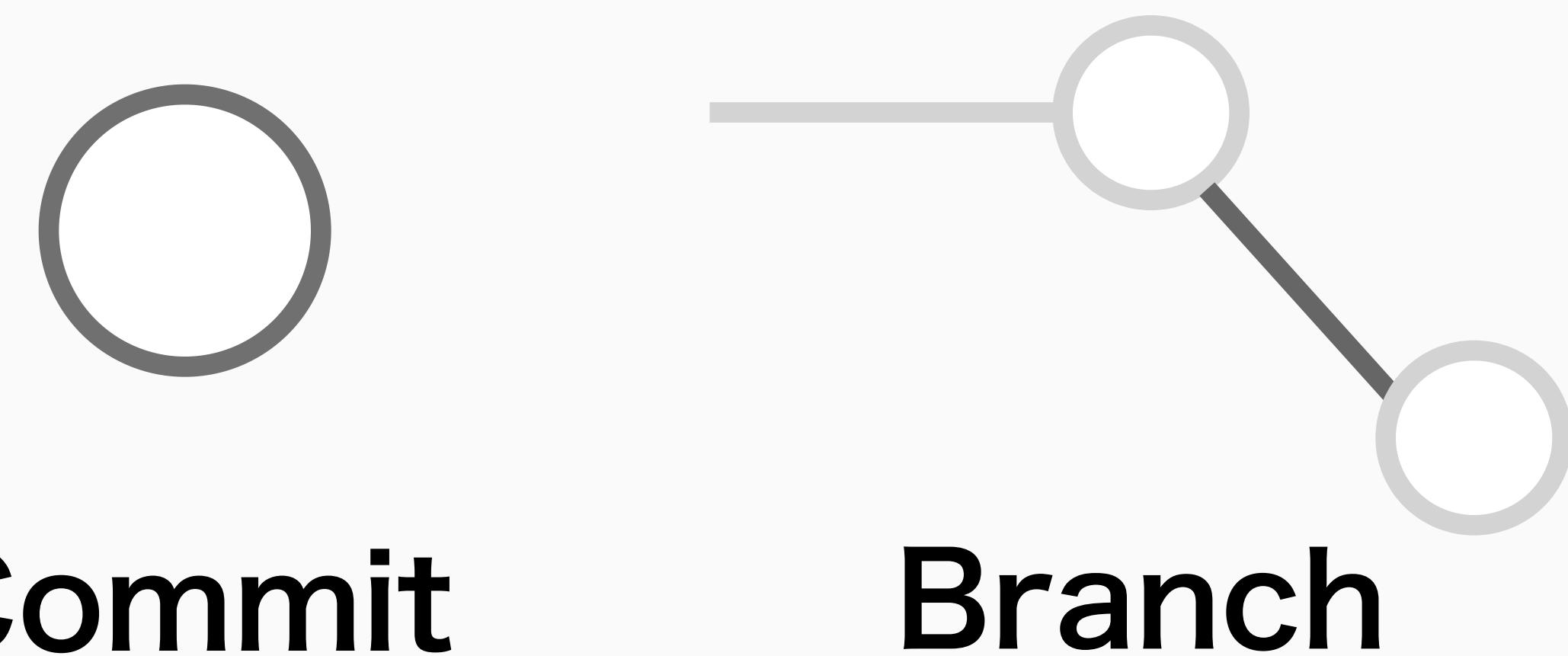
イメージつきづらいかもしませんが、
やればわかります



Study

要するに

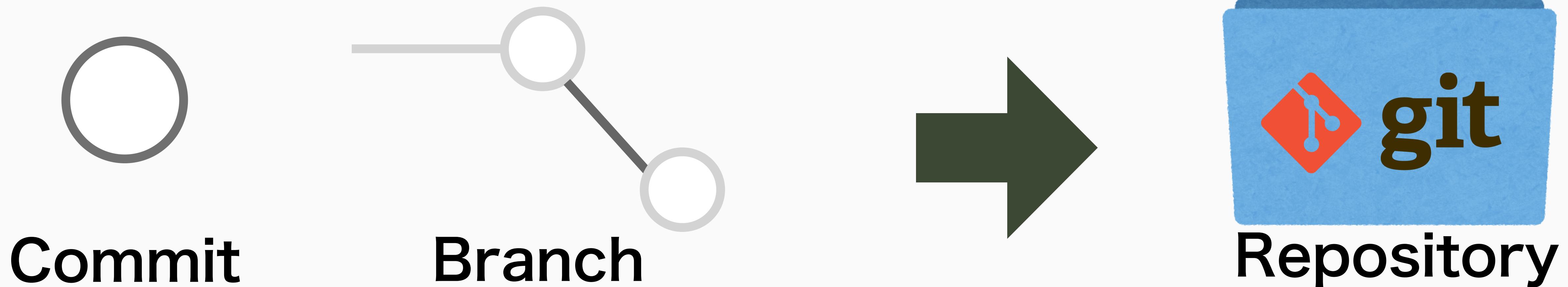
Study



チェックポイントと異世界を駆使して開発する

重要な用語③

Study



Gitで履歴を管理するフォルダをリポジトリと呼ぶ

重要な用語④

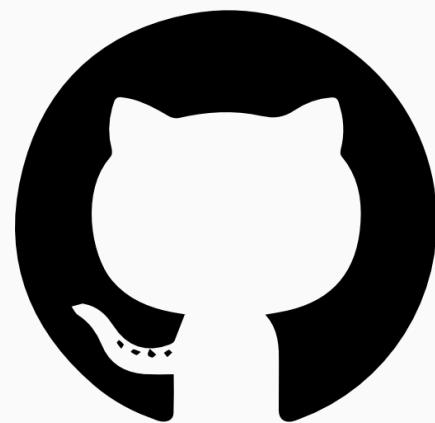
Study

Local



Local Repository

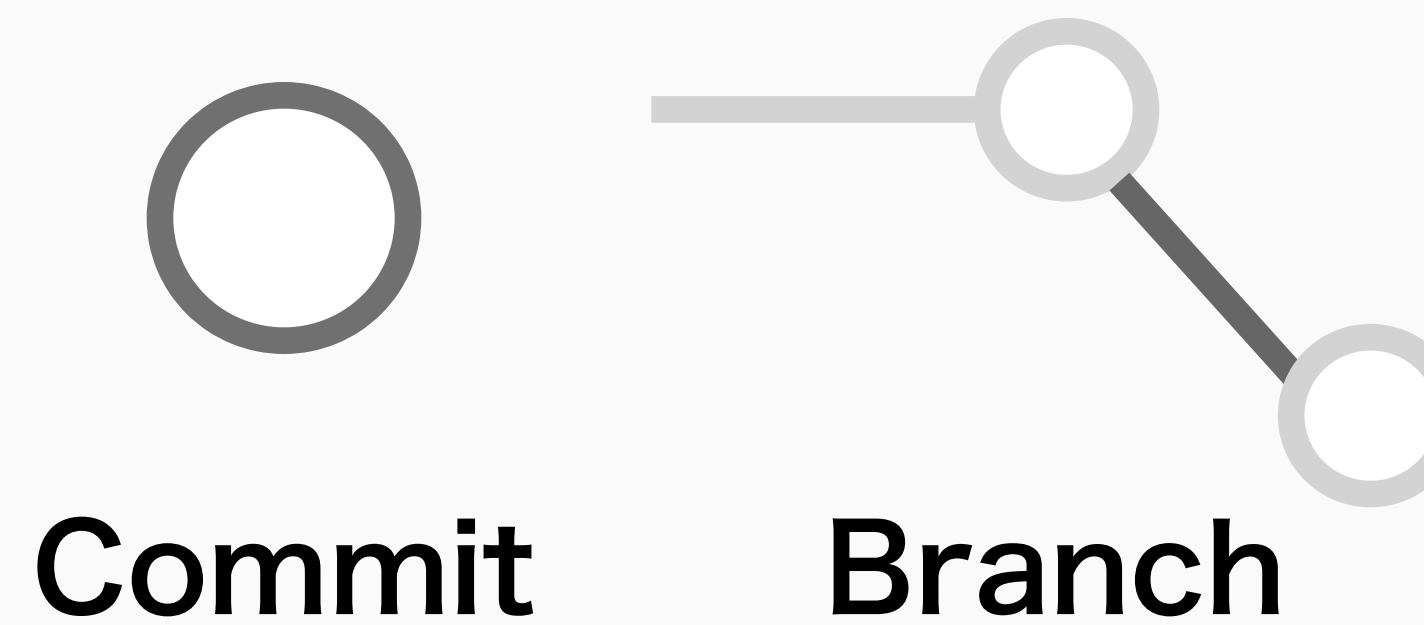
Remote



Remote Repository

Github上のリポジトリはリモートリポジトリと呼ぶ

ここまでまとめ



- ・コミットは過去に戻るためのチェックポイント
- ・ブランチは異世界

GitはSF

1人でGitを使う方法

How to use GitKraken

Practice



GitKrakenを使い方を説明します

やること

Practice

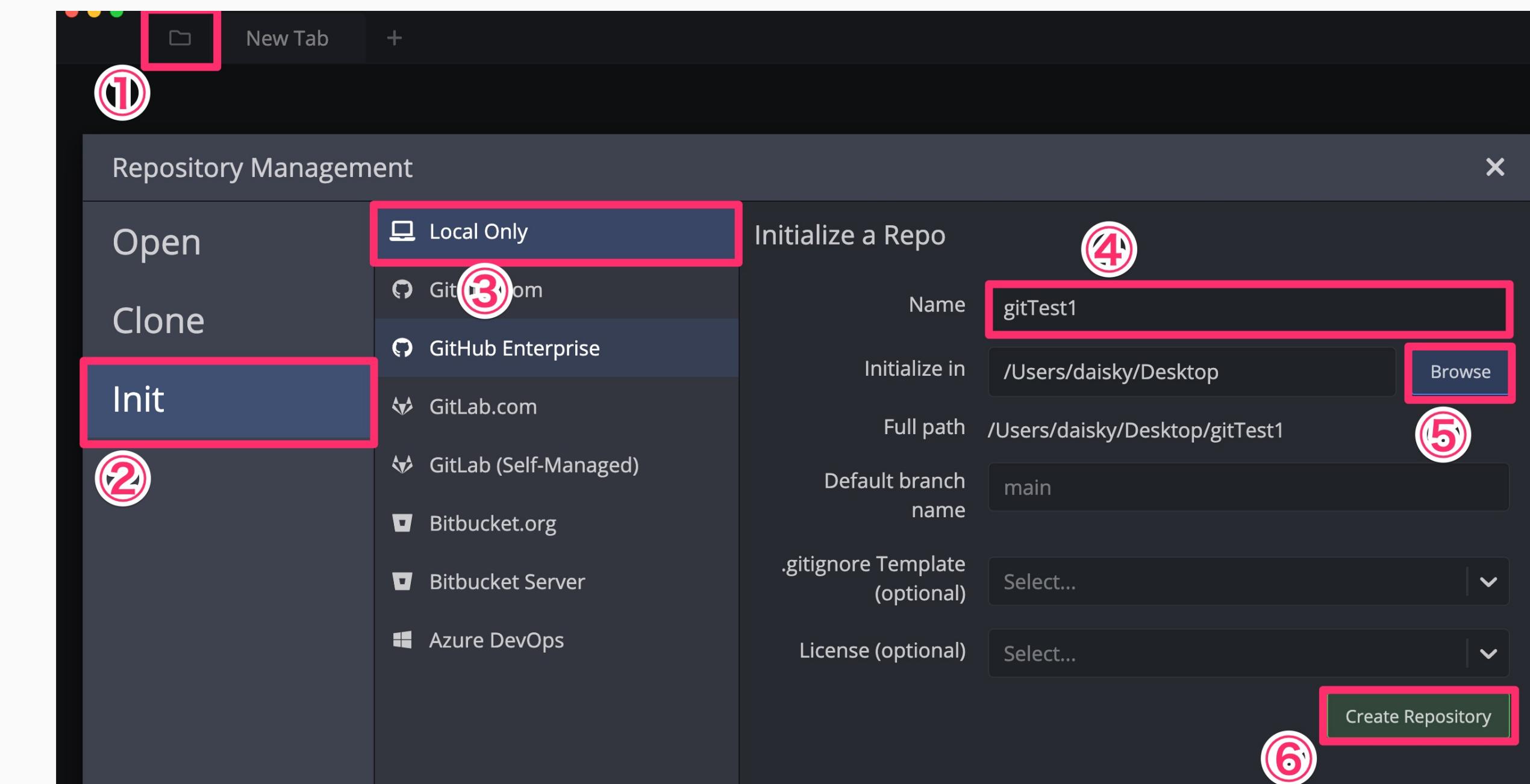
- ローカルリポジトリの作成
- はじめてのコミット
- はじめての取り消し
- はじめての取り消しの取り消し
- はじめてのブランチ
- はじめてのマージ
- はじめてのコンフリクト
- はじめてのプッシュ
- はじめての.gitignore

GitKrakenの使い方をザーッと見ます

Practice

リポジトリの作成

1. 左上のフォルダをクリック
2. Initをクリック
3. LocalOnlyをクリック
4. repositoryの名前を決める
5. PC上のフォルダを作る場所を指定
6. 作成をクリック

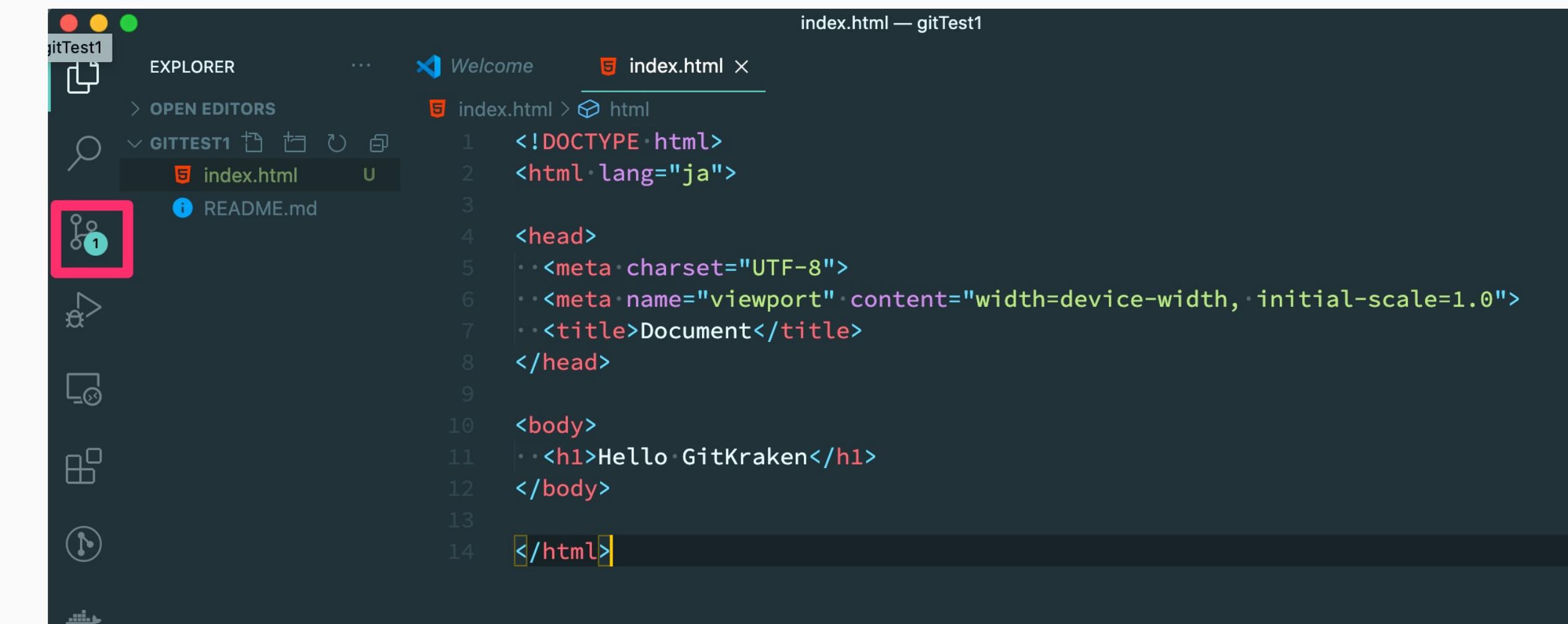


指定した場所にフォルダが出来ます

はじめてのコミット①

Practice

1. 作成したフォルダをVSCodeで開く
2. HTMLを作ります



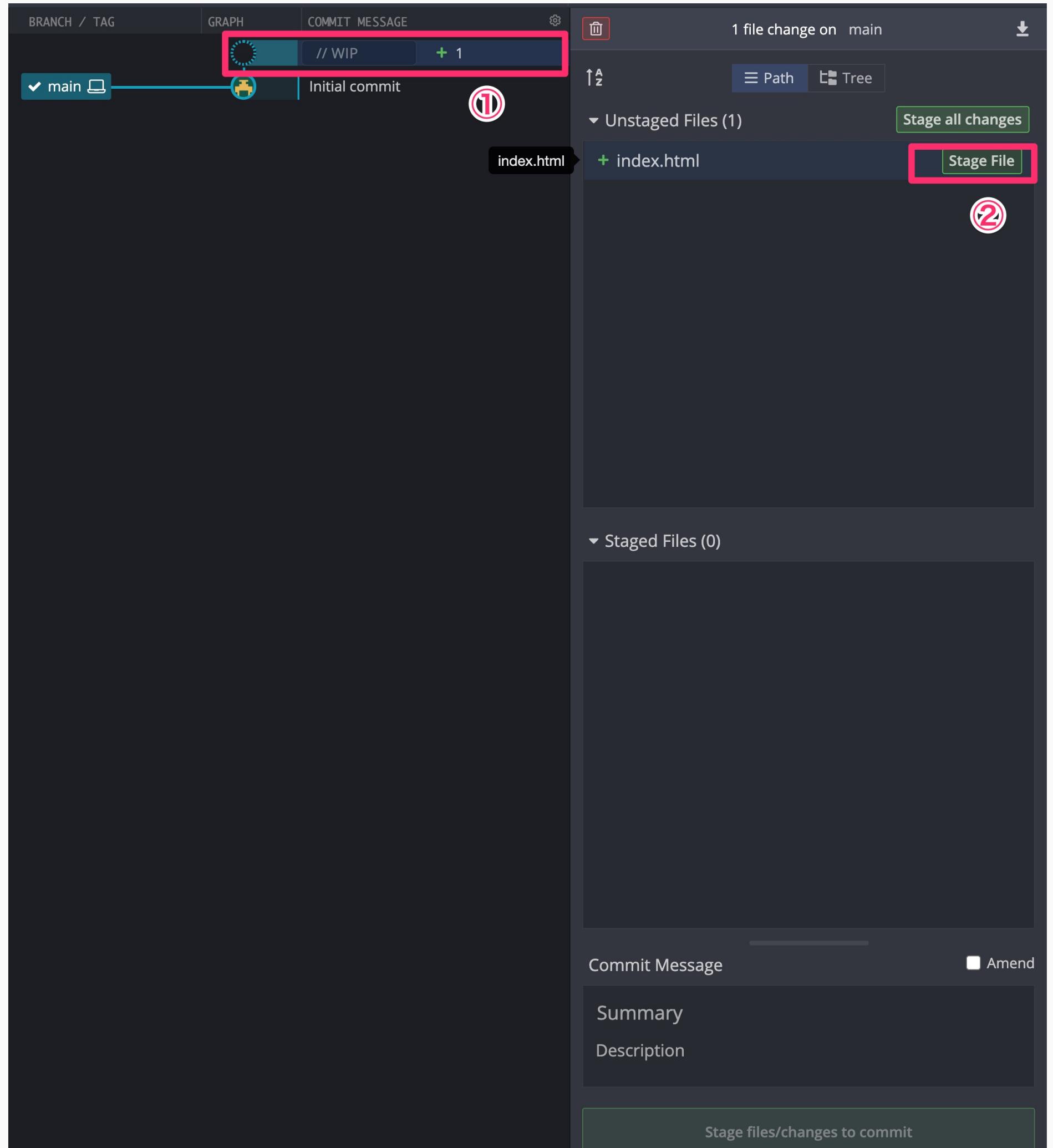
```
<!DOCTYPE html>
<html lang="ja">
<head>
..<meta charset="UTF-8">
..<meta name="viewport" content="width=device-width, initial-scale=1.0">
..<title>Document</title>
</head>
<body>
..<h1>Hello GitKraken</h1>
</body>
</html>
```

この場合はDesktop > gitTest1

はじめてのコミット②

Practice

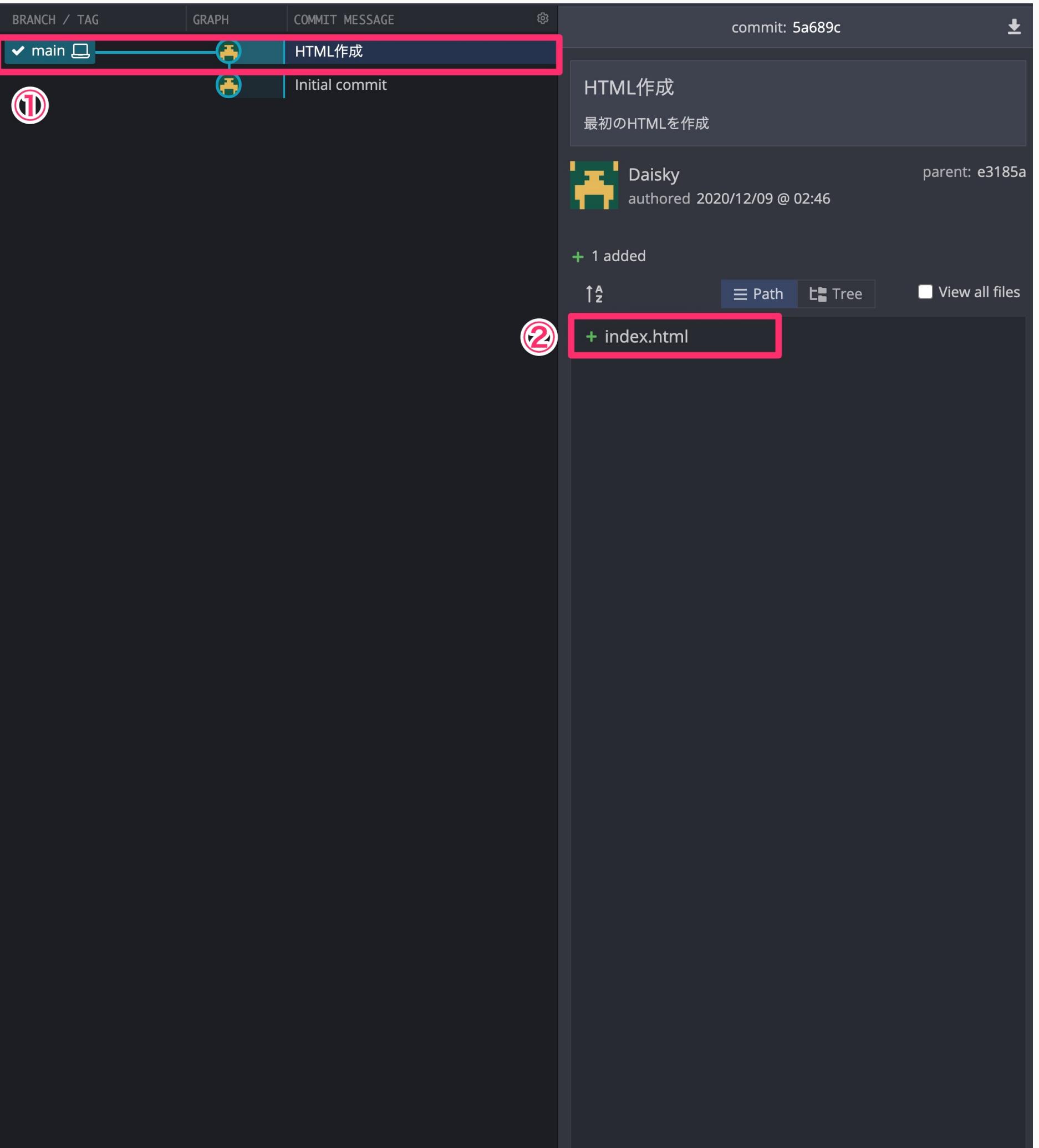
1. //WIPをクリック
2. ホバーするとボタンが出るのでクリック



はじめてのコミット③

Practice

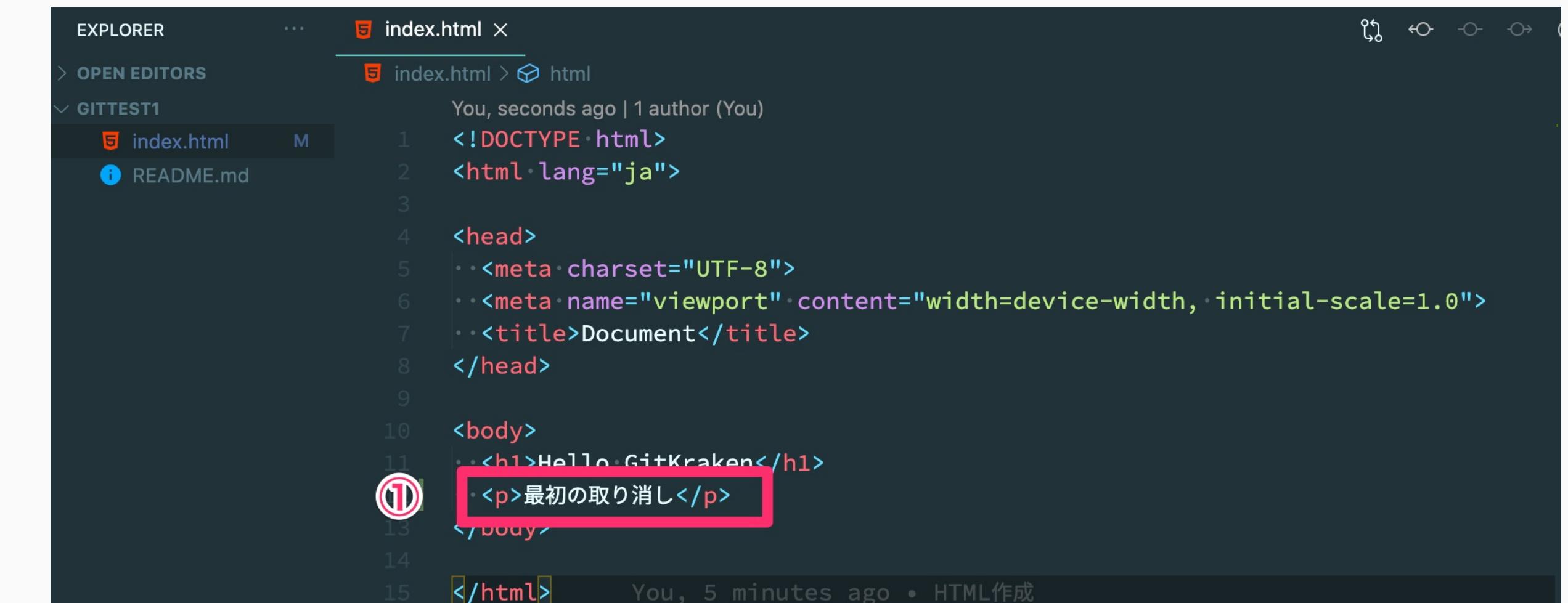
1. //WIPが変化！初めてのコミット成功！
2. ここをクリックすると変更の内容が見える



はじめての取り消し①

Practice

1. VSCodeでHTMLを編集する



A screenshot of the Visual Studio Code (VSCode) interface. The title bar shows "index.html". The Explorer sidebar on the left lists "OPEN EDITORS" and "GITTEST1" which contains "index.html" and "README.md". The main editor area displays the following HTML code:

```
<!DOCTYPE html>
<html lang="ja">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
</head>
<body>
<h1>Hello GitKraken</h1>
<p>最初の取り消し</p>
</body>
</html>
```

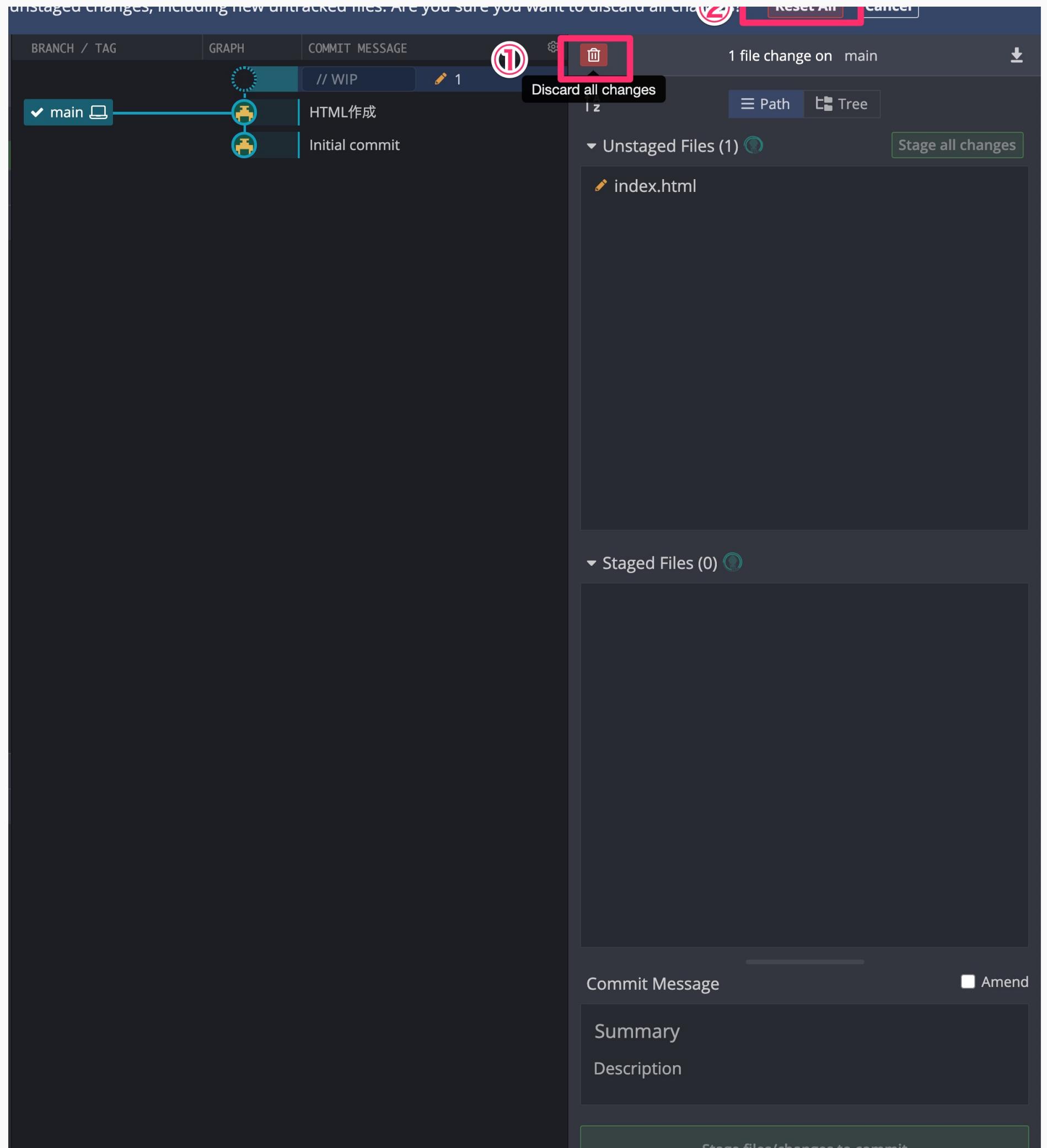
The line containing the text "最初の取り消し" is highlighted with a red rectangular selection. A small circular icon with a "T" inside is positioned to the left of the line number 11.

内容はなんでも良いです

はじめての取り消し②

Practice

1. ゴミ箱をクリック
2. Rest Allをクリック



はじめての取り消し③

Practice

The screenshot shows the GitKraken application interface. On the left, the Explorer sidebar displays a repository named 'GITTEST1' containing two files: 'index.html' and 'README.md'. The 'index.html' file is selected and highlighted with a dark gray background. In the main central area, the code editor shows the content of 'index.html'. The code is as follows:

```
1 <!DOCTYPE html>
2 <html lang="ja">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9
10 <body>
11   <h1>Hello GitKraken</h1>
12 </body>
13
14 </html>
```

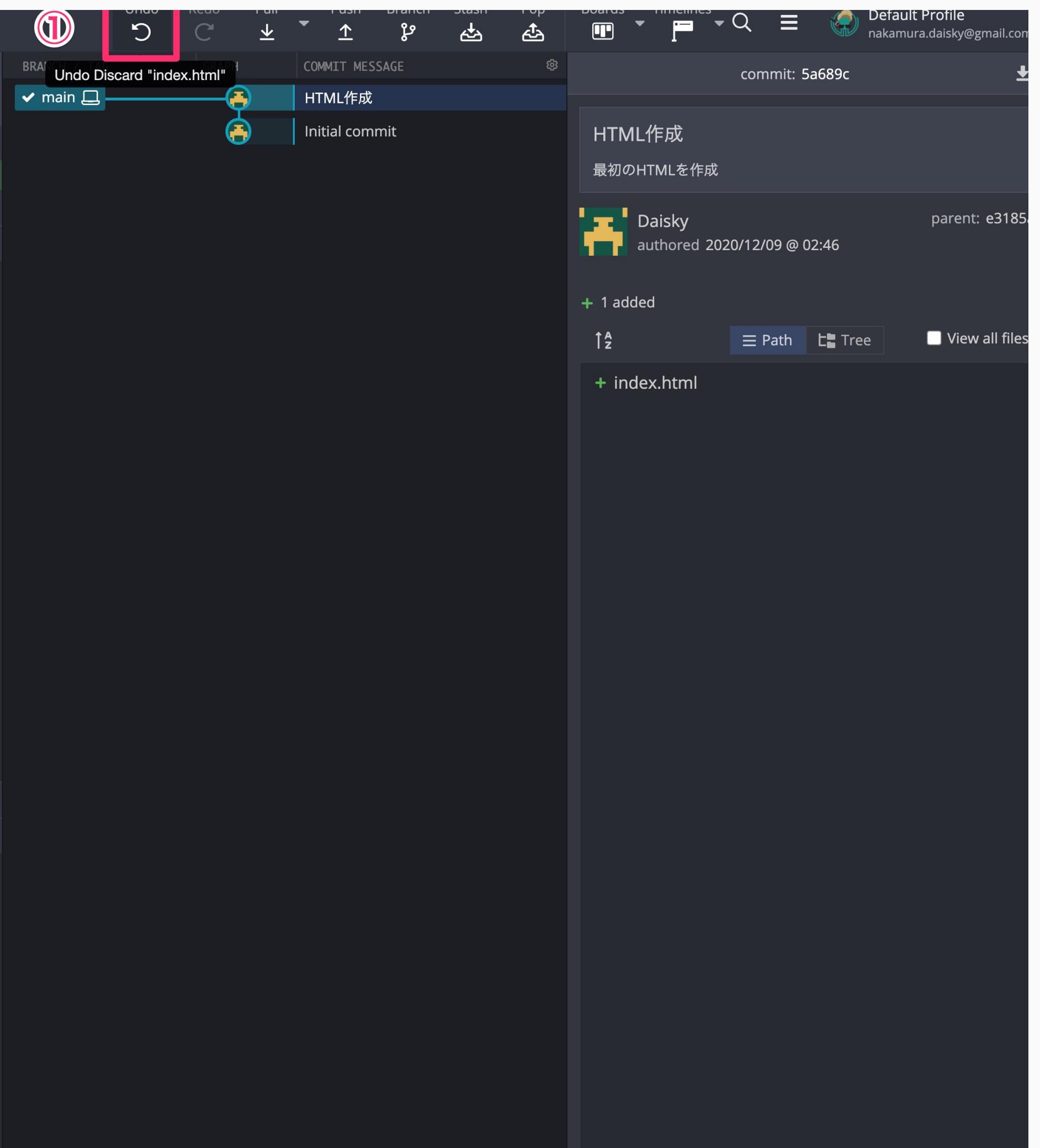
The status bar at the bottom right of the code editor indicates 'You, 16 minutes ago • HTML作成'.

元に戻ってる！

取り消しの取り消し①

Practice

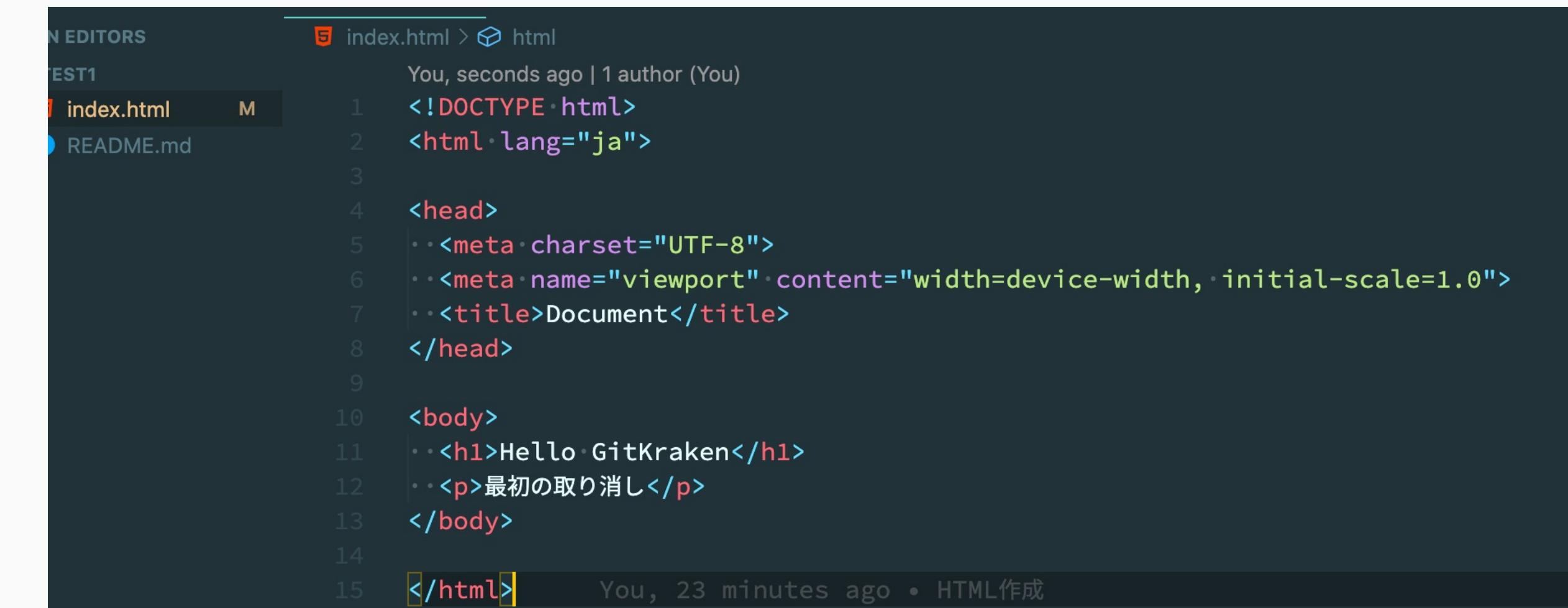
1. UnDoをクリック



取り消しの取り消し②

Practice

元の元に戻ってる！



The screenshot shows the GitKraken interface with a dark theme. On the left, the sidebar lists 'N EDITORS' and 'TEST1' with files 'index.html' and 'README.md'. The main area displays the content of 'index.html' as follows:

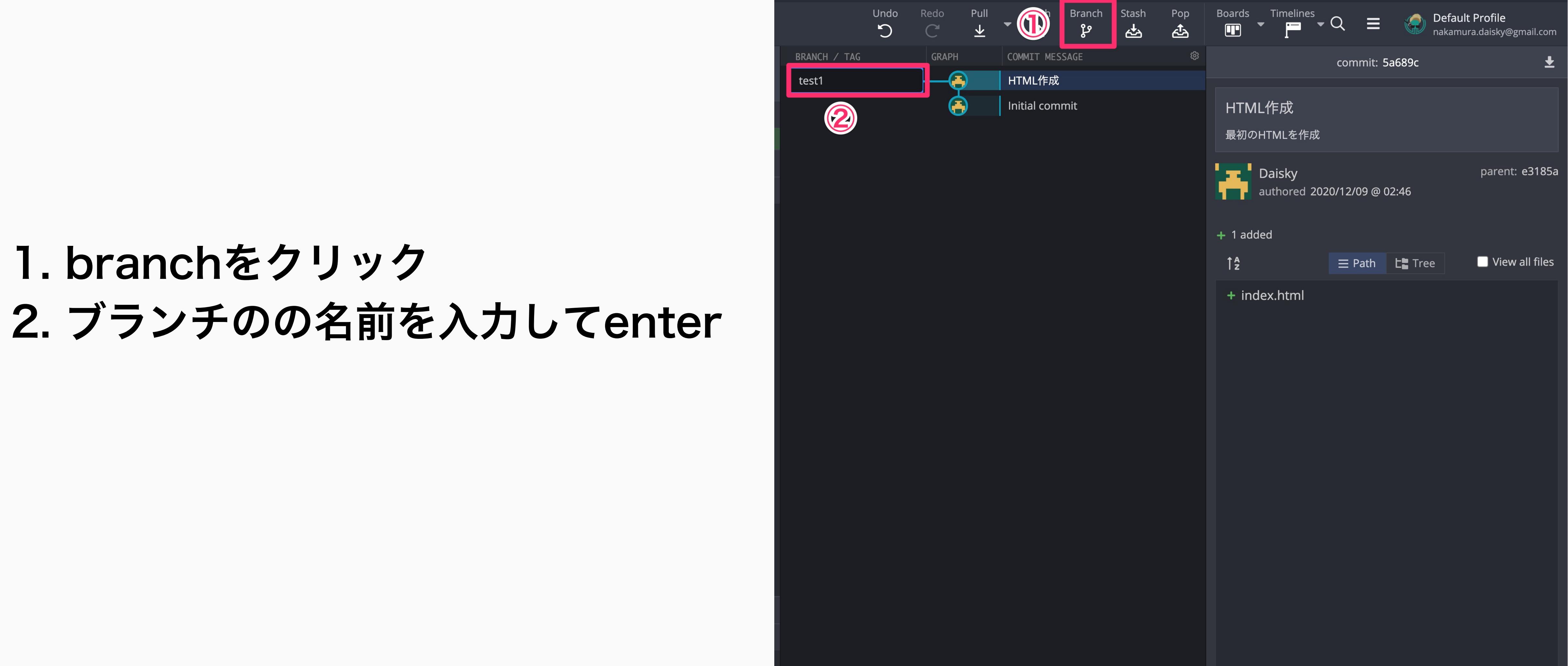
```
index.html > html
You, seconds ago | 1 author (You)
1 <!DOCTYPE html>
2 <html lang="ja">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9
10 <body>
11   <h1>Hello GitKraken</h1>
12   <p>最初の取り消し</p>
13 </body>
14
15 </html>
```

At the bottom right, it says 'You, 23 minutes ago • HTML作成'.

⌘ + zでも出来ます (winはcontrol + z)

はじめてのブランチ①

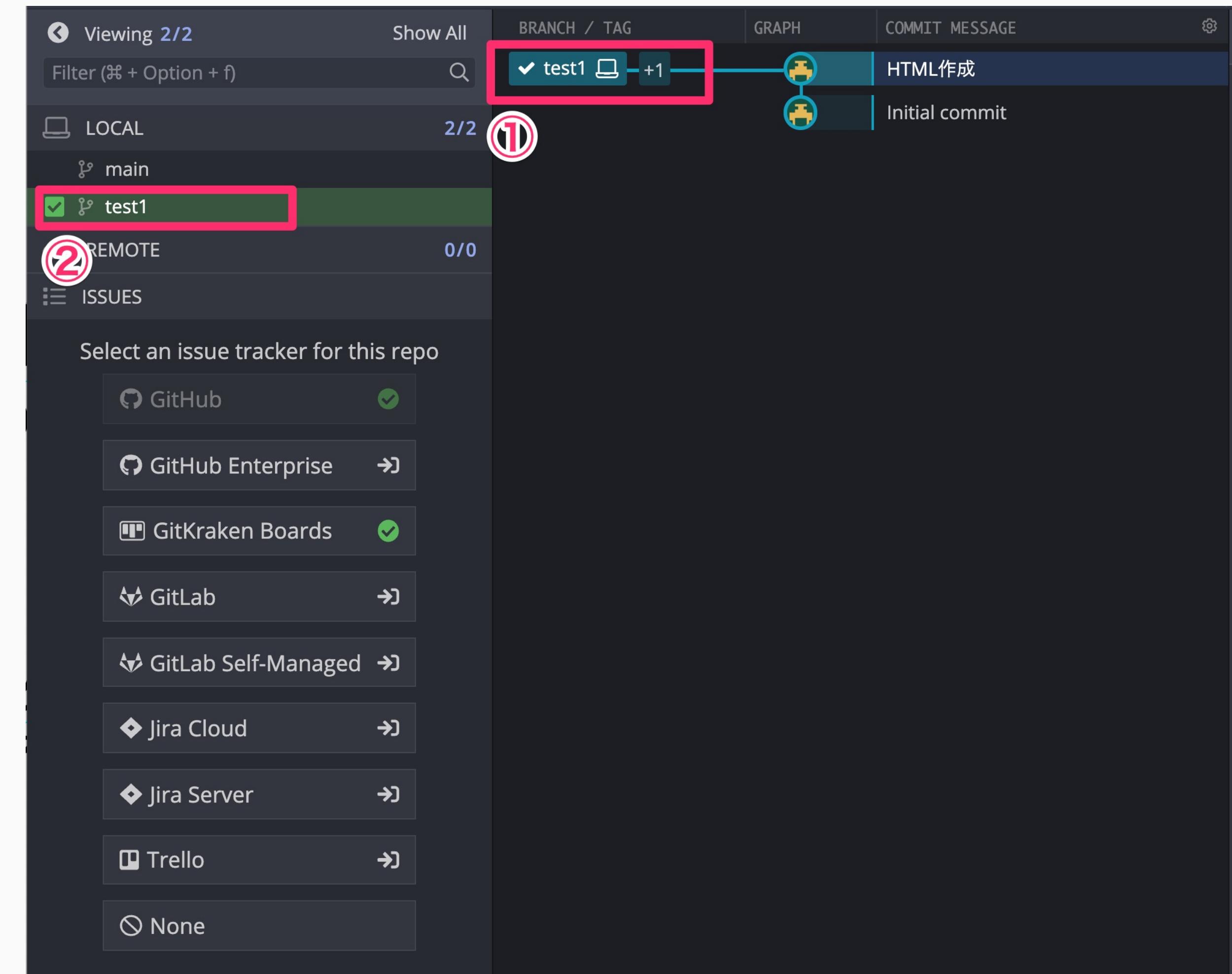
Practice



はじめてのブランチ②

Practice

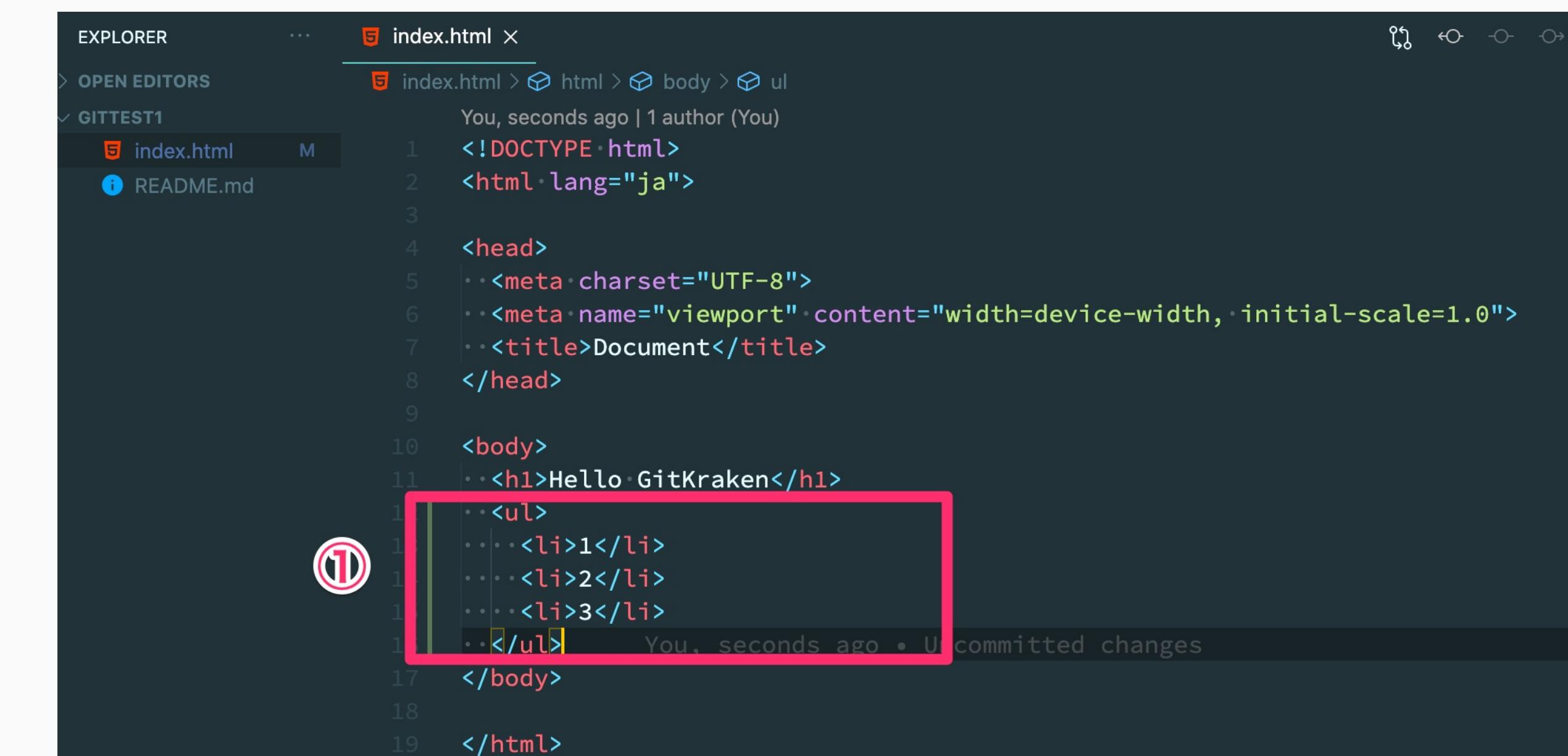
1. ブランチが変わっている
2. ブランチが増えている



はじめてのブランチ③

Practice

1. VSCodeでHTMLを変更



A screenshot of the Visual Studio Code interface. The Explorer sidebar shows a folder named 'GITTEST1' containing 'index.html' and 'README.md'. The 'index.html' file is open in the editor. The code is as follows:

```
<!DOCTYPE html>
<html lang="ja">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
</head>
<body>
<h1>Hello GitKraken</h1>
<ul>
<li>1</li>
<li>2</li>
<li>3</li>
</ul>
</body>
</html>
```

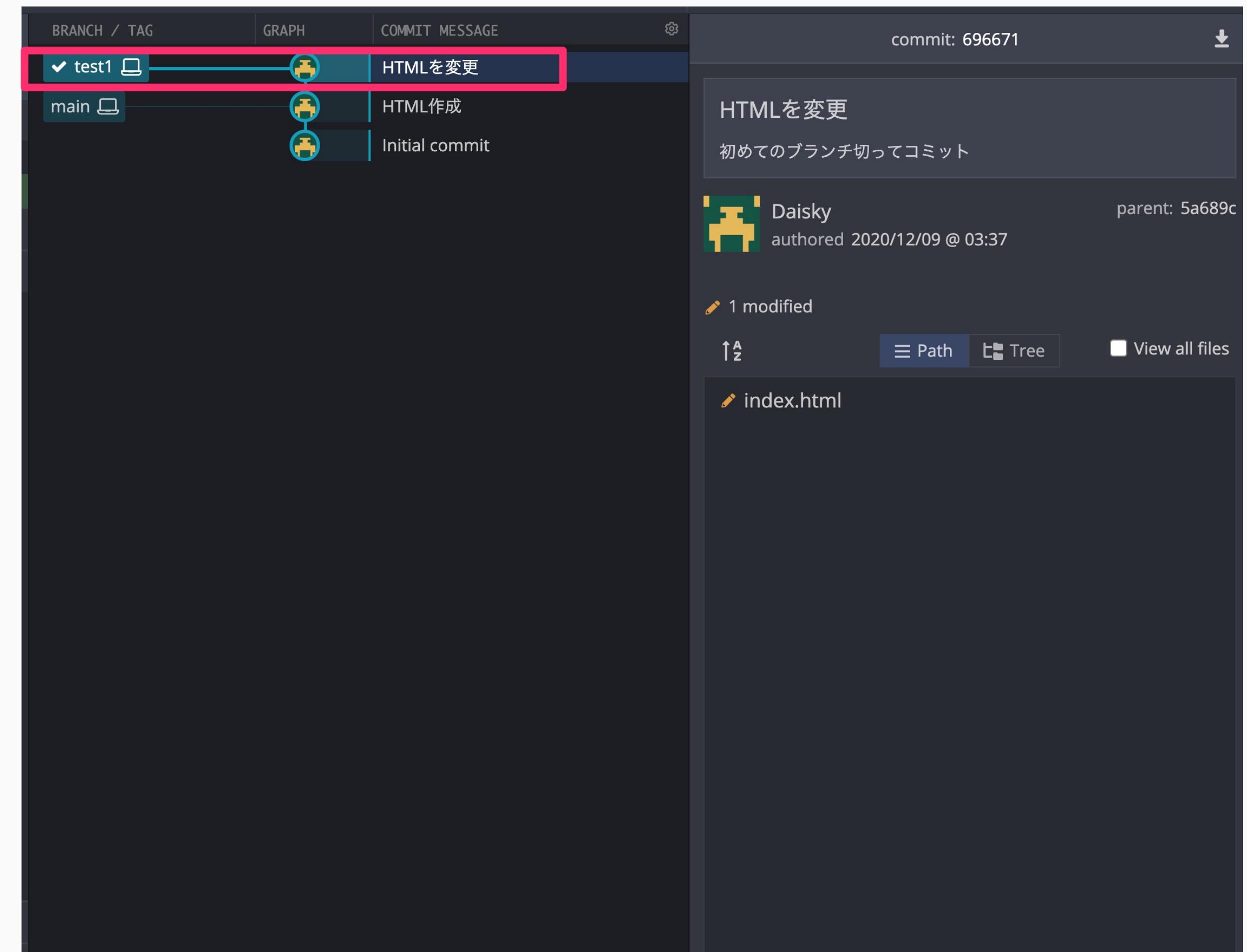
A red box highlights the line containing the three list items. A circular icon with a '1' is positioned to the left of the line numbers, indicating there is one uncommitted change.

内容は何でも良いです

はじめてのブランチ④

Practice

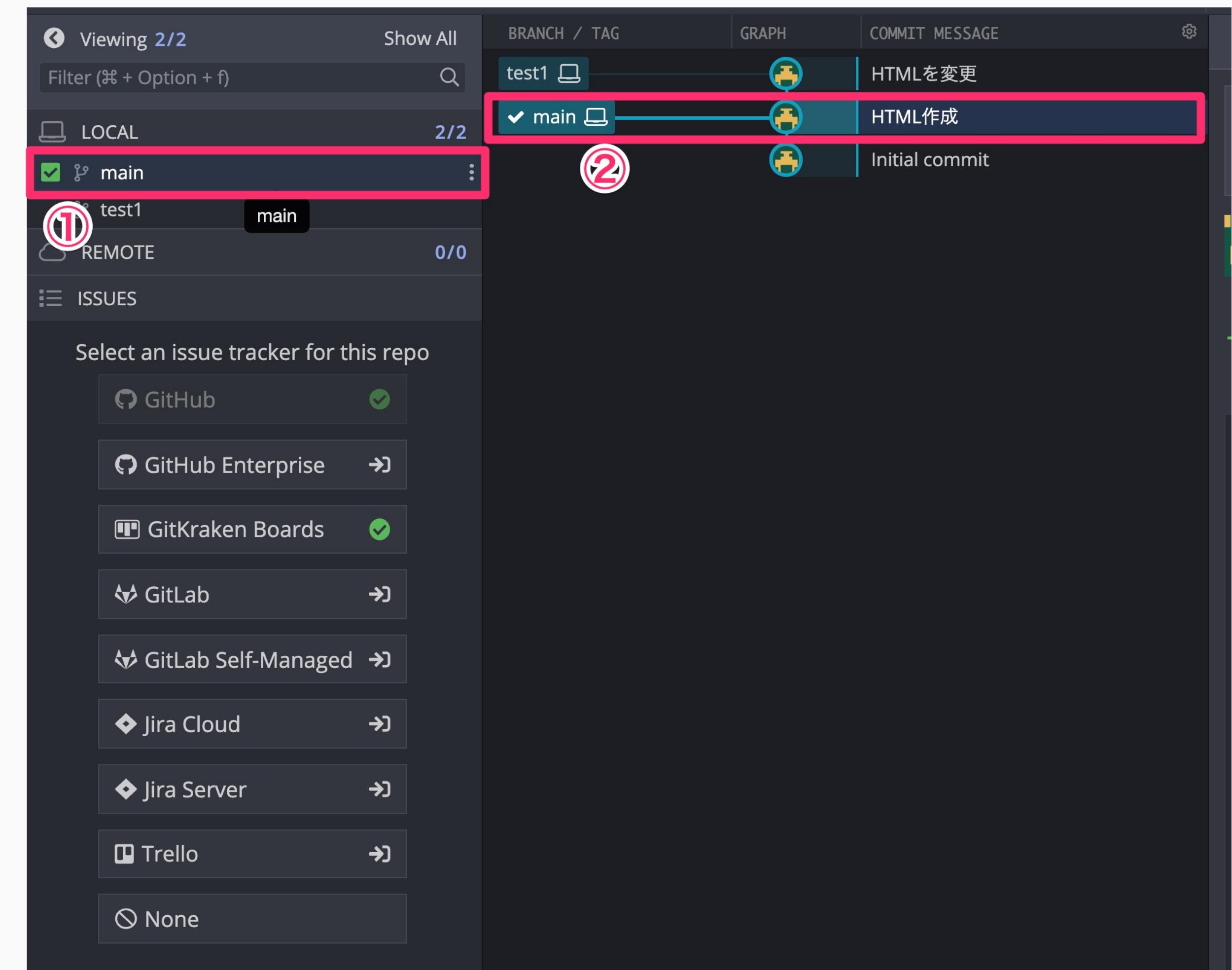
変更をコミットする



はじめてのブランチ⑤

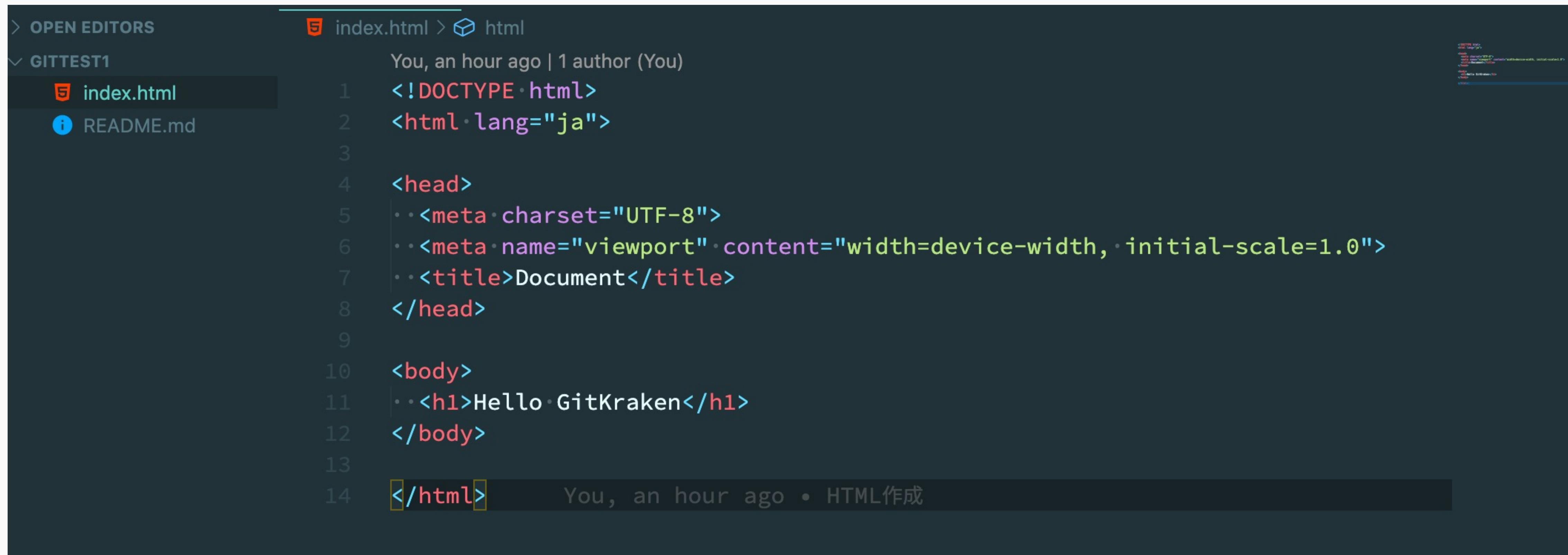
Practice

1. ダブルクリック
2. ブランチが変わるのがわかる



はじめてのブランチ⑥

Practice



The screenshot shows the GitKraken interface with a dark theme. On the left, the sidebar lists 'OPEN EDITORS' and a repository named 'GITTEST1' containing 'index.html' and 'README.md'. The 'index.html' file is currently open in the editor. The code editor displays the following HTML content:

```
You, an hour ago | 1 author (You)
1 <!DOCTYPE html>
2 <html lang="ja">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9
10 <body>
11   <h1>Hello GitKraken</h1>
12 </body>
13
14 </html>
```

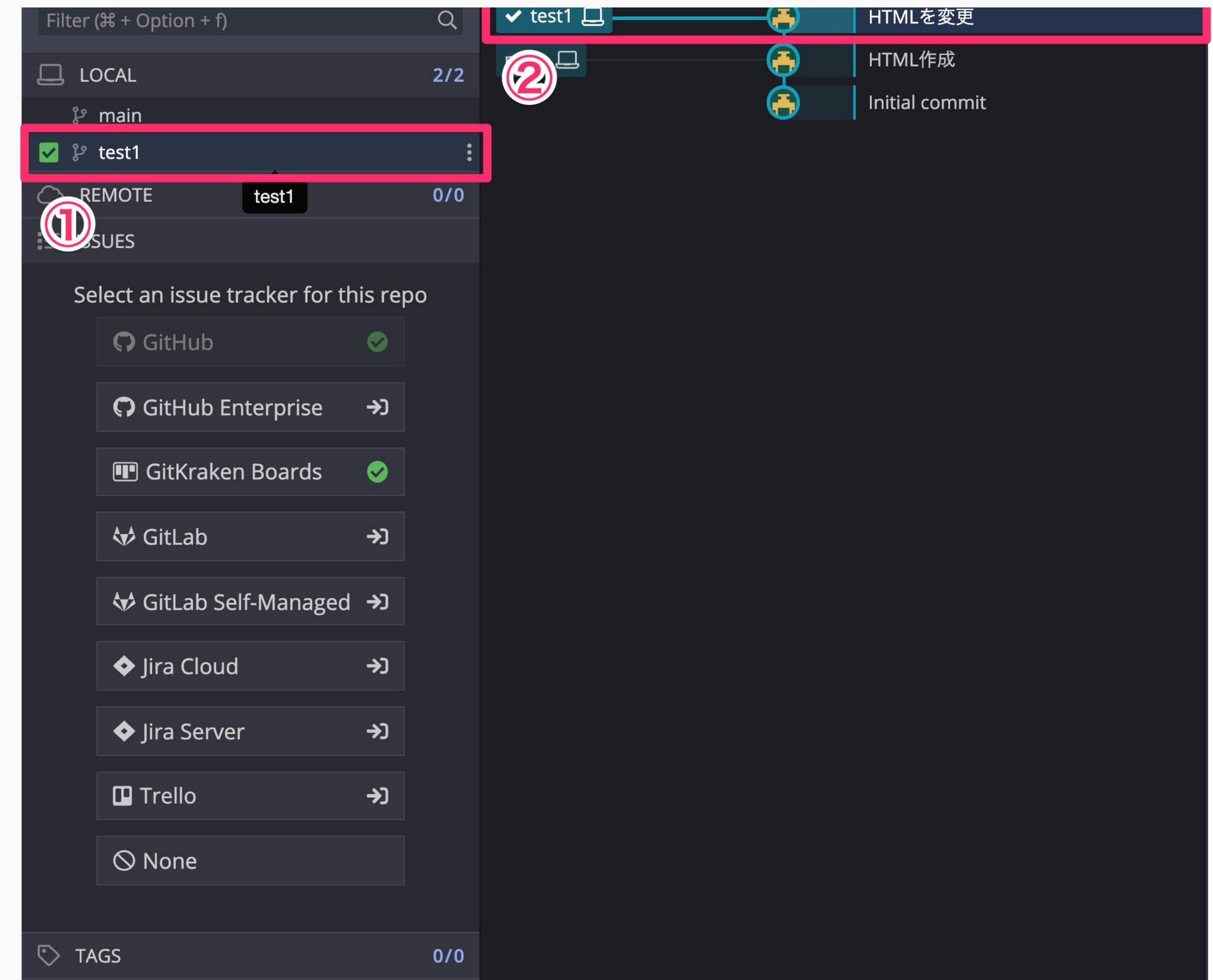
The commit message at the bottom of the editor is 'You, an hour ago • HTML作成'.

元に戻ってる！

はじめてのブランチ⑦

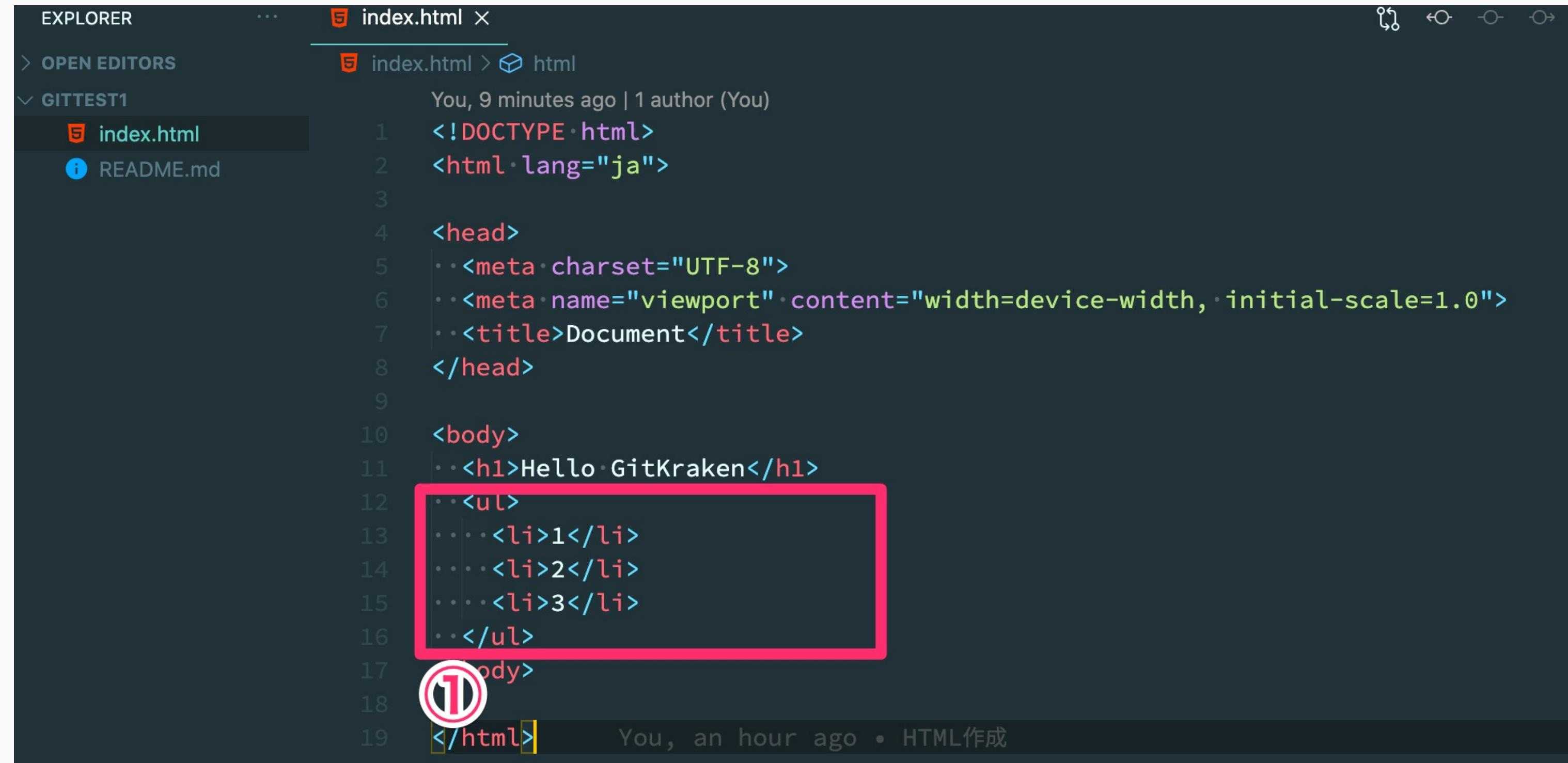
Practice

1. test1 ブランチを選択
2. ブランチが変わっているのがわかる



はじめてのブランチ⑧

Practice



The screenshot shows the GitKraken application interface. On the left is the Explorer sidebar with a tree view of files: 'OPEN EDITORS' and 'GITTEST1' (which is expanded) containing 'index.html' and 'README.md'. The main area is a code editor showing the content of 'index.html'. The code is as follows:

```
1 <!DOCTYPE html>
2 <html lang="ja">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9
10 <body>
11   <h1>Hello GitKraken</h1>
12   <ul>
13     <li>1</li>
14     <li>2</li>
15     <li>3</li>
16   </ul>
17 </body>
18
19 </html>
```

A red rectangular box highlights the list of numbers from line 12 to line 16. In the bottom right corner of the code editor, there is a circular icon with the number '11' and a red border.

復活してます！

はじめてのブランチ⑨

Practice

Main Branch

```
> OPEN EDITORS
GITTEST1
index.html > html
You, an hour ago | 1 author (You)
1 <!DOCTYPE html>
2 <html lang="ja">
3
4 <head>
5 ..<meta charset="UTF-8">
6 ..<meta name="viewport" content="width=device-width, initial-scale=1.0">
7 ..<title>Document</title>
8 </head>
9
10 <body>
11 ..<h1>Hello GitKraken</h1>
12 </body>
13
14 </html> You, an hour ago • HTML作成
```

index.html

Test1 Branch

```
> OPEN EDITORS
GITTEST1
index.html > html
You, 9 minutes ago | 1 author (You)
1 <!DOCTYPE html>
2 <html lang="ja">
3
4 <head>
5 ..<meta charset="UTF-8">
6 ..<meta name="viewport" content="width=device-width, initial-scale=1.0">
7 ..<title>Document</title>
8 </head>
9
10 <body>
11 ..<h1>Hello GitKraken</h1>
12 ..<ul>
13 ...<li>1</li>
14 ...<li>2</li>
15 ...<li>3</li>
16 ..</ul>
17 </body>
18
19 </html> You, 9 minutes ago • HTML作成
```

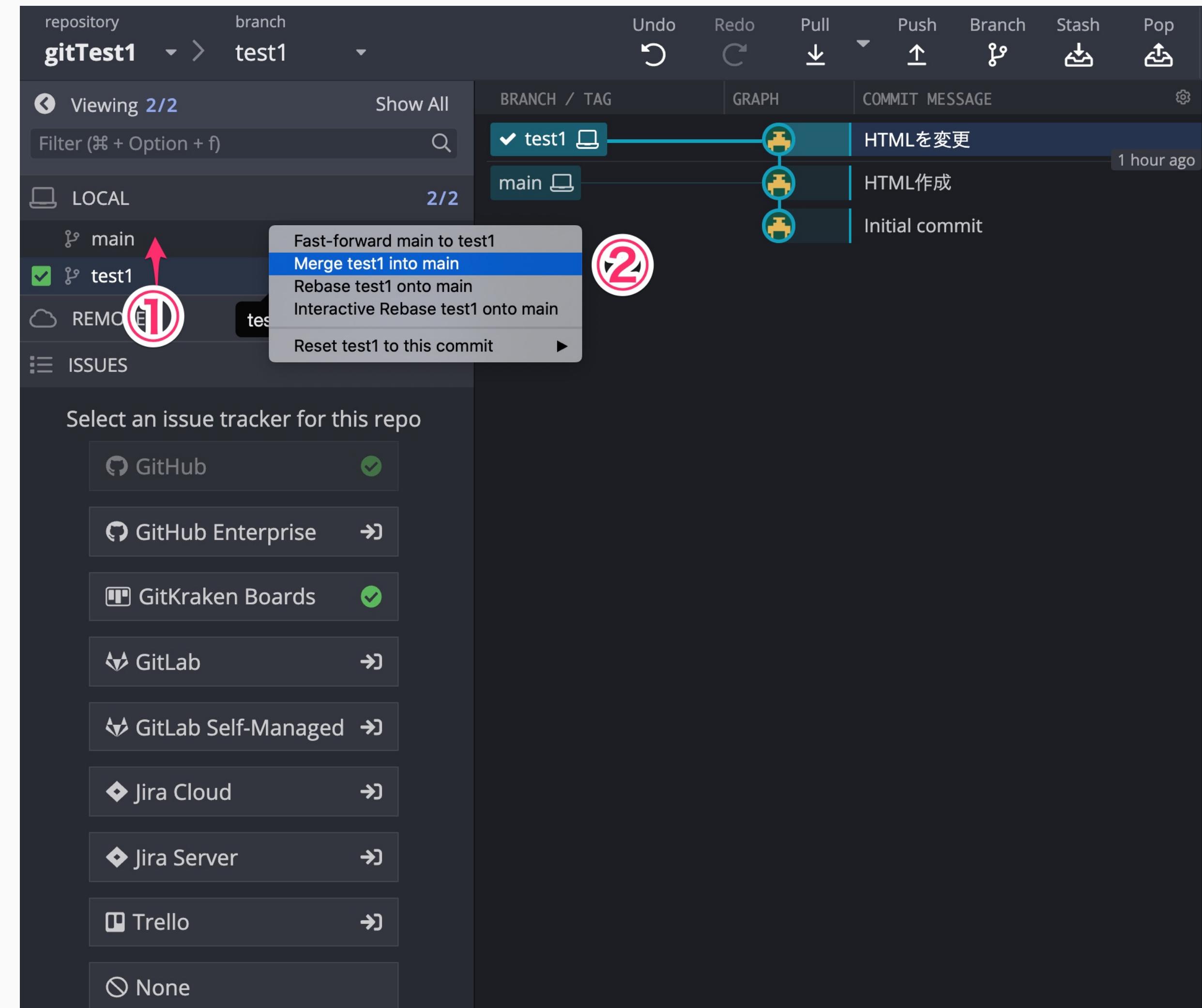
index.html

両方の状態が存在している

はじめてのマージ①

Practice

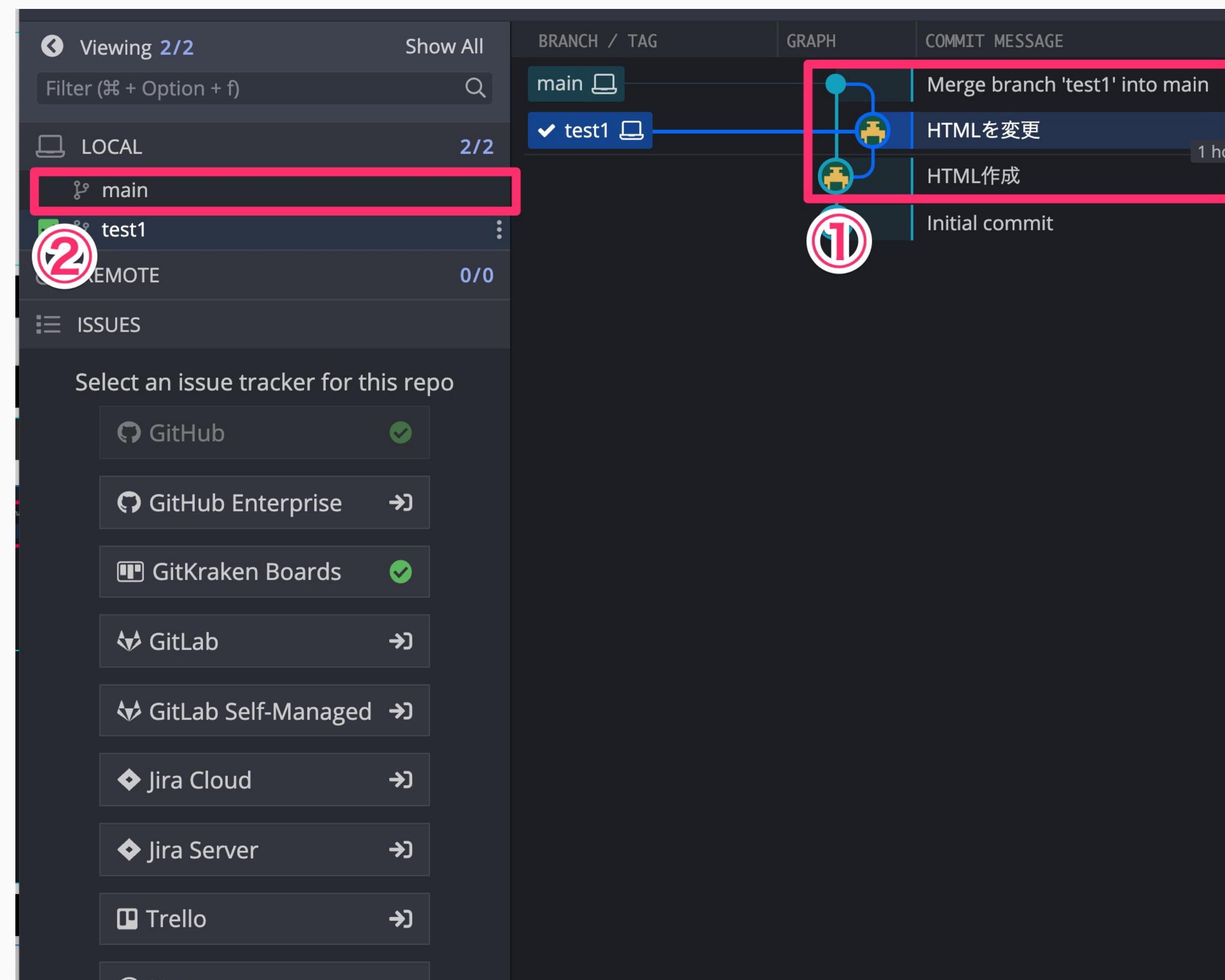
1. test1をmainにドラッグ&ドロップ
2. Merge test1 into mainをクリック



はじめてのマージ②

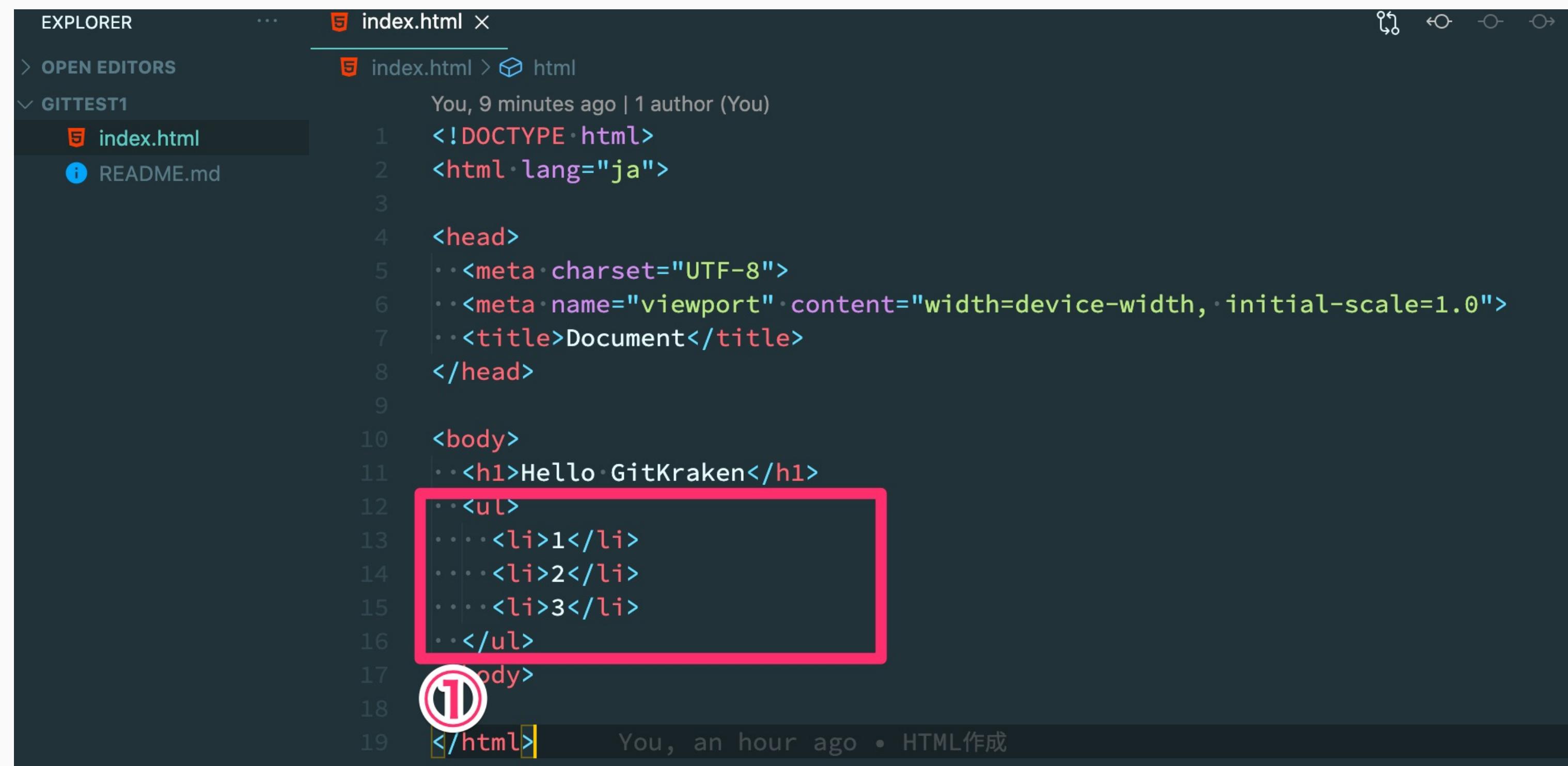
Practice

1. なんかニヨキッとなってる
2. mainブランチに切り替える



はじめてのマージ③

Practice



The screenshot shows the GitKraken interface with a merge conflict in the file `index.html`. The code editor displays the following HTML structure:

```
1 <!DOCTYPE html>
2 <html lang="ja">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9
10 <body>
11   <h1>Hello GitKraken</h1>
12   <ul>
13     <li>1</li>
14     <li>2</li>
15     <li>3</li>
16   </ul>
17 </body>
18
19 </html>
```

A red box highlights the list items from line 12 to 16. A circular icon with the number 11 is positioned next to the opening `` tag on line 12.

復活してる！

はじめてのマージ④

Practice

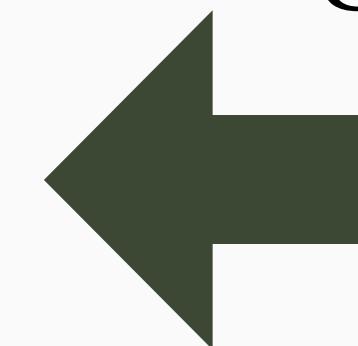
Main Branch



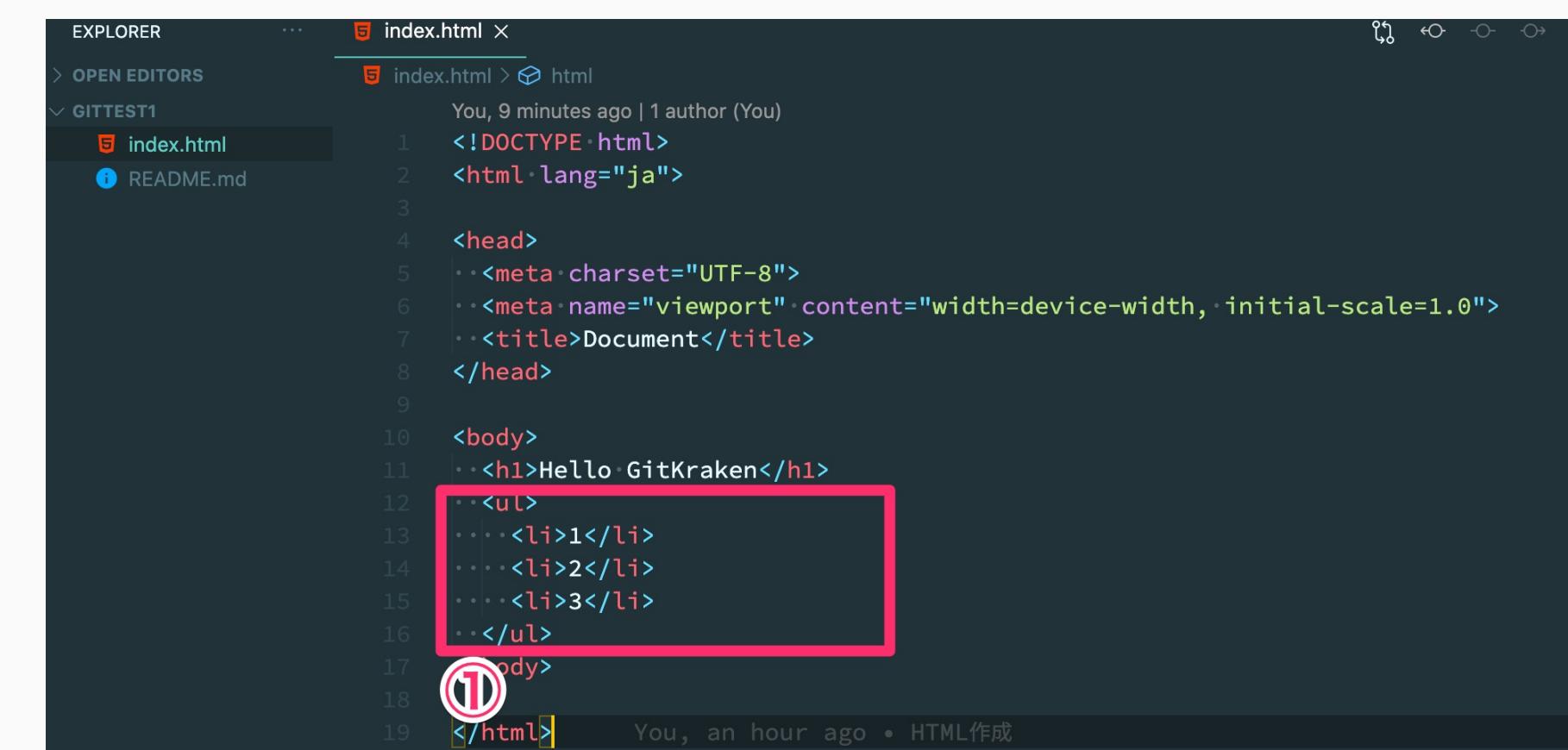
```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
  </head>
  <body>
    <h1>Hello GitKraken</h1>
  </body>
</html>
```

index.html

Merge



Test1 Branch



```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
  </head>
  <body>
    <h1>Hello GitKraken</h1>
    <ul>
      <li>1</li>
      <li>2</li>
      <li>3</li>
    </ul>
  </body>
</html>
```

index.html

並行世界が統合された

作業する時のポイント

Practice

Task

- ・ログインページ作成
- ・ログイン機能作成
- ・一覧ページ作成
- ・一覧ページ表示機能作成
- ・編集ページ作成
- ・編集機能作成

Branch

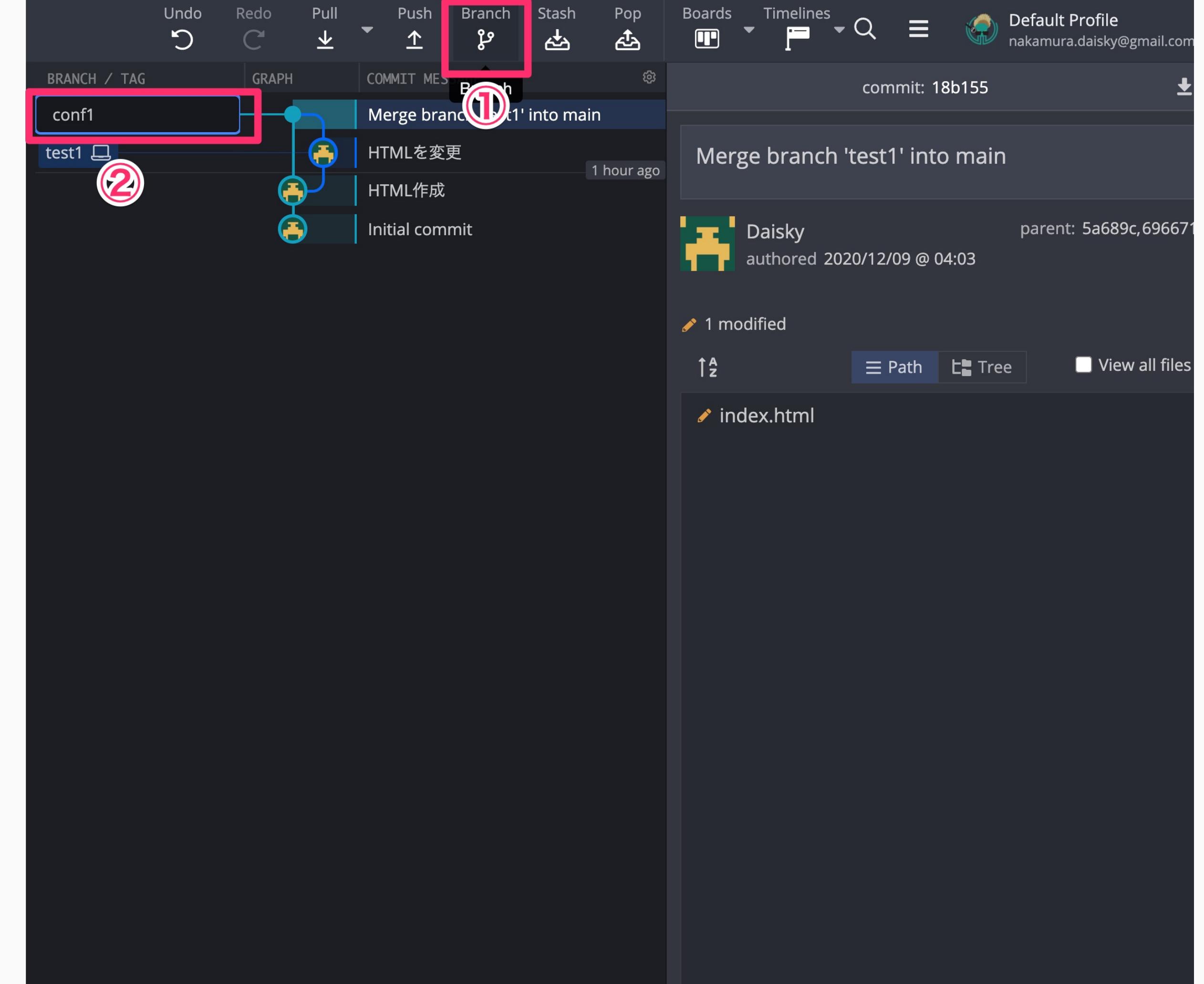
- ・loginPage
- ・loginAct
- ・selectPage
- ・selectAct
- ・editPage
- ・editAct

予め作業を洗い出して、作業毎のブランチを作る

コンフリクト①

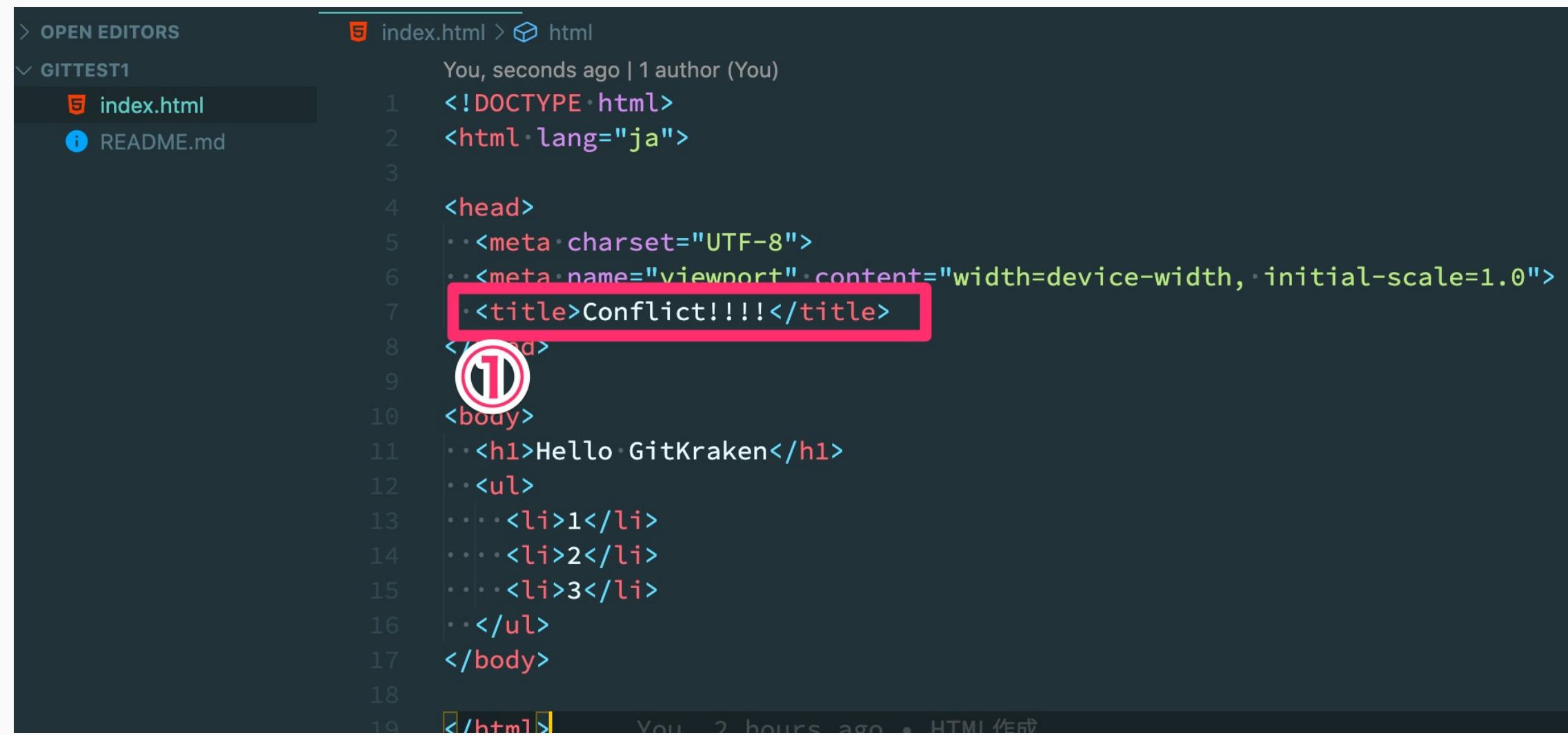
Practice

1. branchをクリック
2. conf1ブランチを作ります



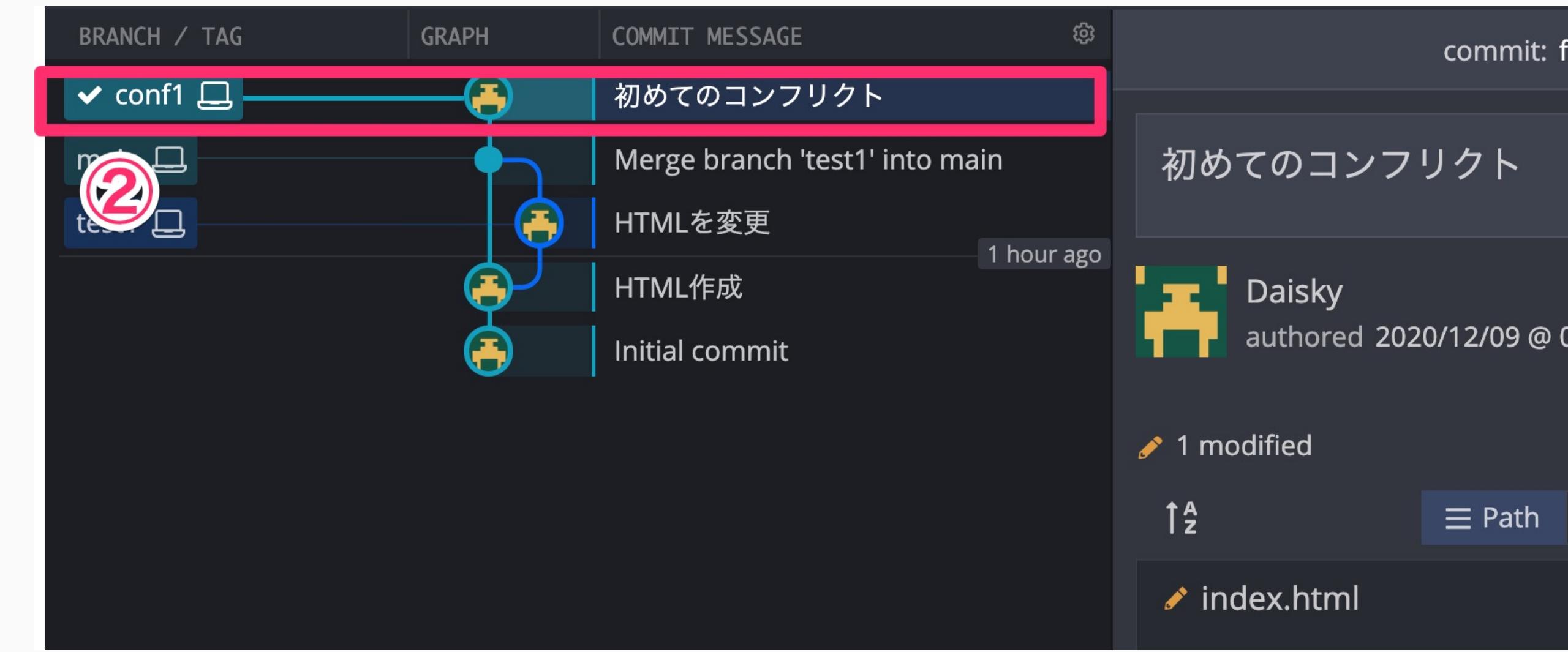
コンフリクト②

Practice



```
> OPEN EDITORS          index.html > html
  GITTEST1
    index.html
    README.md

index.html
You, seconds ago | 1 author (You)
1 <!DOCTYPE html>
2 <html lang="ja">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Conflict!!!!</title>
8 </head>
9 <body>
10  <h1>Hello GitKraken</h1>
11  <ul>
12    <li>1</li>
13    <li>2</li>
14    <li>3</li>
15  </ul>
16 </body>
17 </html>
You, 2 hours ago | HTML 作成
```

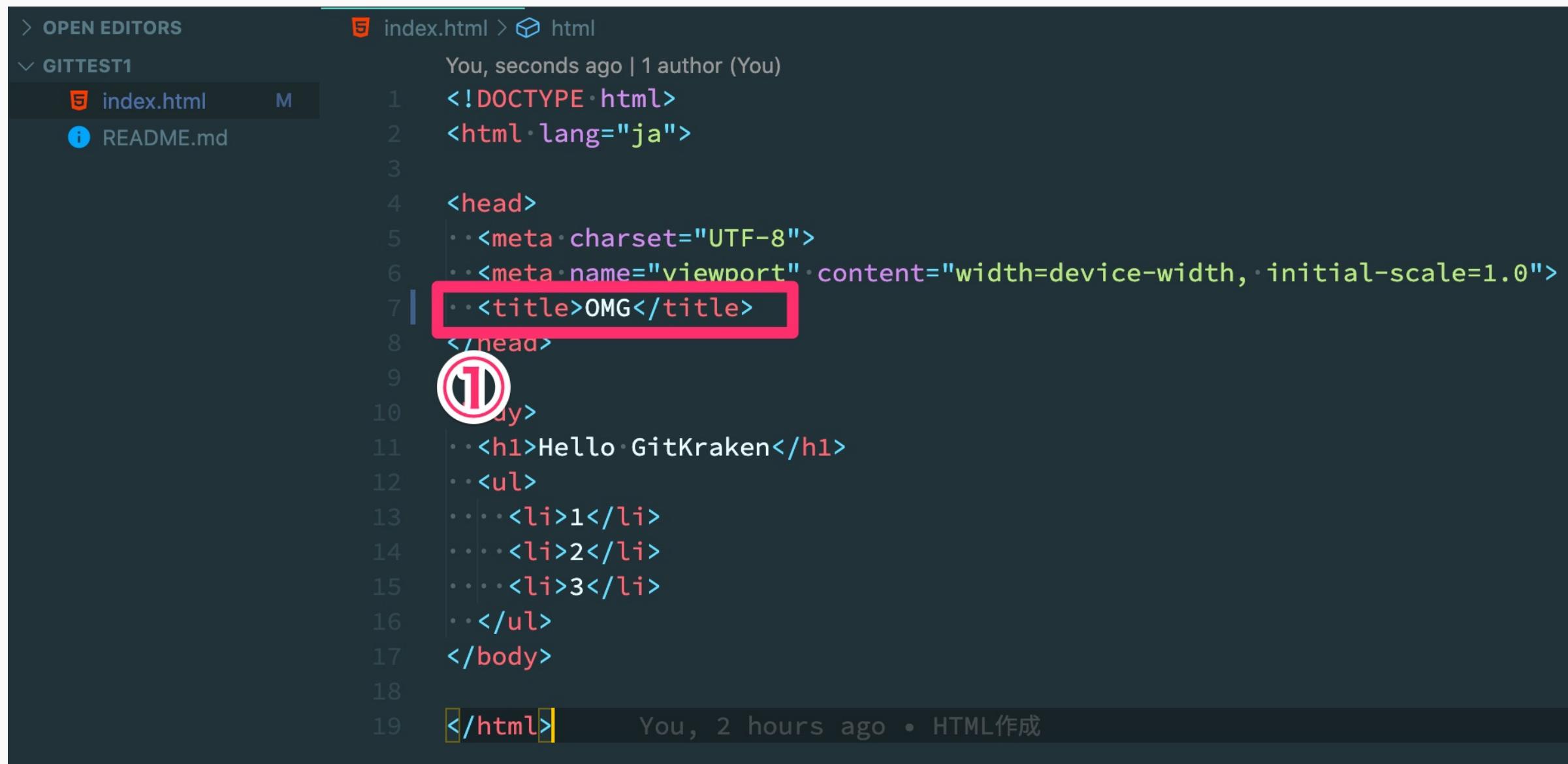


1. conf1 ブランチに切り替えてHTML変更

2. 修正内容をconf1にコミット

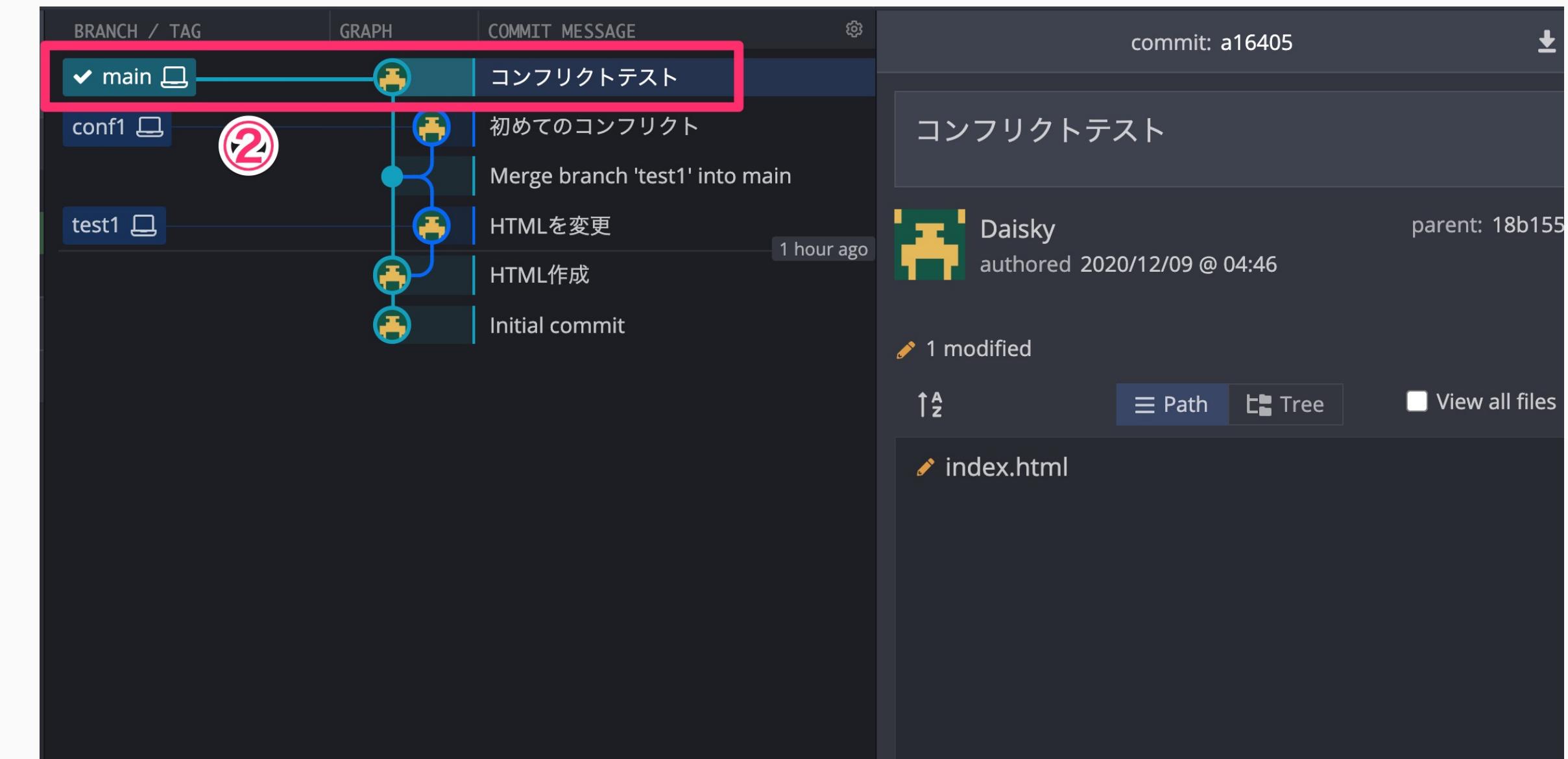
コンフリクト③

Practice



```
> OPEN EDITORS          index.html > html
GITTEST1
  index.html M
  README.md

index.html
You, seconds ago | 1 author (You)
1  <!DOCTYPE html>
2  <html lang="ja">
3
4  <head>
5    <meta charset="UTF-8">
6    <meta name="viewport" content="width=device-width, initial-scale=1.0">
7    <title>OMG</title> (1)
8  </head>
9  <body>
10   <h1>Hello GitKraken</h1>
11   <ul>
12     <li>1</li>
13     <li>2</li>
14     <li>3</li>
15   </ul>
16 </body>
17
18 </html>      You, 2 hours ago • HTML作成
```

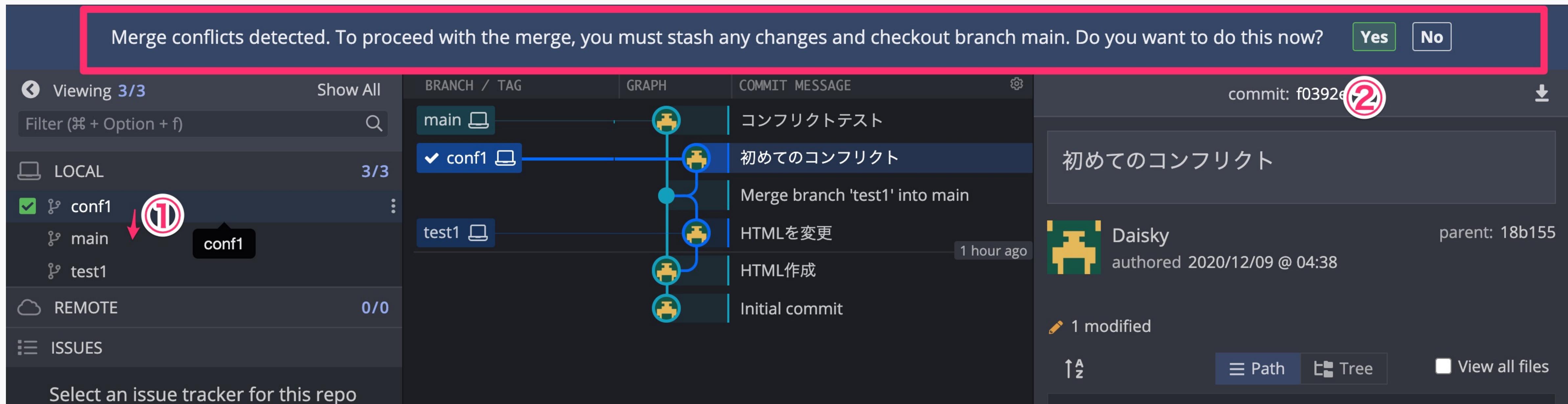


1. conf1をマージせずmainに切り替えて変更

2. mainの修正をコミット

コンフリクト④

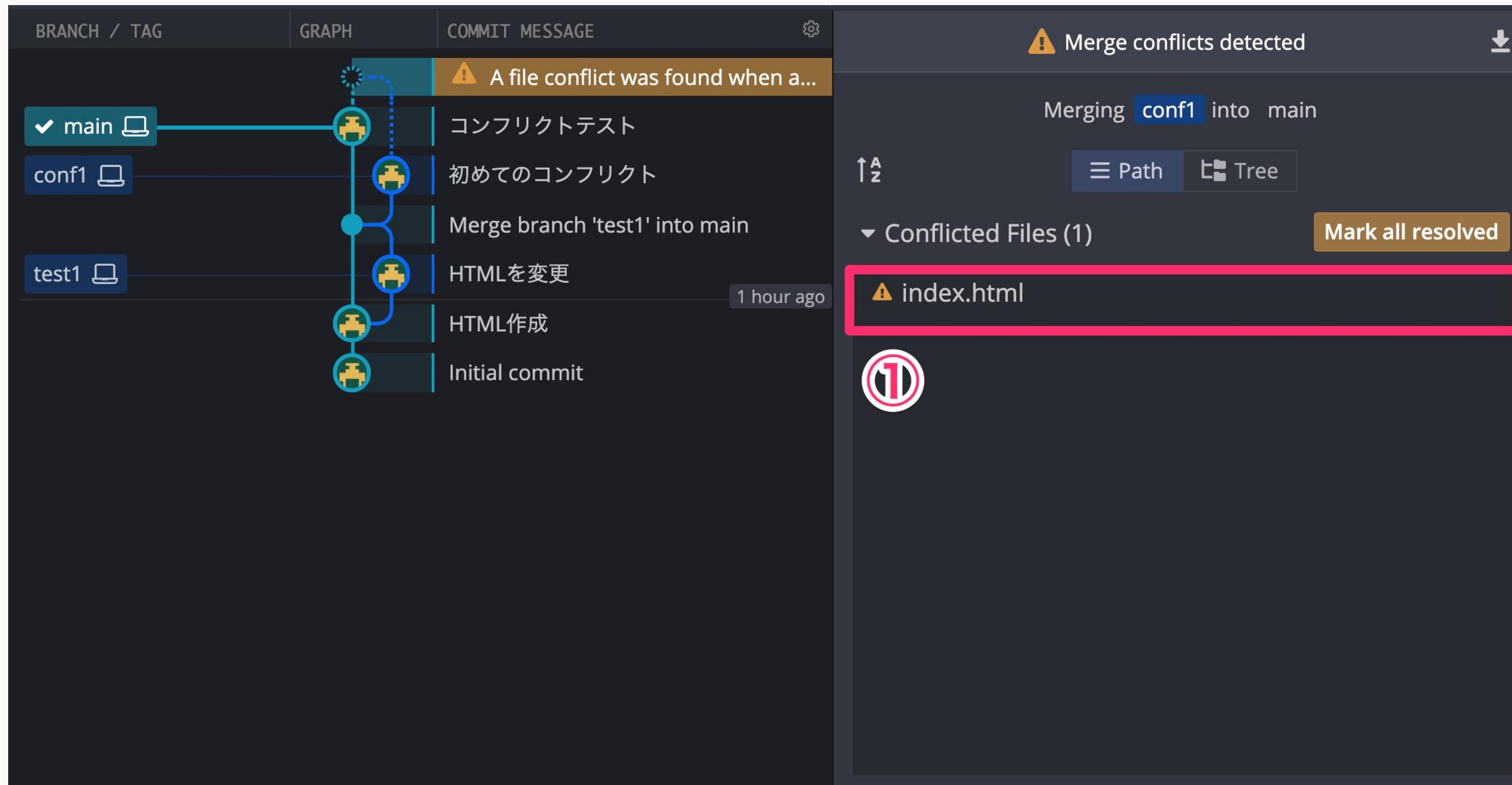
Practice



1. conf1をmainにマージ
2. コンフリクトの警告が出るのでYesをクリック

コンフリクト④

Practice



1. Conflict Filesの中のindex.htmlをクリック

コンフリクト⑤

Practice

1. mainブランチの内容
2. conf1ブランチの内容
3. 競合している場所
4. 最終的なアウトプット

The screenshot shows a merge conflict in the GitKraken interface. The left panel [A] shows the content of the 'main' branch, and the right panel [B] shows the content of the 'conf1' branch. A conflict is indicated by a yellow warning icon at the top left of the main panel.

Panel A (Commit a16405 on main):

```
1 <!DOCTYPE html>
2 <html lang="ja">
3 
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>OMG</title>
8 </head>
9 
10 <body>
11   <h1>Hello GitKraken</h1>
12   <ul>
13     <li>1</li>
14     <li>2</li>
15     <li>3</li>
16   </ul>
17 </body>
18 
19 </html>
```

Panel B (Commit f0392e on conf1):

```
1 <!DOCTYPE html>
2 <html lang="ja">
3 
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Conflict!!!!</title>
8 </head>
```

A conflict marker is shown in the title tag of the 'conf1' panel. The 'conflict' button in the top right of the main panel is highlighted with a red box and a circled number 1.

Output:

```
1 <!DOCTYPE html>
2 <html lang="ja">
3 
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Conflict!!!!</title>
8 </head>
```

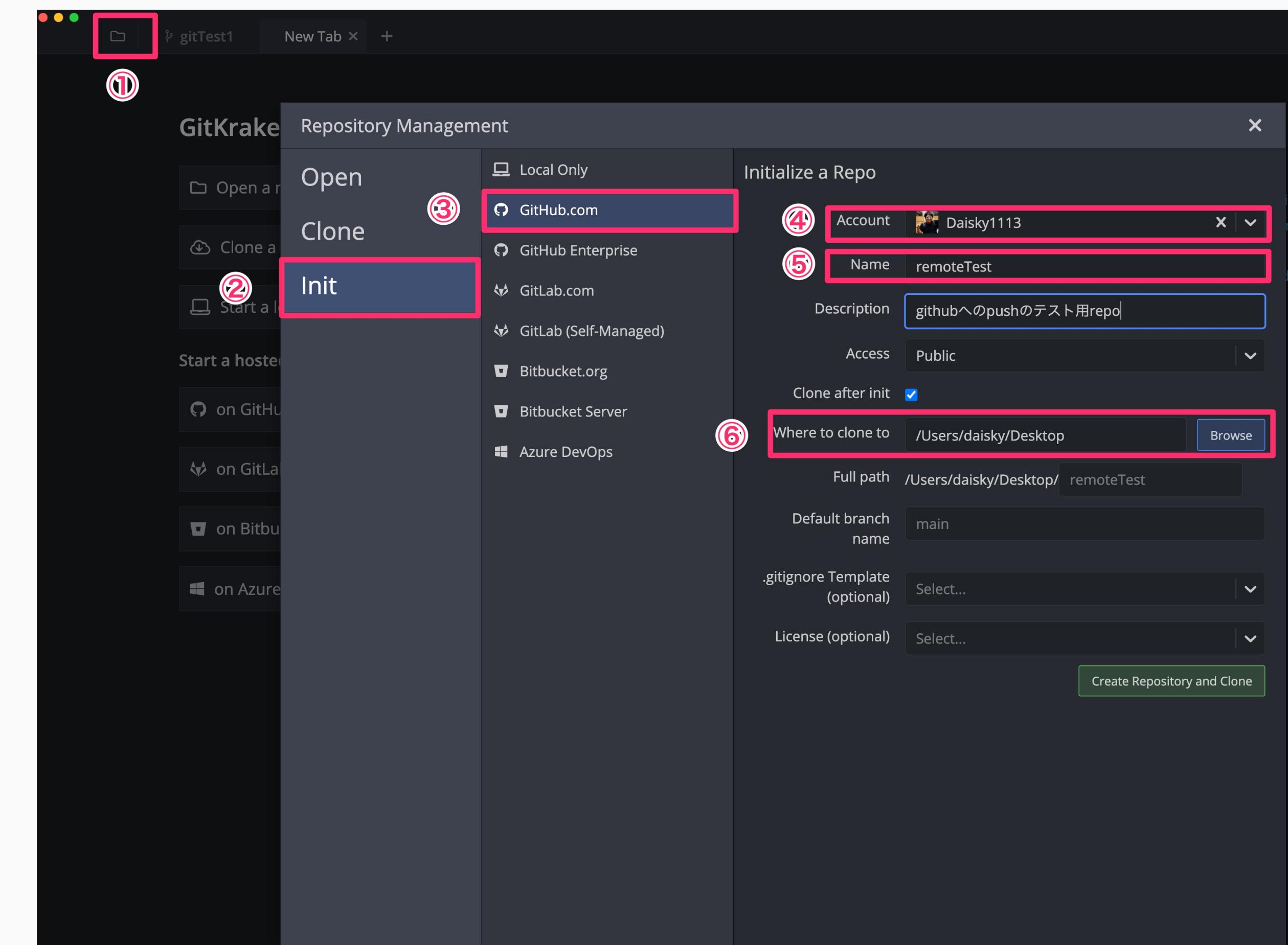
The title tag from the 'conf1' branch is highlighted with a red box and a circled number 4.

ポチポチしてればコンフリクト解消する

初めてのプッシュ①

Practice

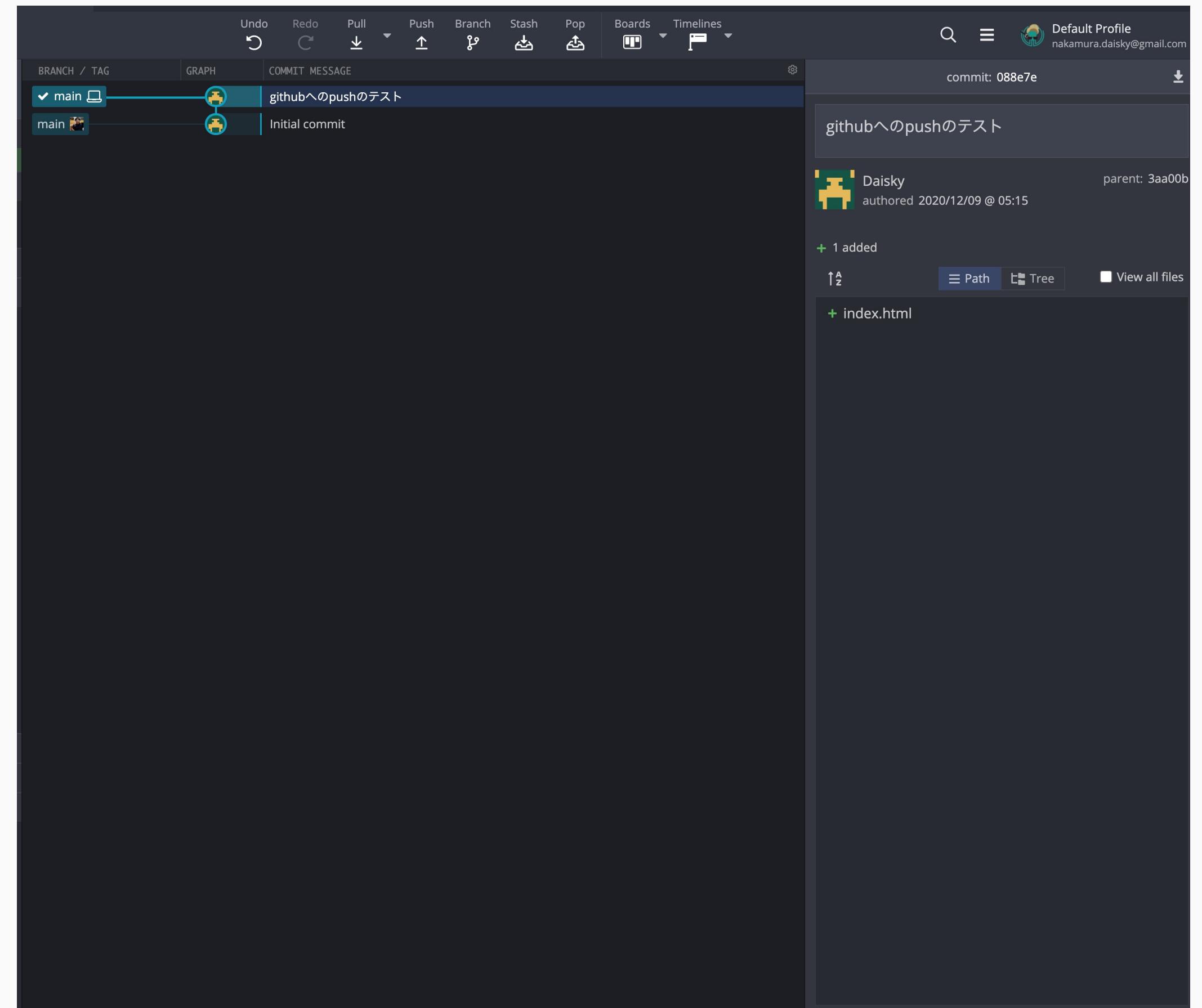
0. 新しいリポジトリを作る
1. 左上のフォルダをクリック
2. Initをクリック
3. Github .comをクリック
4. 自分のアカウントを選択する
5. リポジトリの名前を決める
6. 自分のパソコン上のどこに作るか決める



初めてのプッシュ②

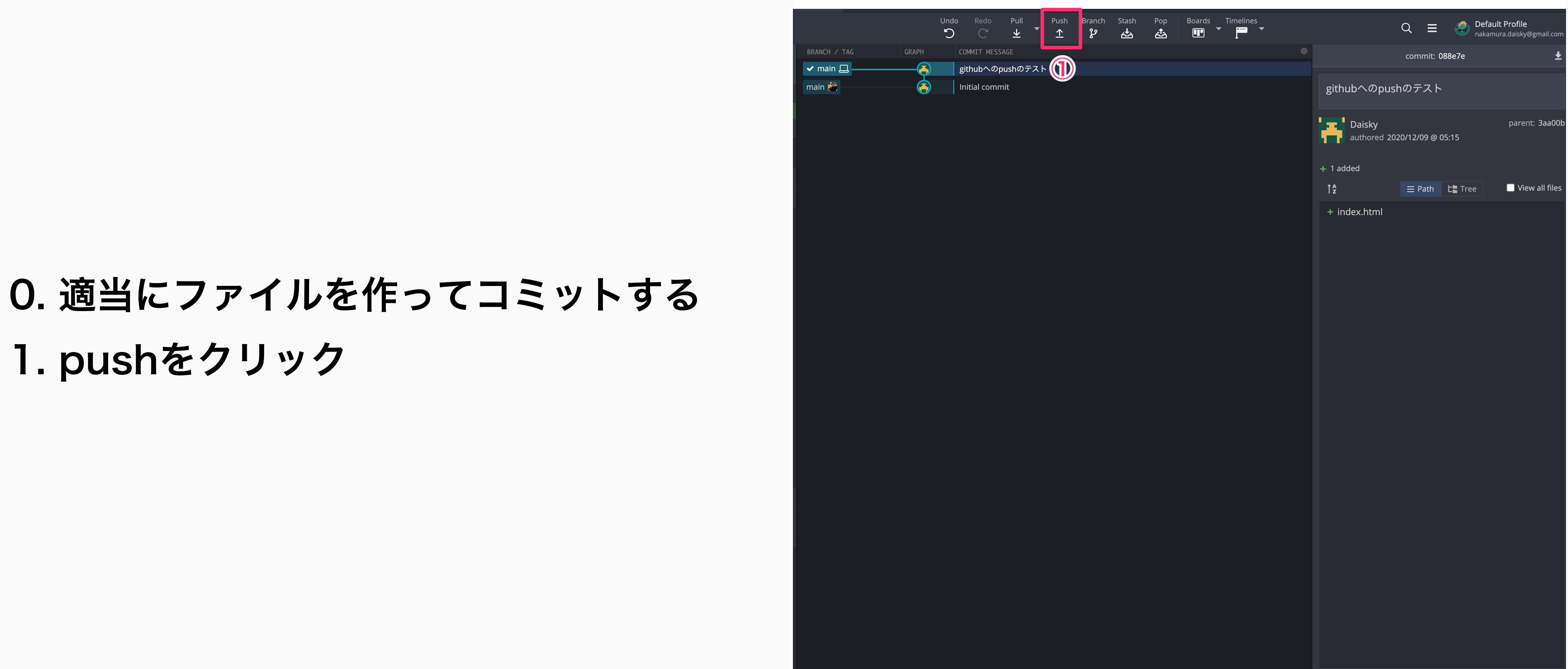
Practice

適当にファイルを作ってコミットする



初めてのプッシュ②

Practice



0. 適当にファイルを作ってコミットする

1. pushをクリック

初めてのプッシュ③

Practice

The screenshot shows a GitHub repository page for 'Daisky1113 / remoteTest'. The 'Code' tab is selected. A file named 'index.html' is displayed, showing the following code:

```
1 <!DOCTYPE html>
2 <html lang="ja">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9
10 <body>
11   <h1>Hello Github!</h1>
12 </body>
13
14 </html>
```

The commit message is 'githubへのpushのテスト' (Test push to GitHub). The commit was made by 'Daisky1113' 1 minute ago. There is 1 contributor.

ブラウザから確認する

ここまでまとめ

Practice

- ・ 基本的に何もかもボタンクリックすれば出来る
- ・ ブランチは作業毎に切る
- ・ コンフリクトは焦る必要なし

チームでGitを使う方法

Team Development with Github flow

Reader



Daisky

Member

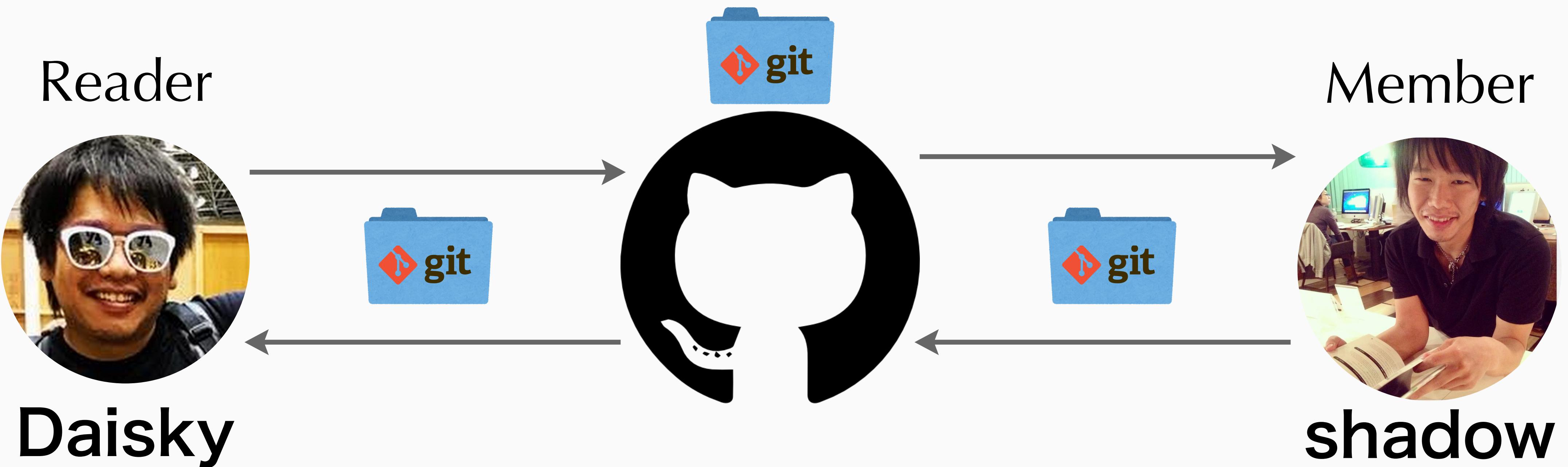


shadow

ここからは2人でお送ります

やりたいこと

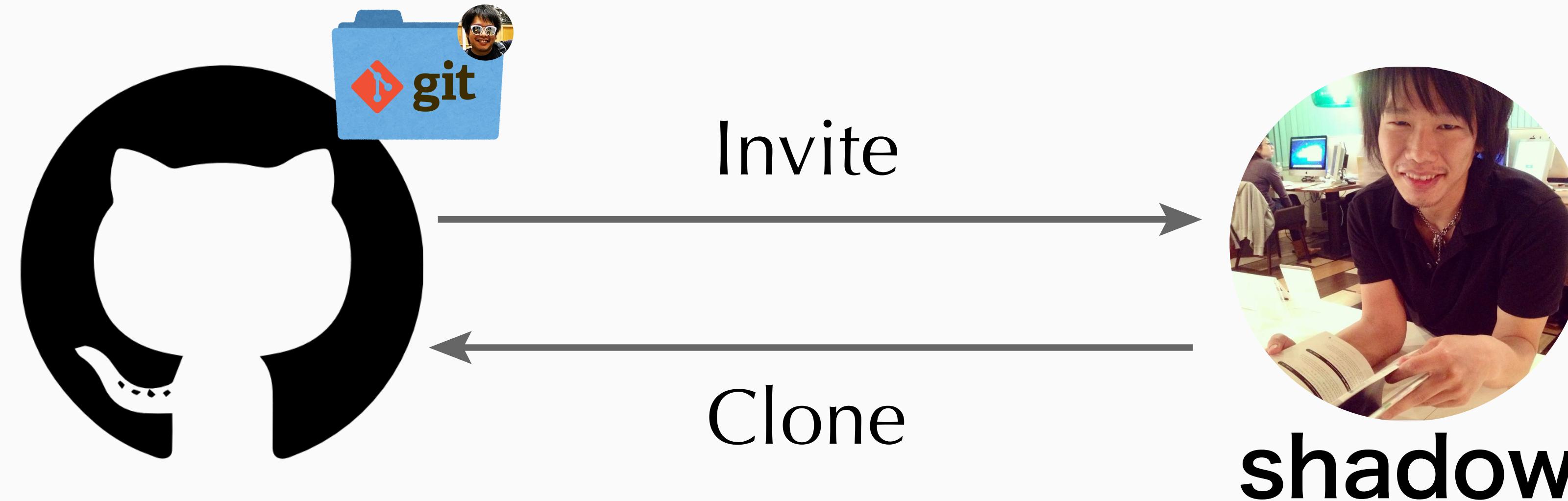
Study



Githubを通じてチーム開発したい

必要なこと

Study



リーダのリモートリポジトリに招待する

やってみよう

Practice

- ・リーダーのリモートリポジトリにメンバーを招待する
- ・メンバーはリーダーのリモートリポジトリをクローンしてローカルに持ってくる

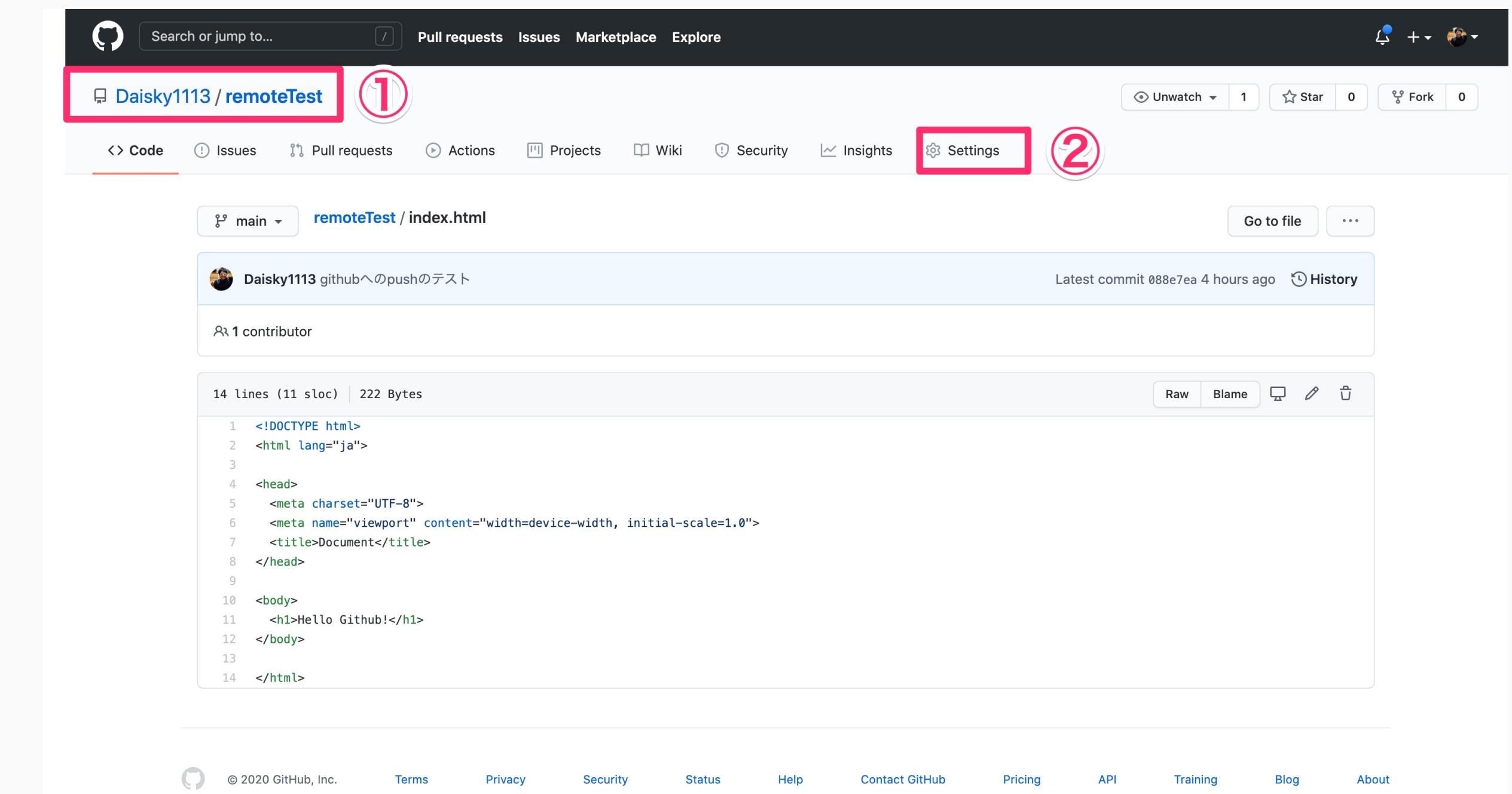
リモートリポジトリを作る

Practice



Reader

1. 共有したいリポジトリを確認
2. Settingsをクリック



The screenshot shows a GitHub repository page for 'Daisky1113 / remoteTest'. The 'Code' tab is selected. A red box highlights the 'Settings' link in the navigation bar. Another red box highlights the 'remoteTest / index.html' file in the main content area. The file content is as follows:

```
main - remoteTest / index.html
Daisky1113 githubへのpushのテスト
Latest commit 088e7ea 4 hours ago History
1 contributor
14 lines (11 sloc) | 222 Bytes
1 <!DOCTYPE html>
2 <html lang="ja">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9
10 <body>
11   <h1>Hello Github!</h1>
12 </body>
13
14 </html>
```

At the bottom of the page, there is a footer with links: © 2020 GitHub, Inc., Terms, Privacy, Security, Status, Help, Contact GitHub, Pricing, API, Training, Blog, About.

メンバーを招待する

Practice



Reader

1. Manage accessをクリック
2. Invite a coraboratorをクリック

The screenshot shows the GitHub repository settings page for 'Daisky1113 / remoteTest'. The 'Settings' tab is selected. On the left, a sidebar menu has 'Manage access' highlighted with a red box and circled with a red number 1. The main content area is titled 'Who has access' and shows two options: 'PUBLIC REPOSITORY' (selected) and 'DIRECT ACCESS'. Under 'PUBLIC REPOSITORY', it says 'This repository is public and visible to anyone.' and has a 'Manage' link. Under 'DIRECT ACCESS', it says '0 collaborators have access to this repository. Only you can contribute to this repository.' In the center, there's a section titled 'Manage access' with a red box and circled with a red number 2. It displays the message 'You haven't invited any collaborators yet' and a green 'Invite a collaborator' button.

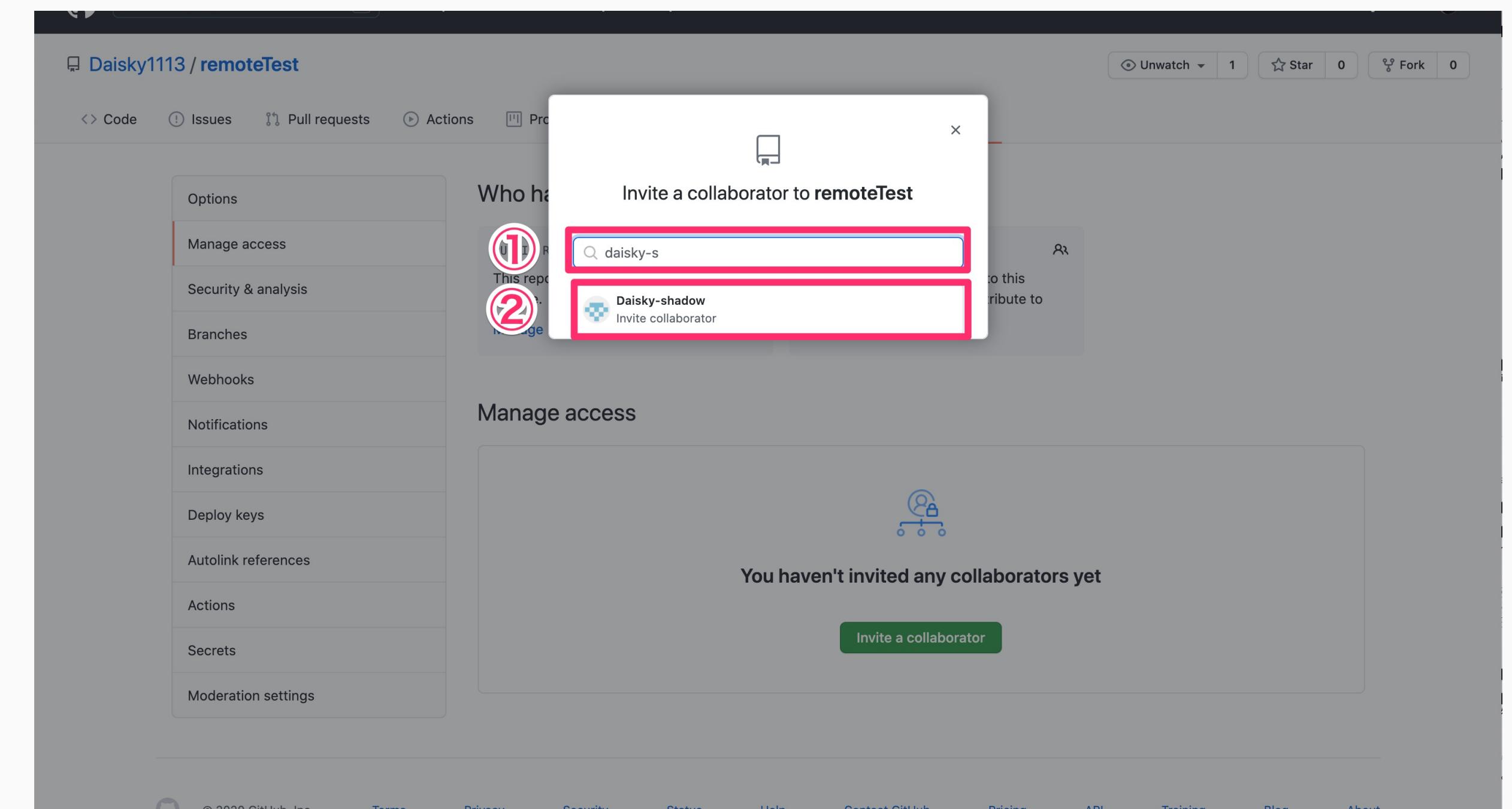
メンバーを招待する

Practice



Reader

1. アカウント名を入力
2. メンバーのアカウントを選択



The screenshot shows a GitHub repository page for 'Daisky1113 / remoteTest'. A modal window titled 'Invite a collaborator to remoteTest' is open. In the search bar, the text 'daisky-s' is entered. Below the search bar, a list of users includes 'Daisky-shadow' with the note 'Invite collaborator'. The sidebar on the left lists various repository settings: Options, Manage access (which is selected), Security & analysis, Branches, Webhooks, Notifications, Integrations, Deploy keys, Autolink references, Actions, Secrets, and Moderation settings. At the bottom of the modal, there is a green button labeled 'Invite a collaborator'.

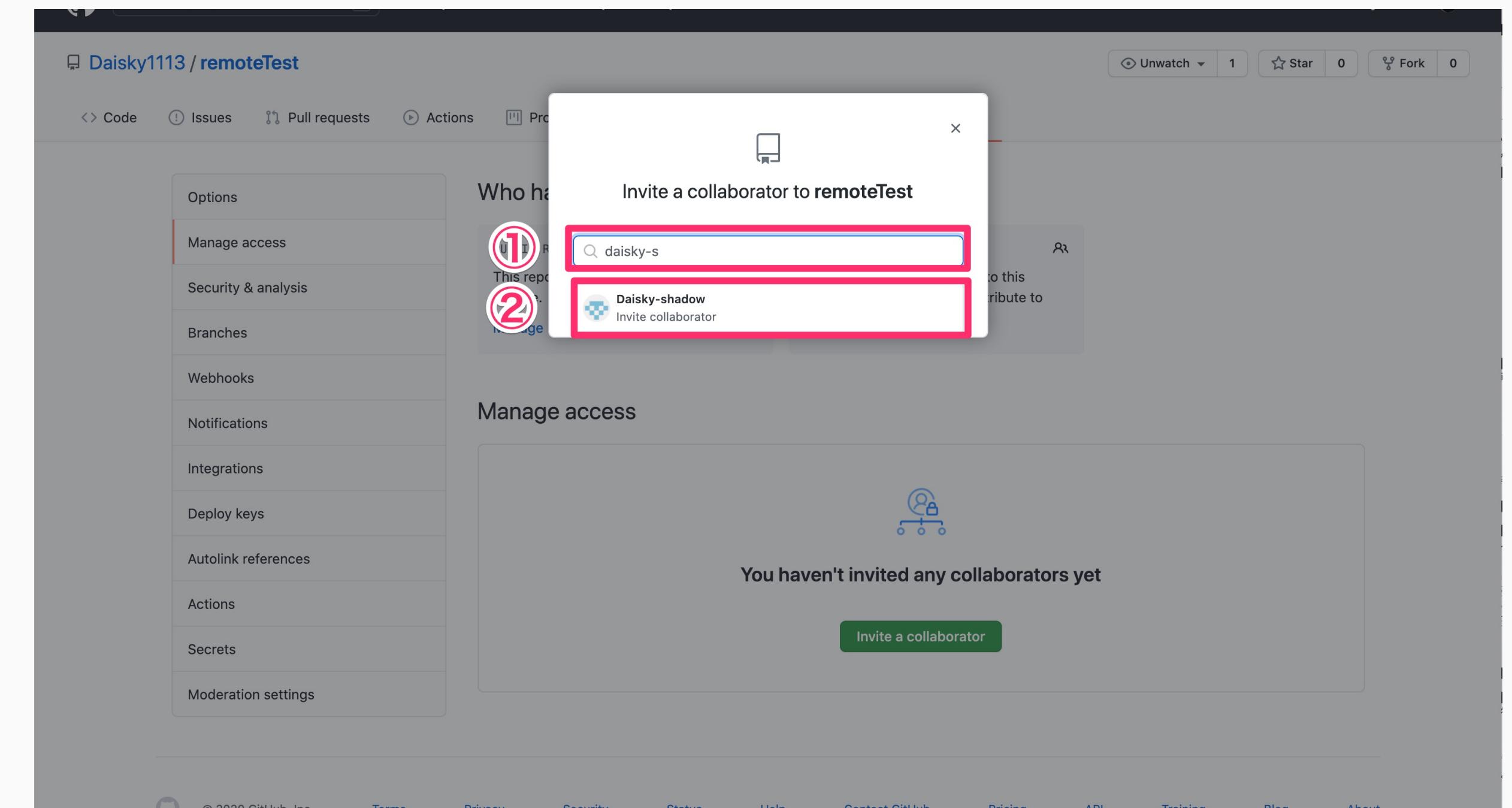
メンバーを招待する

Practice



Reader

1. アカウント名を入力
2. メンバーのアカウントを選択



The screenshot shows a GitHub repository page for 'Daisky1113 / remoteTest'. A modal window titled 'Invite a collaborator to remoteTest' is open. In the search bar, the text 'daisky-s' is entered. Below the search bar, a list of users includes 'Daisky-shadow' with the note 'Invite collaborator'. The sidebar on the left lists various repository settings: Options, Manage access (which is selected), Security & analysis, Branches, Webhooks, Notifications, Integrations, Deploy keys, Autolink references, Actions, Secrets, and Moderation settings. At the bottom of the modal, there is a green button labeled 'Invite a collaborator'.

確認

Practice



Reader

1. 招待したアカウントが表示される

The screenshot shows a GitHub repository settings page for 'Daisky1113 / remoteTest'. A success message at the top states 'Daisky-shadow has been added as a collaborator on the repository.' The 'Settings' tab is selected. On the left, a sidebar menu includes 'Options', 'Manage access', 'Security & analysis', 'Branches', 'Webhooks', 'Notifications', 'Integrations', 'Deploy keys', 'Autolink references', 'Actions', 'Secrets', and 'Moderation settings'. The 'Who has access' section indicates it's a 'PUBLIC REPOSITORY'. It shows 'DIRECT ACCESS' with 1 user invited. A red box highlights the 'Manage access' section where 'Daisky-shadow' is listed with the status 'Awaiting Daisky-shadow's response' and a 'Pending Invite' button.

これでリーダーの準備はOK

招待を受諾

Practice



Member

1. リーダーから招待メールが来る

The screenshot shows a Gmail inbox with 16 messages. The message from GitHub is highlighted. The subject of the email is "GitHub" and it reads: "@Daisky1113 has invited you to collaborate on the **Daisky1113/remoteTest** repository". Below the message, there is a note: "You can accept or decline this invitation. You can also head over to <https://github.com/Daisky1113/remoteTest> to check out the repository or visit [@Daisky1113](#) to learn a bit more about them." A "View invitation" button is visible at the bottom right of the message preview.

招待を受諾

Practice



Member

Daisky1113 / remoteTest

Watch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights

1. Accept Invitationをクリック

Daisky1113 invited you to collaborate

Accept invitation Decline

Owners of remoteTest will be able to see:

- Your public profile information
- Certain activity within this repository
- Country of request origin
- Your access level for this repository
- Your IP address

招待を受諾

Practice



Member

1. 招待されたリポジトリ内
2. Codeをクリック

You now have push access to the Daisky1113/remoteTest repository.

Daisky1113 / remoteTest 1

Watch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights

main Go to file Add file Code 2

Daisky1113 githubへのpushのテスト ... 4 hours ago 2

README.md Initial commit 4 hours ago

index.html githubへのpushのテスト 4 hours ago

README.md

remoteTest

Releases No releases published Create a new release

Packages

githubへのpushのテスト用 repo

Readme

クローンする

Practice



Member

You now have push access to the Daisky1113/remoteTest repository.

Daisky1113 / remoteTest

Watch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights

main Go to file Add file Code

Daisky1113 githubへのpushのテスト ... README.md index.html README.md remoteTest

Clone HTTPS SSH GitHub CLI New https://github.com/Daisky1113/remoteTest 1 Use Git or checkout with SVN using the web URL.

About

githubへのpushのテスト用 repo

Readme

Releases

No releases published Create a new release

Packages

1

1. URLをコピーする

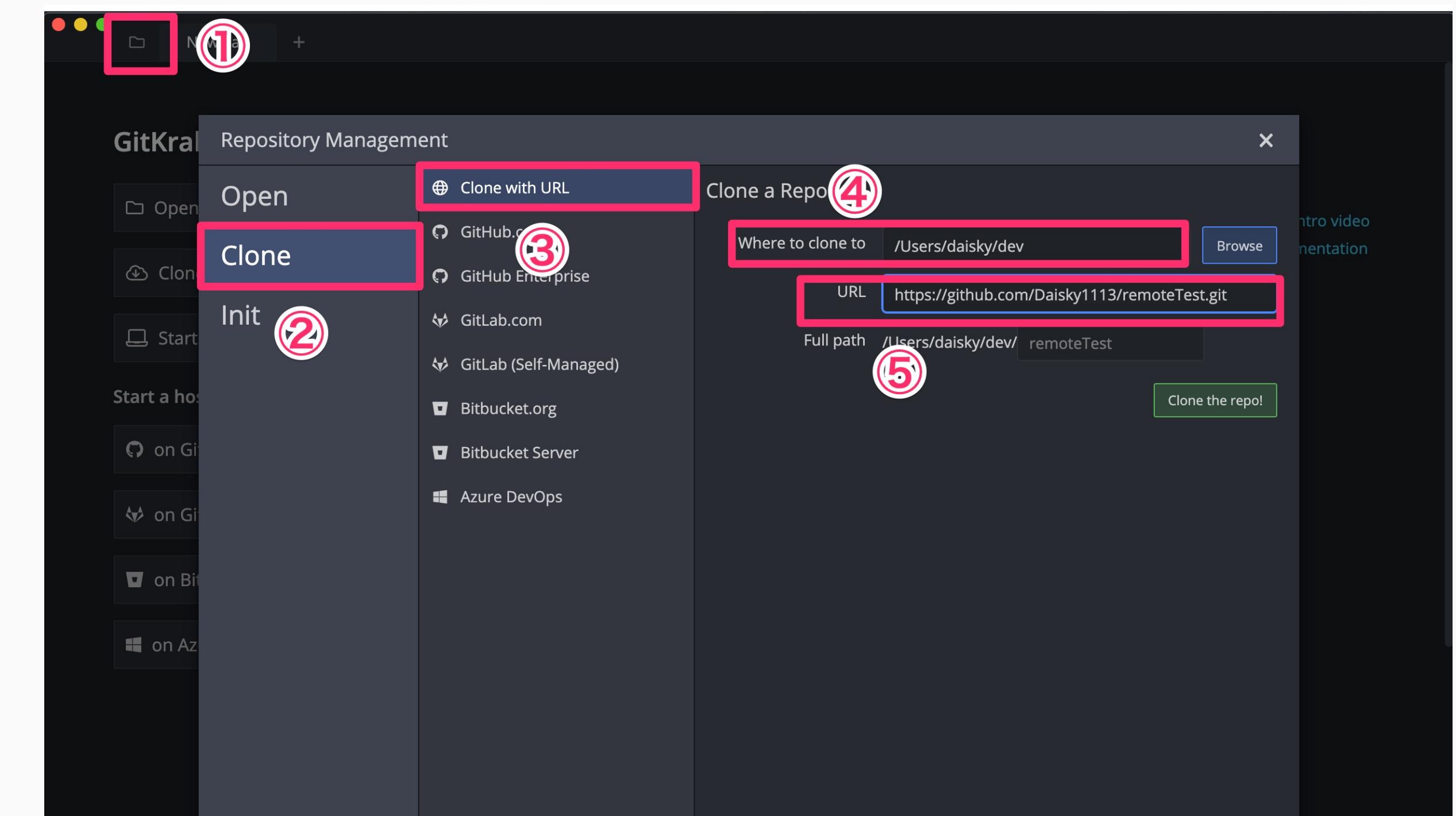
GitKrakenでクローン

Practice



Member

1. フォルダボタンをクリック
2. Cloneをクリック
3. Clone with URLをクリック
4. クローンする場所を指定
5. URLを貼り付ける



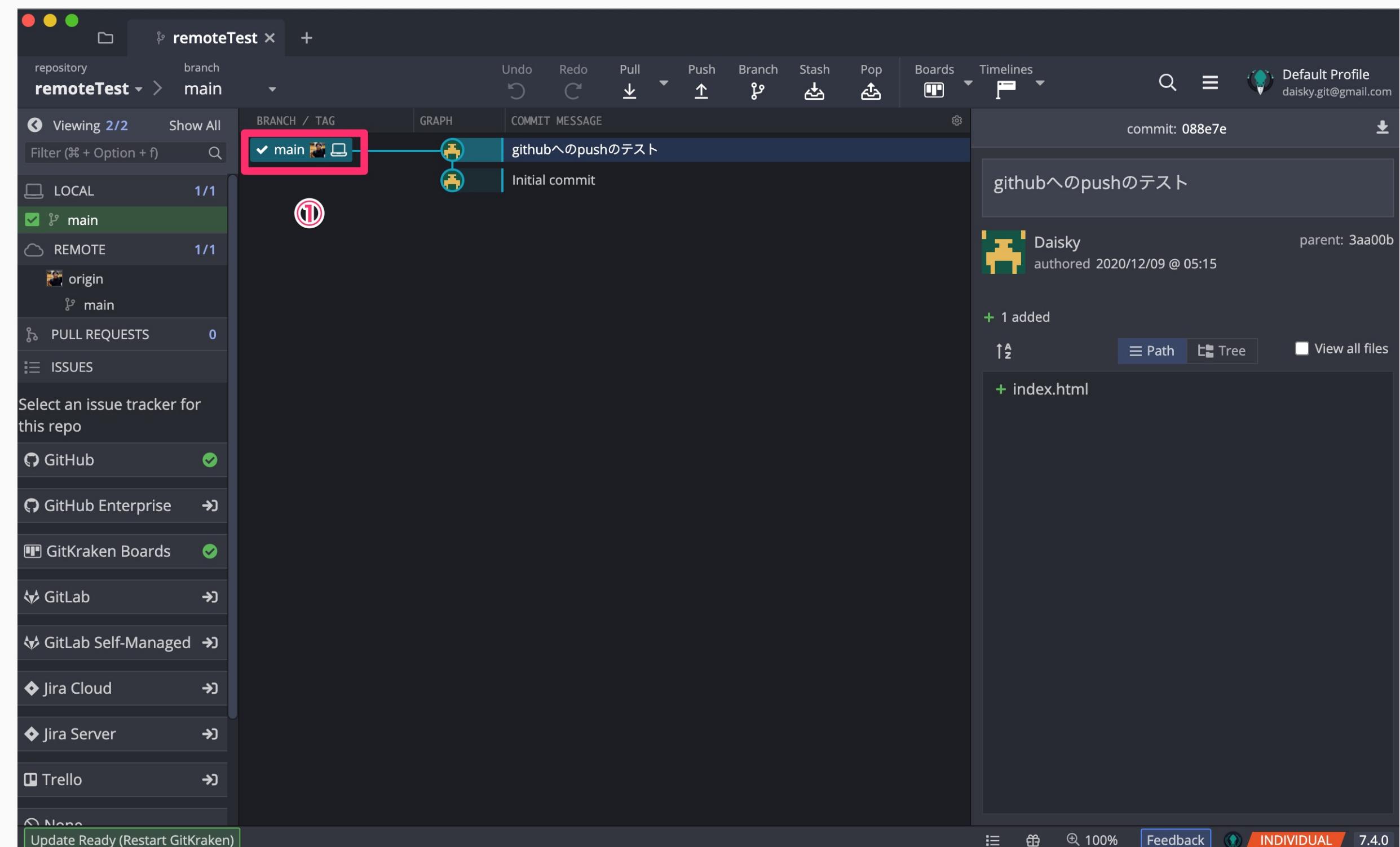
クローンする

Practice



Member

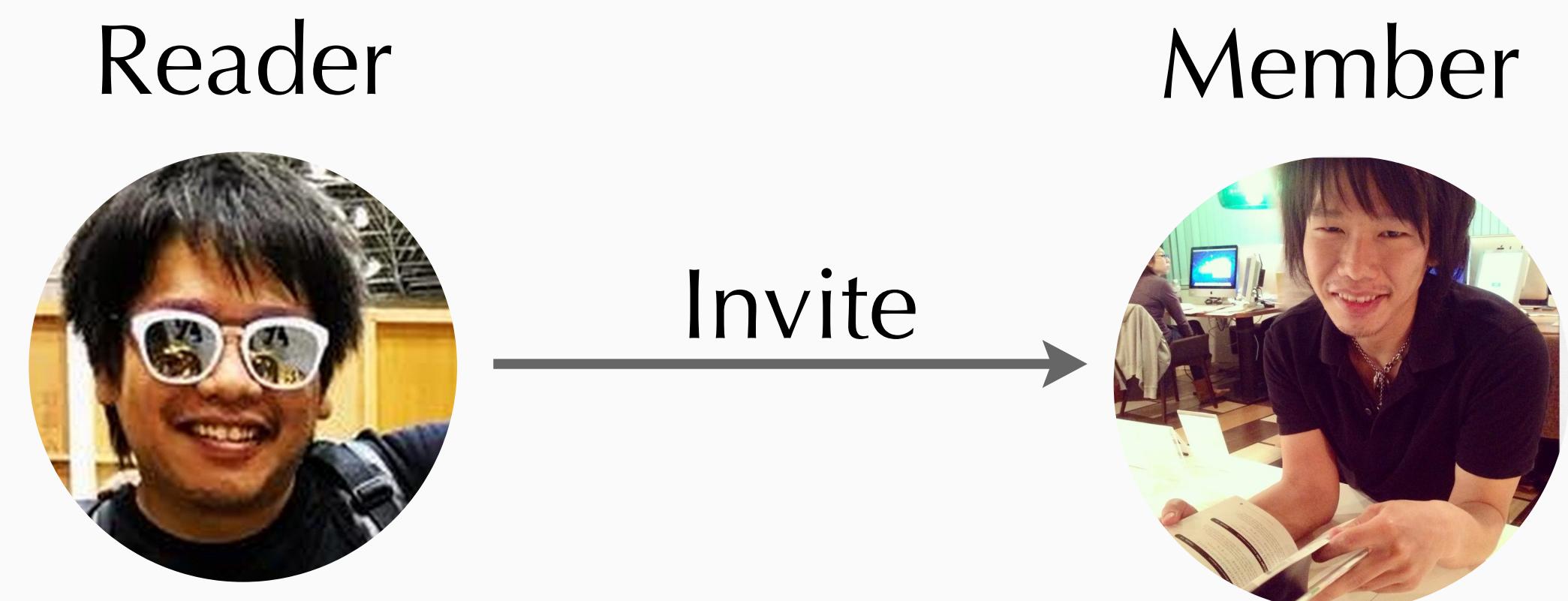
1. リーダーが作ったリポジトリがきてる



これでメンバーの準備もOK

ここまでまとめ

Practice

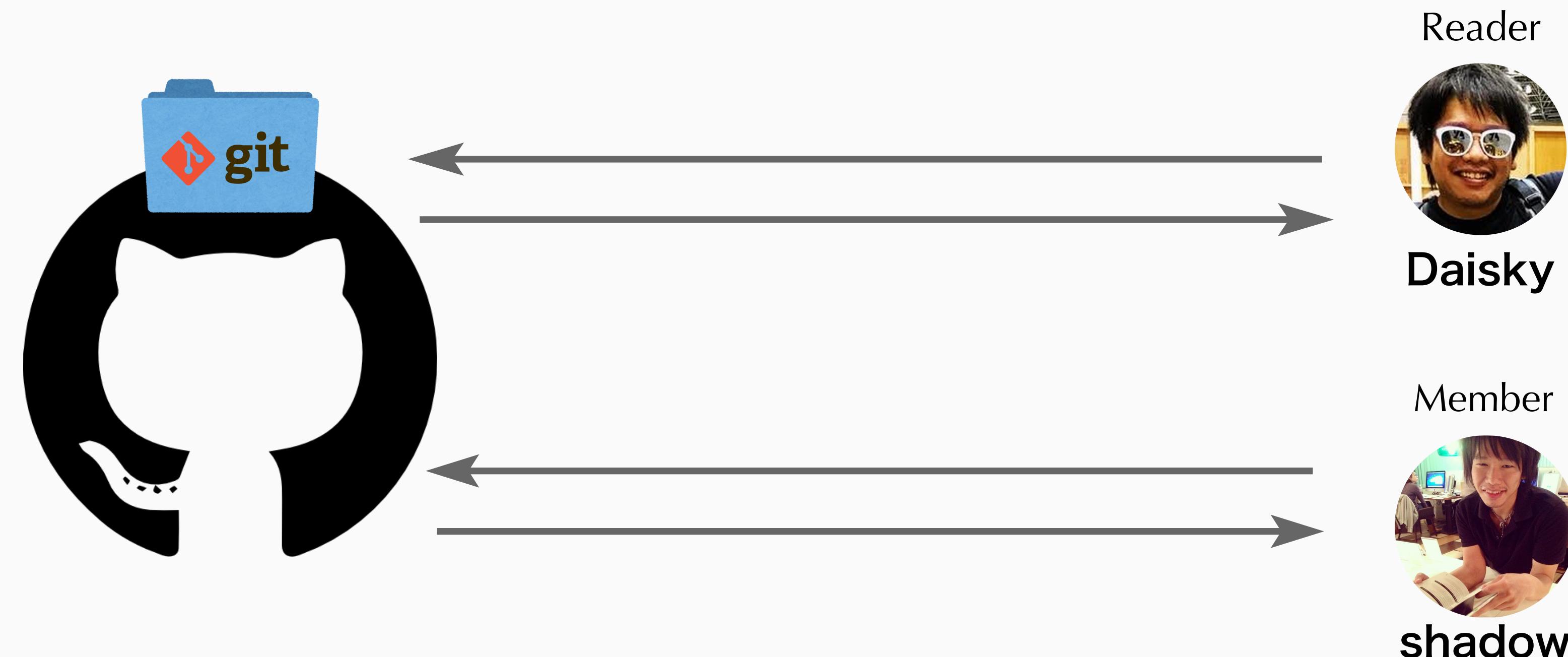


- ・リーダーは自分のリモートリポジトリにメンバーを招待する

次は実際に作業してみます

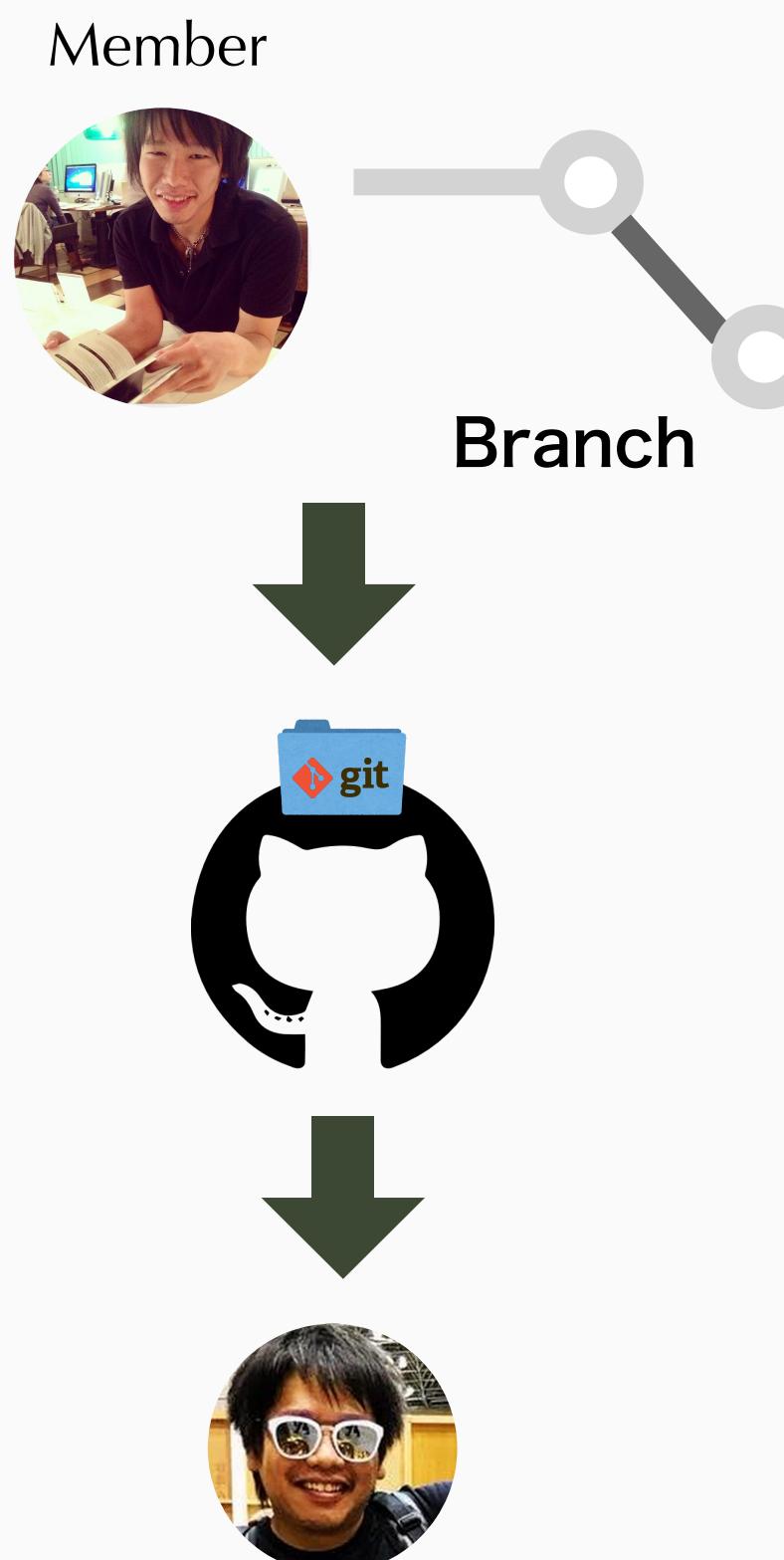
やりたいこと

Study



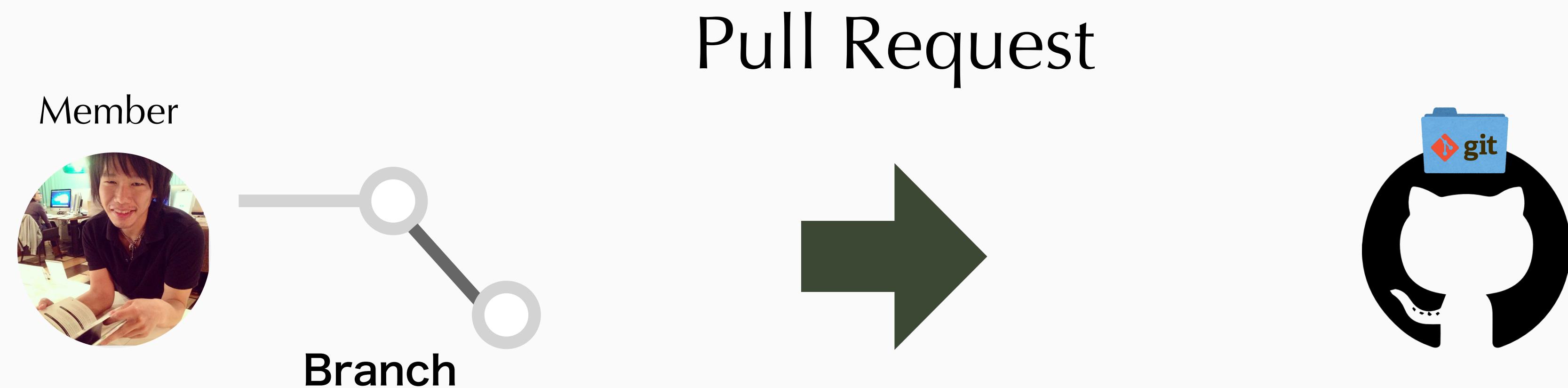
ひとつのリポジトリを複数人で扱う

作業の流れ



- ・ メンバーがブランチをリモートに送る
- ・ リモートでマージされる
- ・ 他のメンバーは修正を自分のローカルに取り込む

細かい流れ①



作業者はまずプルリクを送る

細かい流れ②

Study



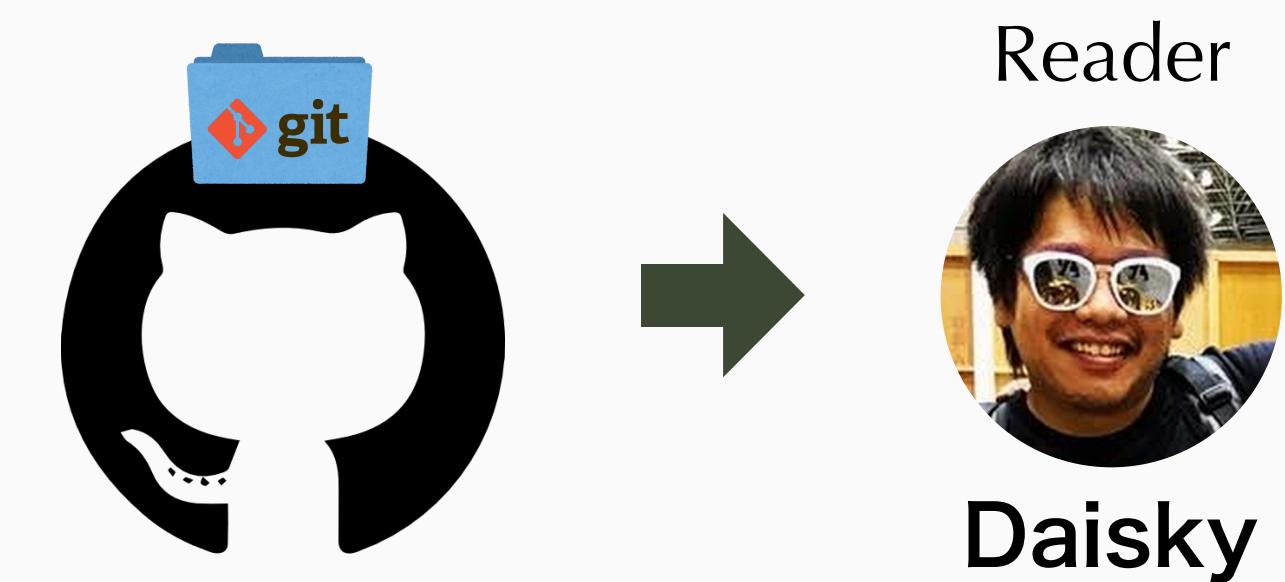
Reader



Daisky

リーダーがプルリクをマージする

細かい流れ③



他のメンバーはpullする

重要な事

チームでやる時はmainに直接pushしない

やってみよう

Practice

- ・メンバーは~~プルリク~~^{プルリク}を送る
- ・リーダーはマージする
- ・各自masterをpullしてくる

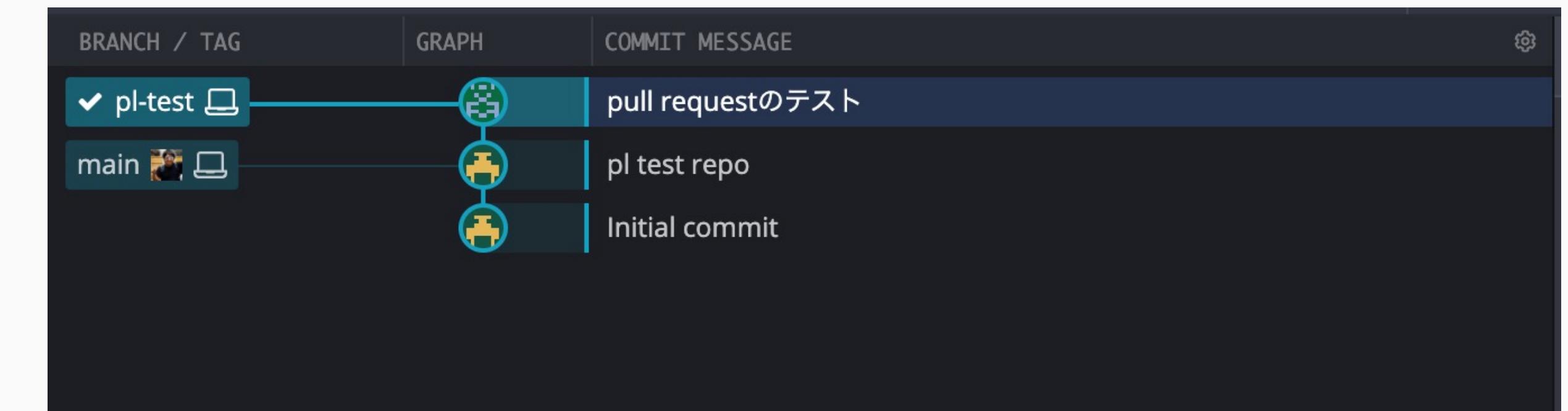
プルリク用のブランチを作る

Practice



Member

1. 共有したいリポジトリを確認
2. Settingsをクリック



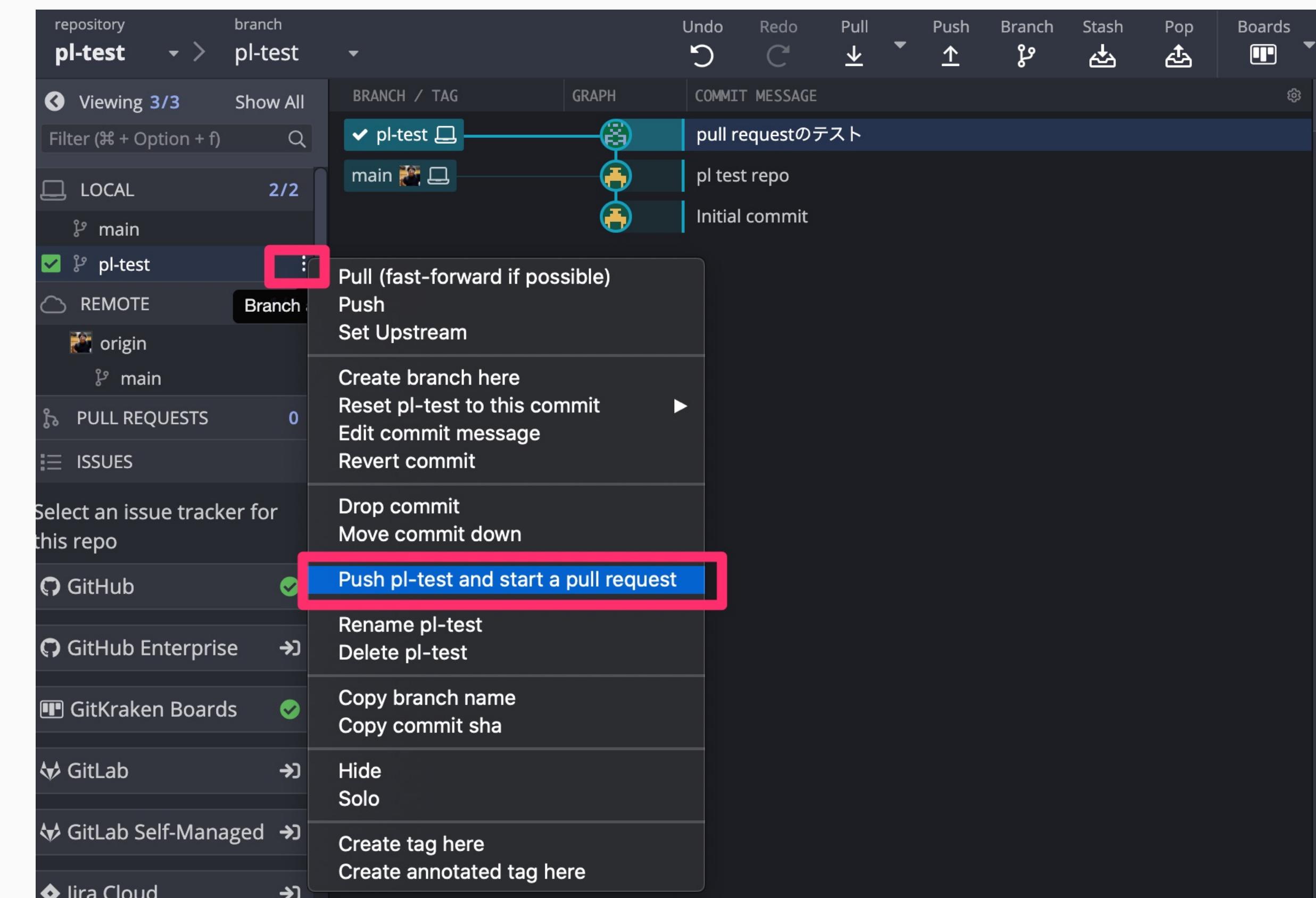
プルリクの準備をする

Practice



Member

1. ブランチの横のドットをクリック
2. start pull requestをクリック



プルリクの準備をする

Practice



Member

1. 送り元
2. 宛先

下の方にボタンがあるので送信

The screenshot shows a GitHub interface for creating a pull request. On the left, there's a sidebar with repository details: LOCAL (main, pl-test checked), REMOTE (origin, main, pl-test), PULL REQUESTS (0), and ISSUES. Below the sidebar is a list of integration options: GitHub (selected), GitHub Enterprise, GitKraken Boards, GitLab, GitLab Self-Managed, Jira Cloud, and Jira Server. The main area is titled 'Create Pull Request' for GitHub. It has two dropdowns: 'From Repo' set to 'Daisky1113/pl-test' and 'Branch' set to 'pl-test'; and 'To Repo' set to 'Daisky1113/pl-test' and 'Branch' set to 'main'. There are also fields for 'Title' (containing 'pull requestのテスト') and 'Description' (containing 'Pull request description'). At the bottom, there are sections for 'Reviewers' and 'Assignees' with 'Add reviewers...' and 'Add assignees...' buttons.

プルリクを確認する

Practice



Reader

A screenshot of a GitHub repository page for 'Daisky1113 / pl-test'. The page shows a single pull request from 'Daisky1113' with the title 'pl test repo'. The pull request was created 8 minutes ago and has 2 approvals. The repository contains files 'README.md' and 'index.html'. The 'About' section notes 'No description, website, or topics provided.' The 'Languages' section shows 'HTML 100.0%'. The 'Pull requests' tab is highlighted with a red box.

Daisky1113 / pl-test

Pulls Issues Marketplace Explore

Unwatch 1 Star 0 Fork 0

Code Issues Pull requests 1 Actions Projects Wiki Security ...

main Go to file Add file Code About

No description, website, or topics provided.

Readme

Releases

No releases published Create a new release

Packages

No packages published Publish your first package

Languages

HTML 100.0%

プルリクを確認する

Practice



Reader

Screenshot of a GitHub repository page showing pull requests.

The repository is **Daisky1113 / pl-test**. The **Pull requests** tab is selected, showing 1 open pull request.

The pull request is titled **pull requestのテスト** and was opened 2 minutes ago by **Daisky-shadow**.

A modal dialog is displayed: **Label issues and pull requests for new contributors**. It states: "Now, GitHub will help potential first-time contributors [discover issues](#) labeled with [good first issue](#)". A [Dismiss](#) button is at the top right of the modal.

Filters applied: **is:pr is:open**. Other filter options shown are **1 Open** and **0 Closed**.

Other repository navigation links include: Code, Issues, Actions, Projects, Wiki, Security, Insights, Settings.

Repository stats: Unwatched (1), Starred (0), Forked (0).

Bottom right of the page: **New pull request** button.

Sort and filter dropdowns: Author, Label, Projects, Milestones, Reviews, Assignee, Sort.

ProTip!: Notify someone on an issue with a mention, like: @Daisky1113.

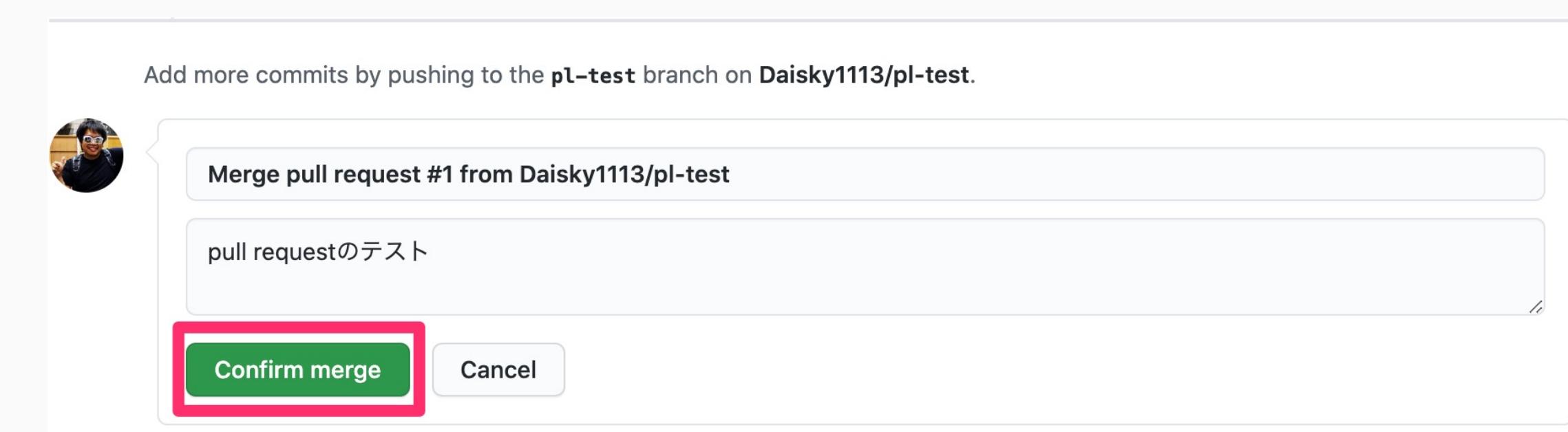
プルリクを確認する

Practice



Reader

The screenshot shows a GitHub pull request page for repository 'Daisky1113/pl-test'. The title is 'pull requestのテスト #1'. A green button labeled 'Open' is visible. The pull request details show a merge from 'pl-test' into 'main'. A comment from 'Daisky-shadow' says 'No description provided.' Below it, a commit titled 'pull requestのテスト' is shown. A note at the bottom says 'Add more commits by pushing to the pl-test branch on Daisky1113/pl-test.' A green box highlights the 'Merge pull request' button, which is also labeled with a red border.



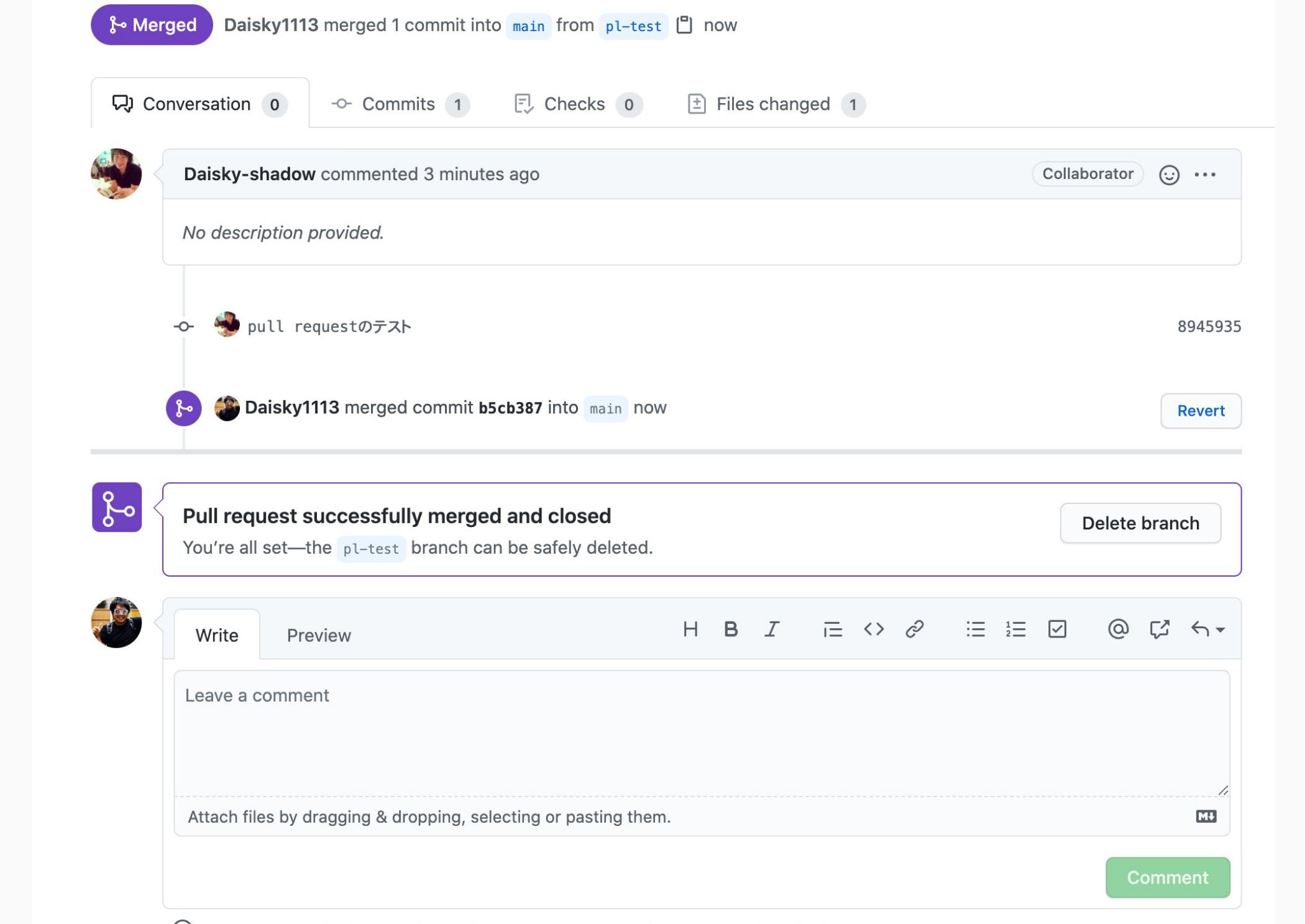
プルリクを確認する

Practice



Reader

pull requestのテスト #1



A screenshot of a GitHub pull request merge interface. At the top, a purple button indicates the pull request has been merged. Below it, a message from user Daisky1113 states: "Daisky1113 merged 1 commit into main from pl-test now". Below this, there are tabs for Conversation (0), Commits (1), Checks (0), and Files changed (1). A comment from user Daisky-shadow is shown, stating: "No description provided." and "-o pull requestのテスト". The commit hash b5cb387 is mentioned. A "Revert" button is available for this commit. A success message at the bottom says: "Pull request successfully merged and closed. You're all set—the pl-test branch can be safely deleted." A "Delete branch" button is also present. A comment input field is at the bottom with a "Comment" button.

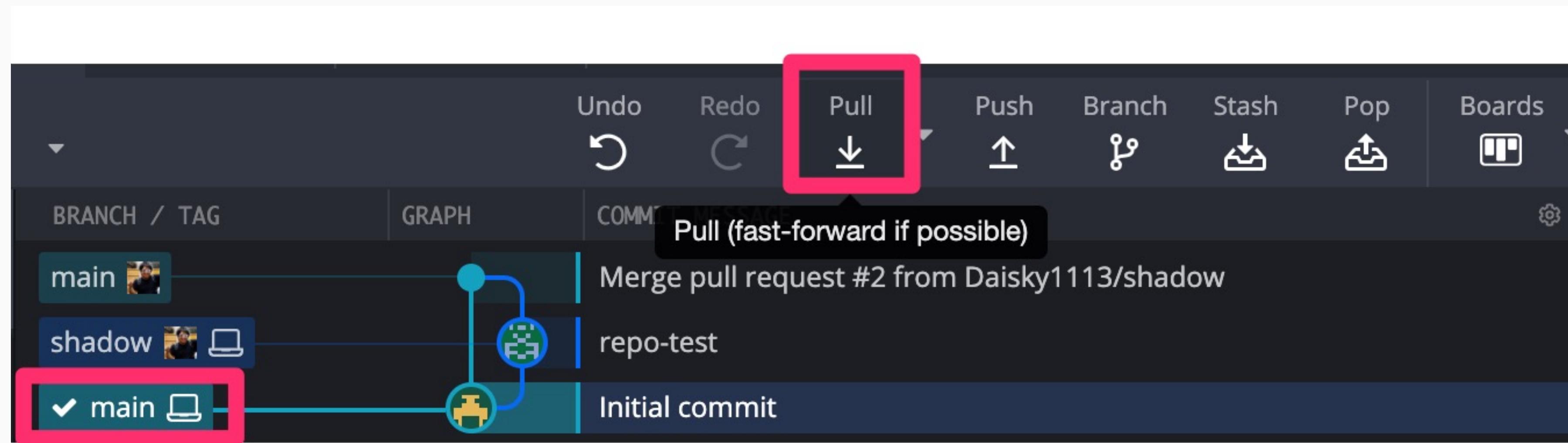
これでローカルの変更がリモートに反映された

リモートの状態をプルする

Practice



Reader



Mainブランチに切り替えてpullを押す

ここまでまとめ

Practice



プルリク・マージ・プルの流れ

Practice

最後のやってみよう

- ・リーダーはリモートリポジトリを作る
- ・リーダーはメンバーを招待する
- ・メンバーはリモートリポジトリをクローンする
- ・メンバーはブランチを切ってプルリクを送る
- ・リーダーはマージする
- ・メンバーはリモートの変更をpullする

基本的にこの流れで作業は進むのでやってみましょう