

Github ワークショップに向けての 事前準備マニュアル

Hack U Project

本イベントへの参加ありがとうございます!!

イベントに参加していただくにあたって

以下の6つの事前準備をしていただく必要があります

1. ワークショップでのZoomのセットアップ
2. ワークショップでのSlackのセットアップ
3. GitHubのアカウント作成
4. Sourcetreeのインストールとセットアップ
5. 演習準備
6. GitHubへの疎通確認



事前準備 1

ワークショップで利用する
Zoom のセットアップ

お願い（アプリ上の名前について）

イベントではZoomやSlackなど、複数のアプリケーションを使います。

それぞれのアプリの準備手順をこれから案内しますが、

そこで使うお名前（表示名）は、Connpassの「表示名」と必ず揃えてください。（参考）Connpassプロフィールページの「表示名」



学校など、本イベント以外で利用しているZoomやSlackには影響がありませんので、ご安心ください。

(事前準備1) ZOOM

ZoomはZoom Video Communications社が開発したWeb会議サービスです。

PC、Android、iOS、ブラウザから利用可能です。

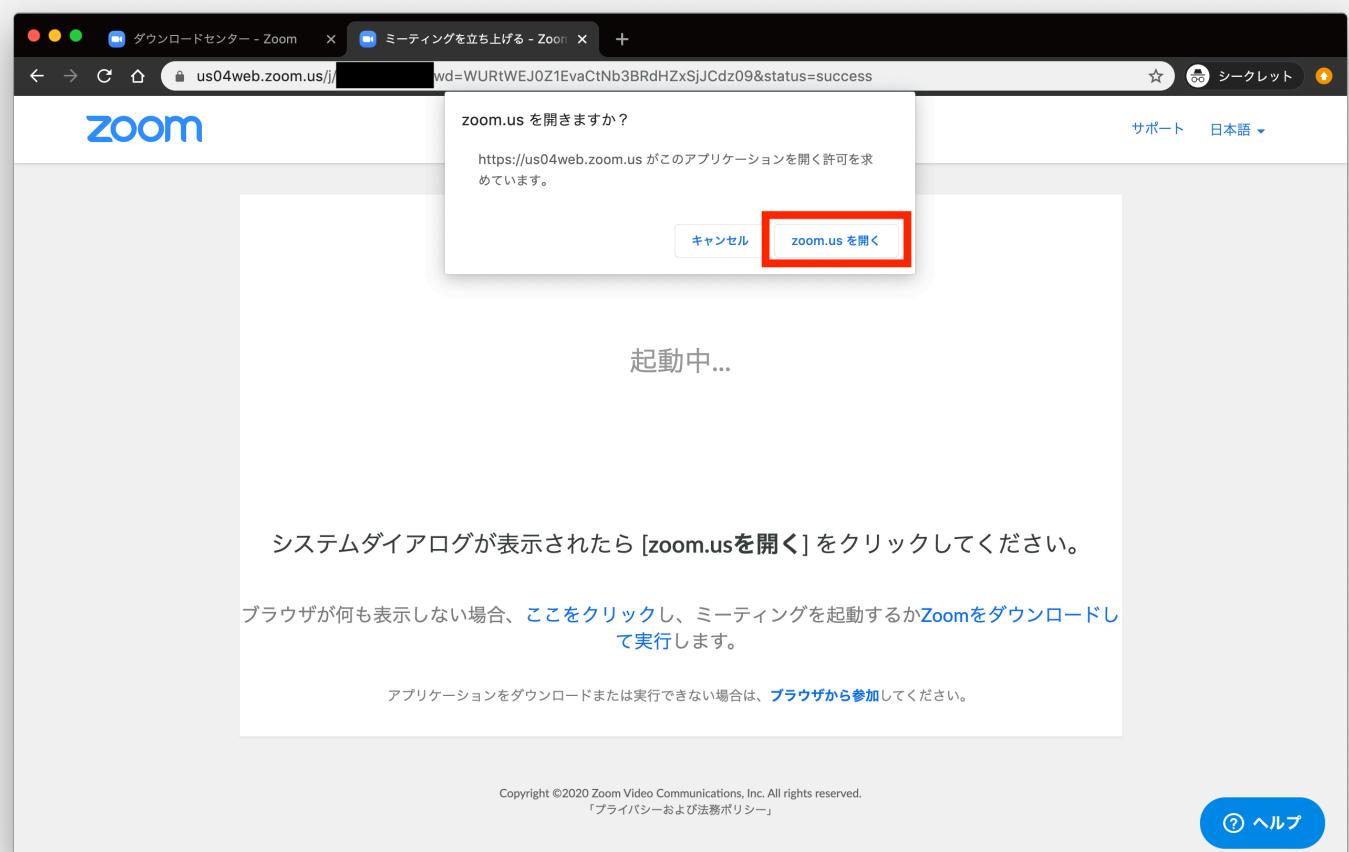
今回のオンラインイベントはZoom（とSlack）を利用します。



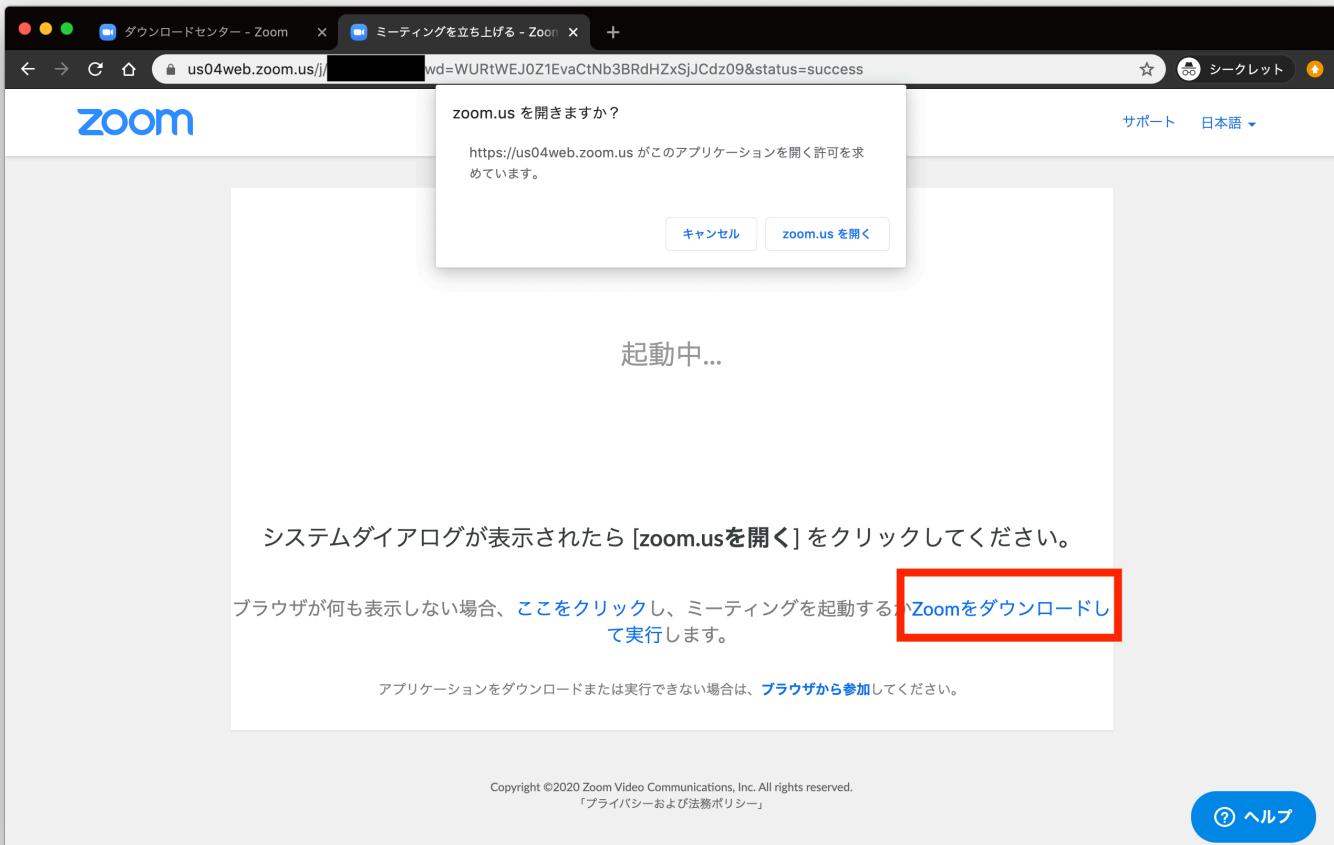
(事前準備 1) 利用方法

別途配布したURLをPCのブラウザから開くと、右図の画面になります。

すでにZoomがインストールされていれば、[zoom.usを開く]でアプリを起動してください。



(事前準備1) アプリインストール



※インストールが済んでいない方のみ

[Zoomをダウンロードして実行]のからインストーラーをダウンロードできます。

インストーラーの指示に従い、インストールを完了させてください。

完了後、もう一度リンクを開いて
[zoom.usを開く]からアプリを起動してください。

(事前準備 1) 手動入力で参加する場合 (1/3)



もしURLを開いてもポップアップ
ウィンドウが出てこない場合は、
アプリを起動して、[ミーティングに参加]から部屋番号を
手入力することで参加できます。

アカウントは必須ではないです
が、アカウントをお持ちの方は
[サイン イン]してからミーティ
ングに参加できます。

(事前準備 1) 手動入力で参加する場合 (2/3)



別途配布したURLに入っている
9~10桁の数字を[会議IDまたは
会議室名]に入力してください。

次に[名前]の欄に、お名前
(Connpassの表示名) を入力
した上で[参加]ボタンを押しま
す。

(事前準備 1) 手動入力で参加する場合 (3/3)



もしパスワードの入力が求められたら、メールにてご連絡しているパスワードを貼り付けてください。

以上で入室の手続きが完了します。

(事前準備1) マイク・カメラチェック

入室したら右図のような画面になります。

画面の左下にマイクとカメラの
オン/オフを切り替えるボタン
があります。

イベントが開始する前に簡単に
マイクとカメラの接続をご確認
してください。



(事前準備 1) connpass の表示名について

connpass のプロフィールページにおける、
この部分の表示名を Slack, zoom に
必ず 登録するようにしてください。

Hack U (Yahoo! JAPAN)

(HackU)



プログラミングやデザインを通して、あなたのアイデアを形にしてみませんか？

Hack U（ハック・ユー）は、限られた期間の中で、学生がプロダクトを自ら企画・開発・発表するハッカソンや各種ワークショップなどを開催しています。

ものづくりの楽しさを体現できるよい機会となるよう、ヤフーの現役社員が学生のみなさんを全力でサポートいたします！



プロフィールを変更する

利用開始日: 2018/05/22



事前準備 2

本ワークショップで利用する
Slack のセットアップ

(事前準備2) 本マニュアルの流れ

1. Slackのワークスペースに参加しよう
2. 本イベントでのSlackチャンネルの使い方

(事前準備2) Slackのワークスペースに参加しよう

- Slackの招待URLからアクセス
- 「HackUイベント企画(または hacku-event)に参加する」と表示されていることを確認して、自分のメールアドレスを記入
- 「メールをチェックしてください」という表示が出たら、登録したメールアドレスに書かれた「ここをクリックして続行」を押す
- 氏名 (**Connpassの表示名**) とパスワードを入力してアカウントを作成



The image displays two screenshots of the Slack sign-up process. The left screenshot shows the initial step where a user is invited to join the 'HackUイベント企画' workspace. It asks for the user's email address ('あなたのメールアドレスは?') and provides a field with the placeholder '自分の名前@example.com'. A green button labeled 'メールアドレスを確認する' (Check email address) is visible. The right screenshot shows the final step of creating the account. It requires the user to enter their name ('氏名') and password ('パスワード (必須)'). Below these fields, there is a note: 'あなたの名前が送信するメッセージと共に表示されます。' (Your name will be displayed next to the messages you send). A checkbox labeled 'Slack のサービスに関するメールの受信を希望します。' (I would like to receive emails about Slack's services.) is checked. At the bottom, a large grey button labeled 'アカウントを作成する' (Create account) is prominent. The background of the right screenshot features a blue theme with icons of people and files.

(事前準備2) connpass の表示名について

connpass のプロフィールページにおける、
この部分の表示名を Slack, zoom に
必ず 登録するようにしてください。

Hack U (Yahoo! JAPAN)

(HackU)



プログラミングやデザインを通して、あなたのアイデアを形にしてみませんか？

Hack U（ハック・ユー）は、限られた期間の中で、学生がプロダクトを自ら企画・開発・発表するハッカソンや各種ワークショップなどを開催しています。

ものづくりの楽しさを体現できるよい機会となるよう、ヤフーの現役社員が学生のみなさんを全力でサポートいたします！



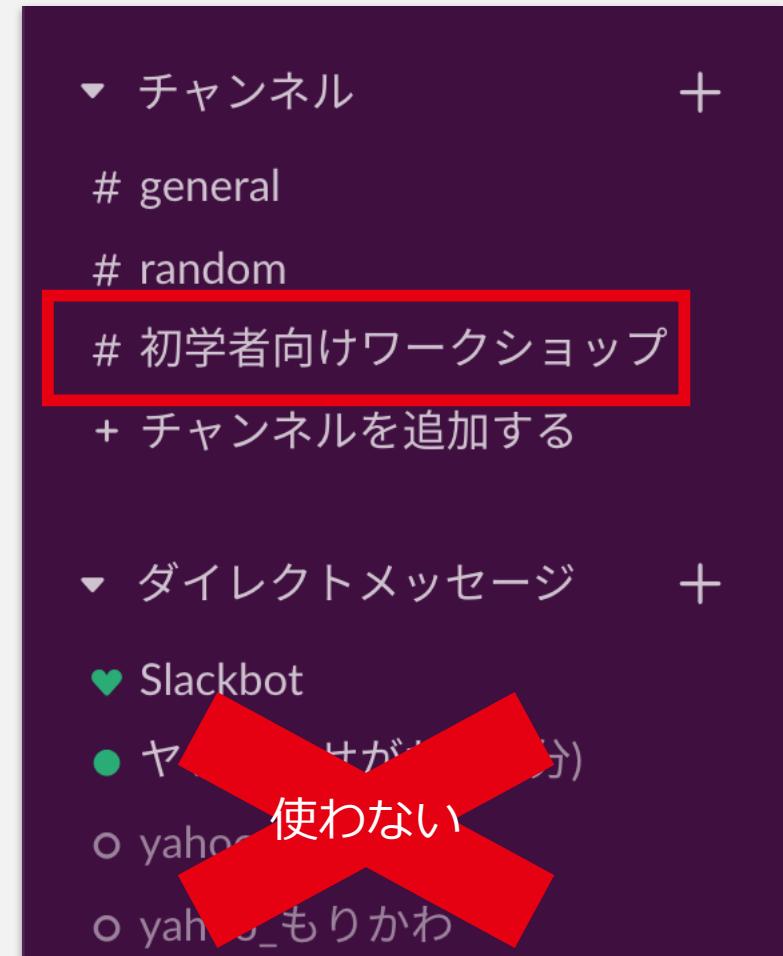
プロフィールを変更する

利用開始日: 2018/05/22



(事前準備2) 本イベントでのSlackチャンネルの使い方

- 運営からの全体連絡
「# 初学者向けワークショップ」チャンネル
- 演習中
学生さん数人の「#group_XX」チャンネル
Zoomもそれぞれのグループに招待します



事前準備 3

GitHub のアカウント作成

Git と Github

Git	バージョン管理の方式 (バージョン管理とは、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)
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のがかんたんにできる

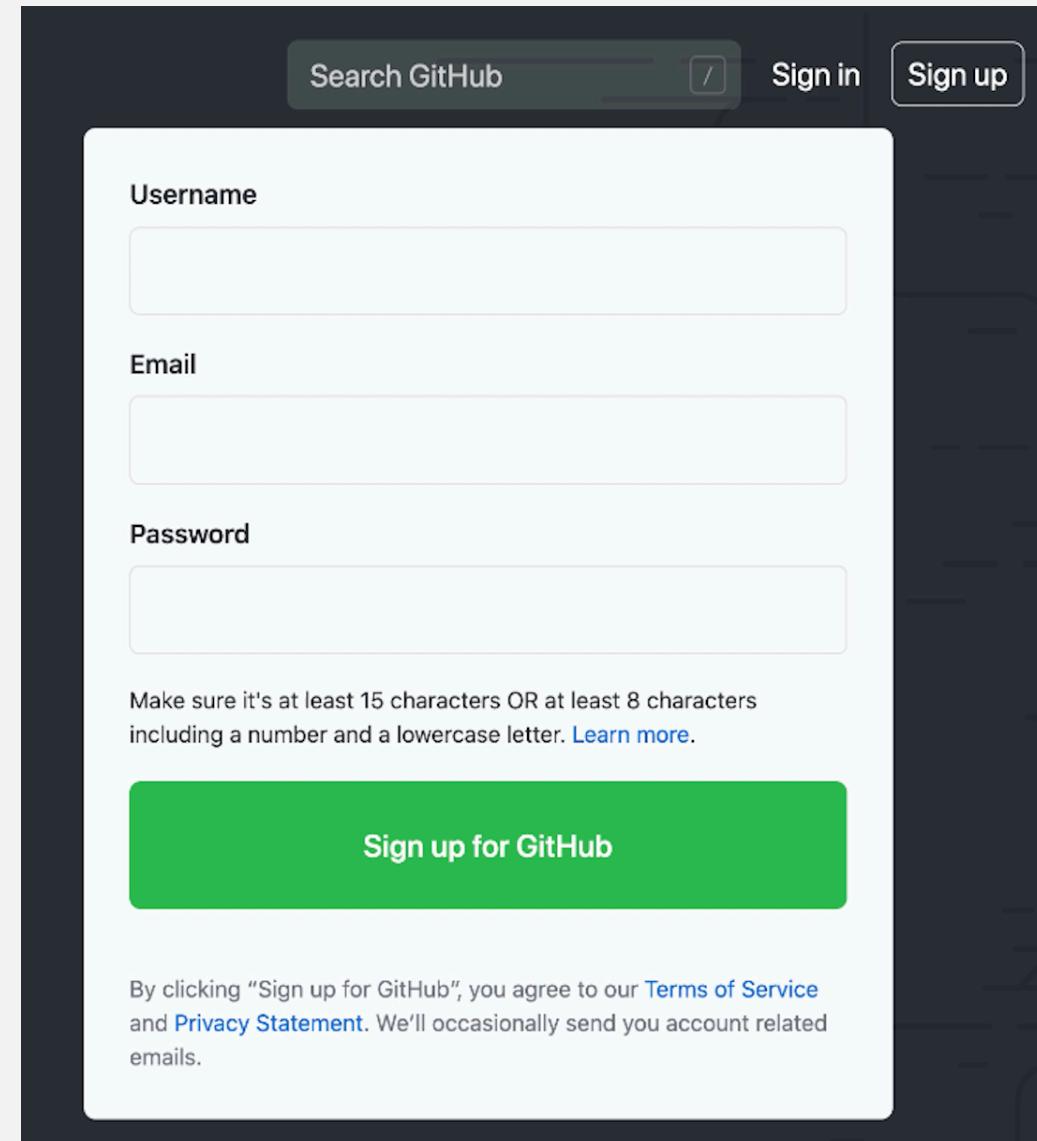
Git を使うには

- 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 Sourcetree という
GUIツールを使います

Github のアカウントを作る

- 右上のSign upをクリック
 - <https://github.com>
- 必要事項を記入する
 - アカウント名
 - メールアドレス
 - パスワード



アカウント名は公開されます

Github のアカウントを作る

- 認証
- Welcome to GitHub
 - 仕事は何か
 - どのくらいプログラミングができるか
 - 何のためにGitHubを使うか
 - 興味があるもの
- メールを確認 “Verify email address” でアクセス

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>

アカウント作成されていない場合、404になります

リポジトリを作る

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 *

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

リポジトリを作る

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 *

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」
というリポジトリ名で
作りましょう

リポジトリを作る

Create repositoryをクリック

Owner には先程作ったアカウント名が入る

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)

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 *

Owner: アカウント名 / Repository name: (未入力)

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

Description (optional)

リポジトリの説明 (オプション) (未入力)

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」
というリポジトリ名で
作りましょう

事前準備 4

Sourcetree のセットアップ

Sourcetree のインストール

- Sourcetree のダウンロード
 - <https://www.sourcetreeapp.com/>
- zip ファイルを展開して、ファイルをアプリケーション以下に移動したら完了

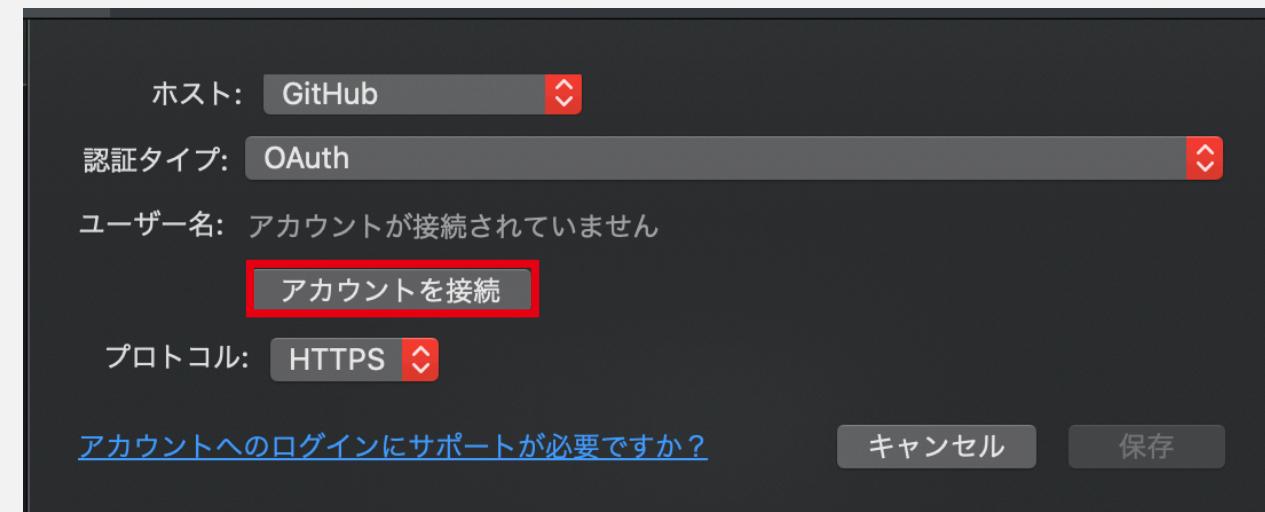
Sourcetree と Github を連携 (Mac)

- 右上の歯車アイコンから「アカウント」を選択
- 「アカウント」を選択して「追加…」を選択



Sourcetree と Github を連携 (Mac)

- 右図のように設定、「アカウントを接続」を選択
- ブラウザでアクセス許可を聞かれるため、「Authorize atlassian」を選択
- ユーザー名が表示されたら「保存」を選択



Sourcetree と Github を連携

アカウント連携後、「リモート」を選択して、先程作ったリポジトリが表示されていれば連携成功🎉



事前準備 5

演習準備

ローカルリポジトリの作成

- Git の機能の一つ `clone` を使って、GitHub で作成した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

- リモートに作成したリポジトリの横にある「クローン」ボタンを押す

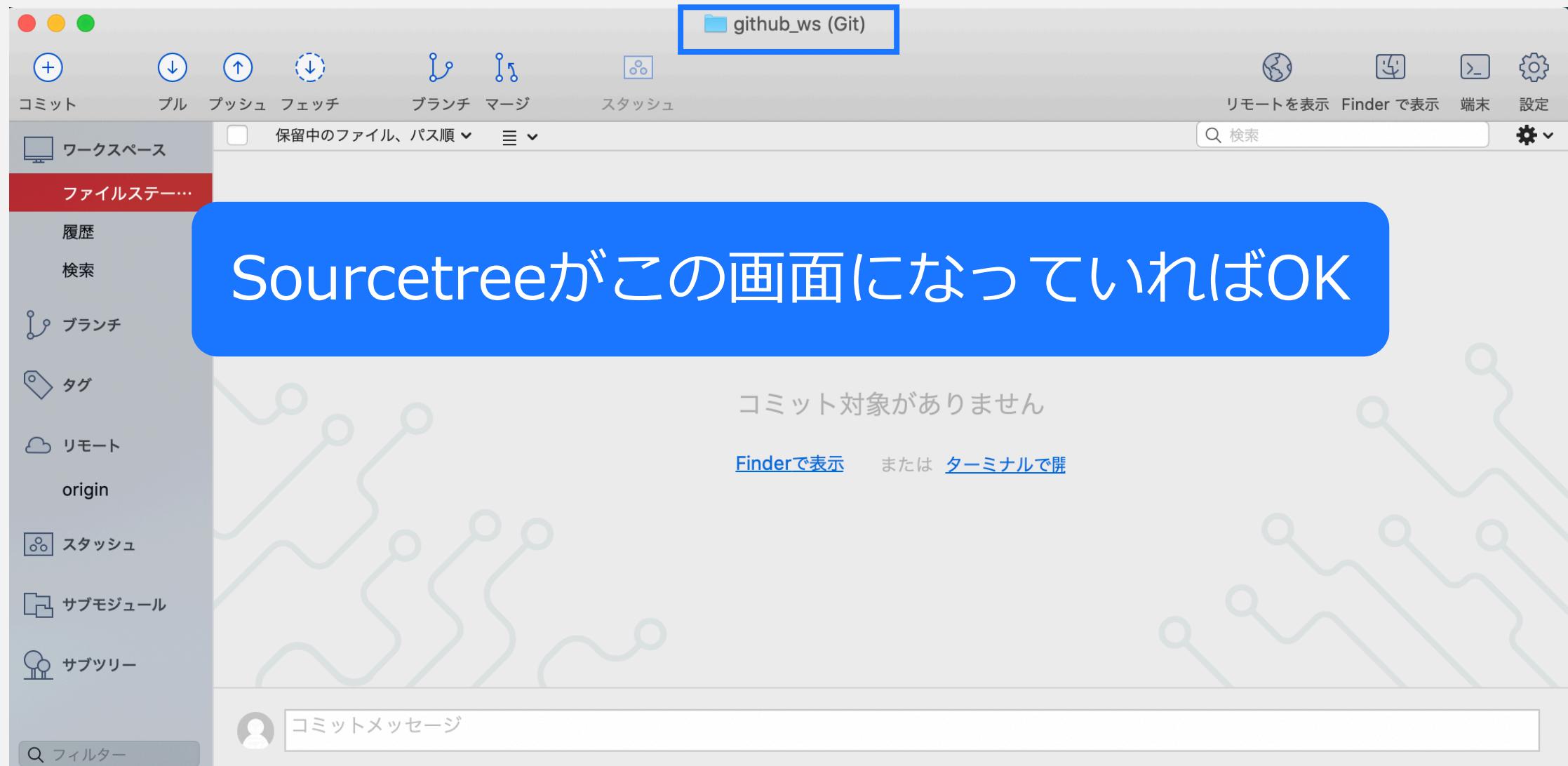


ローカルリポジトリの作成

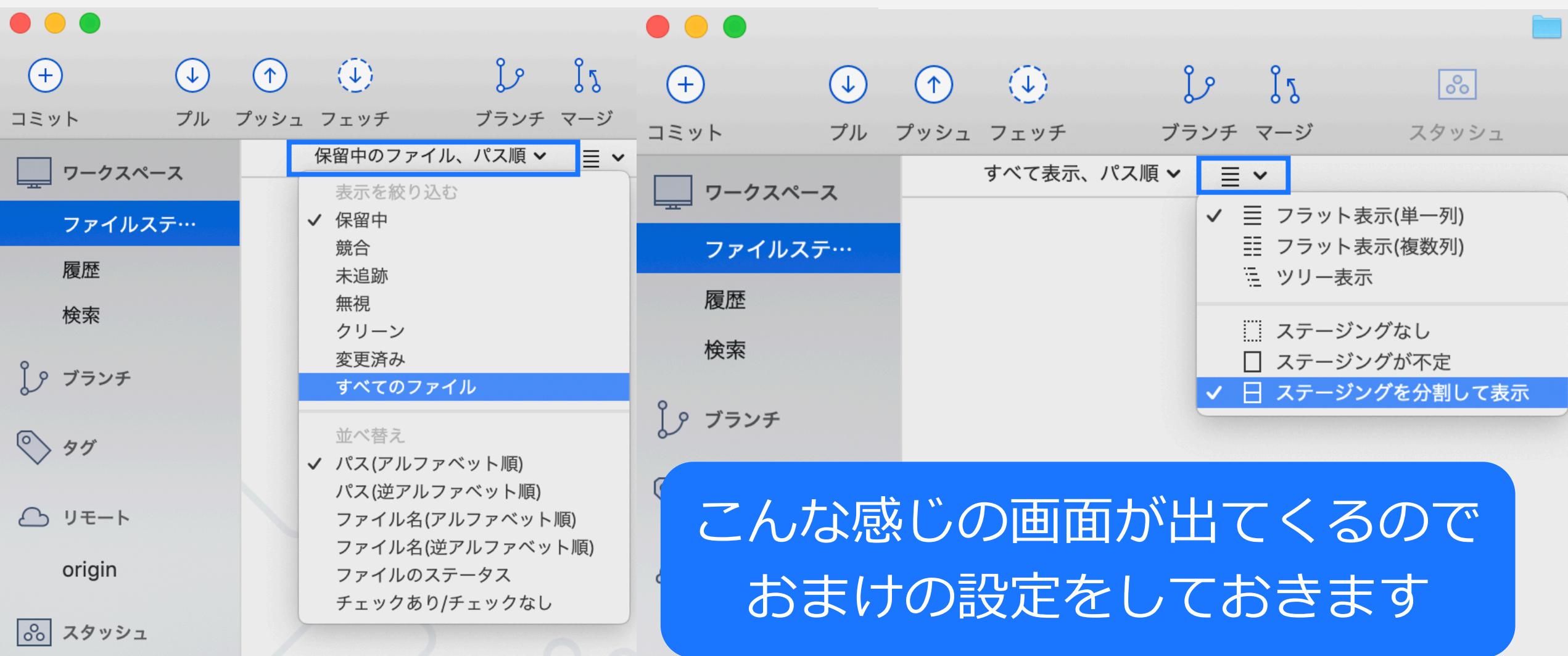
- 出てきた項目を下記のように修正
- 右下のクローンを押す



ローカルリポジトリの作成



Sourcetreeの設定

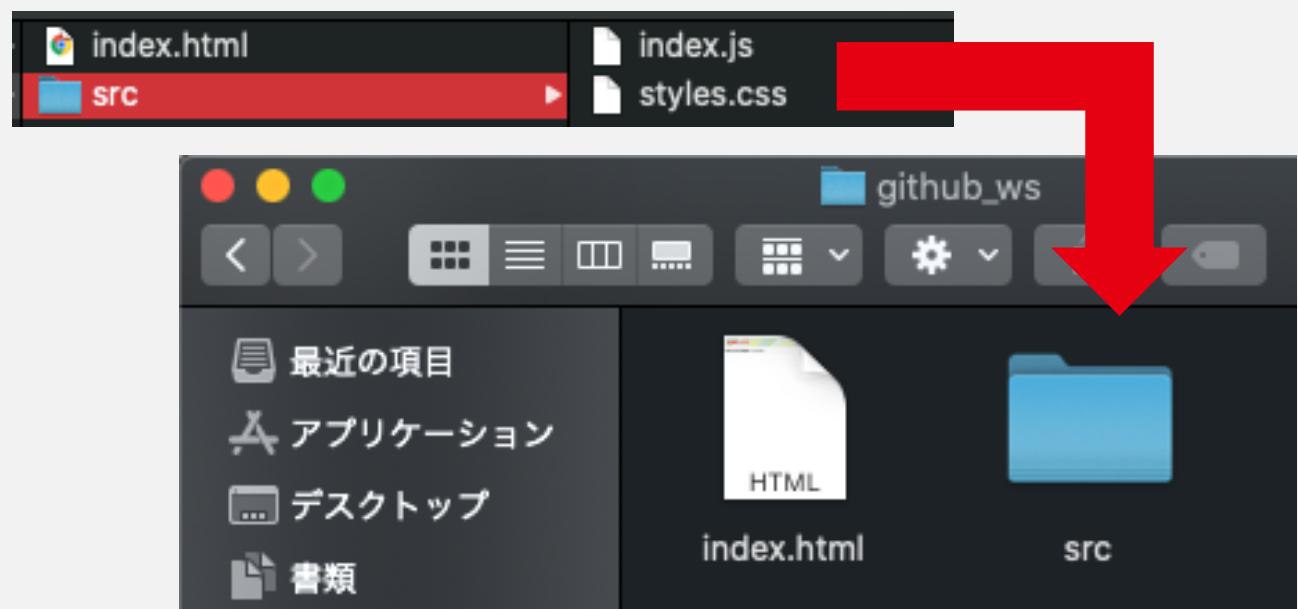


演習ファイルの準備

- GitHub の Search or jump to … から「hackujp/github_tutorial」を検索
- リポジトリ内の ver_GUI/hands-on.zip を開く
 - Download ボタンよりダウンロード
- hands-on.zip を解凍

演習ファイルの準備

- ・ 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
 - ・ 移動させる対象は以下の通り
 - ・ index.html
 - ・ src



事前準備 6

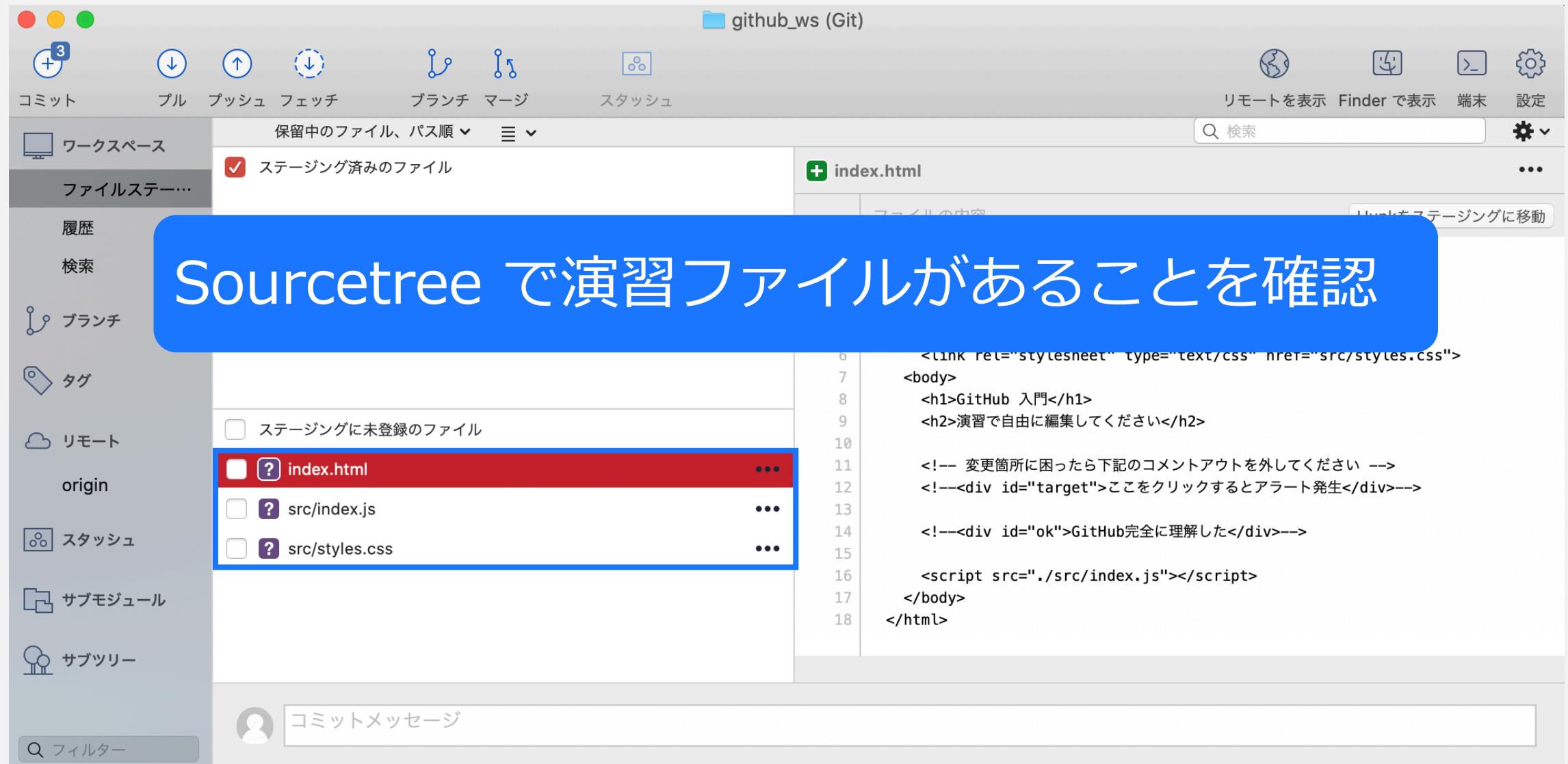
GitHubへの疎通確認



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)



GitHubへの疎通確認 (add)

すべてファイルを

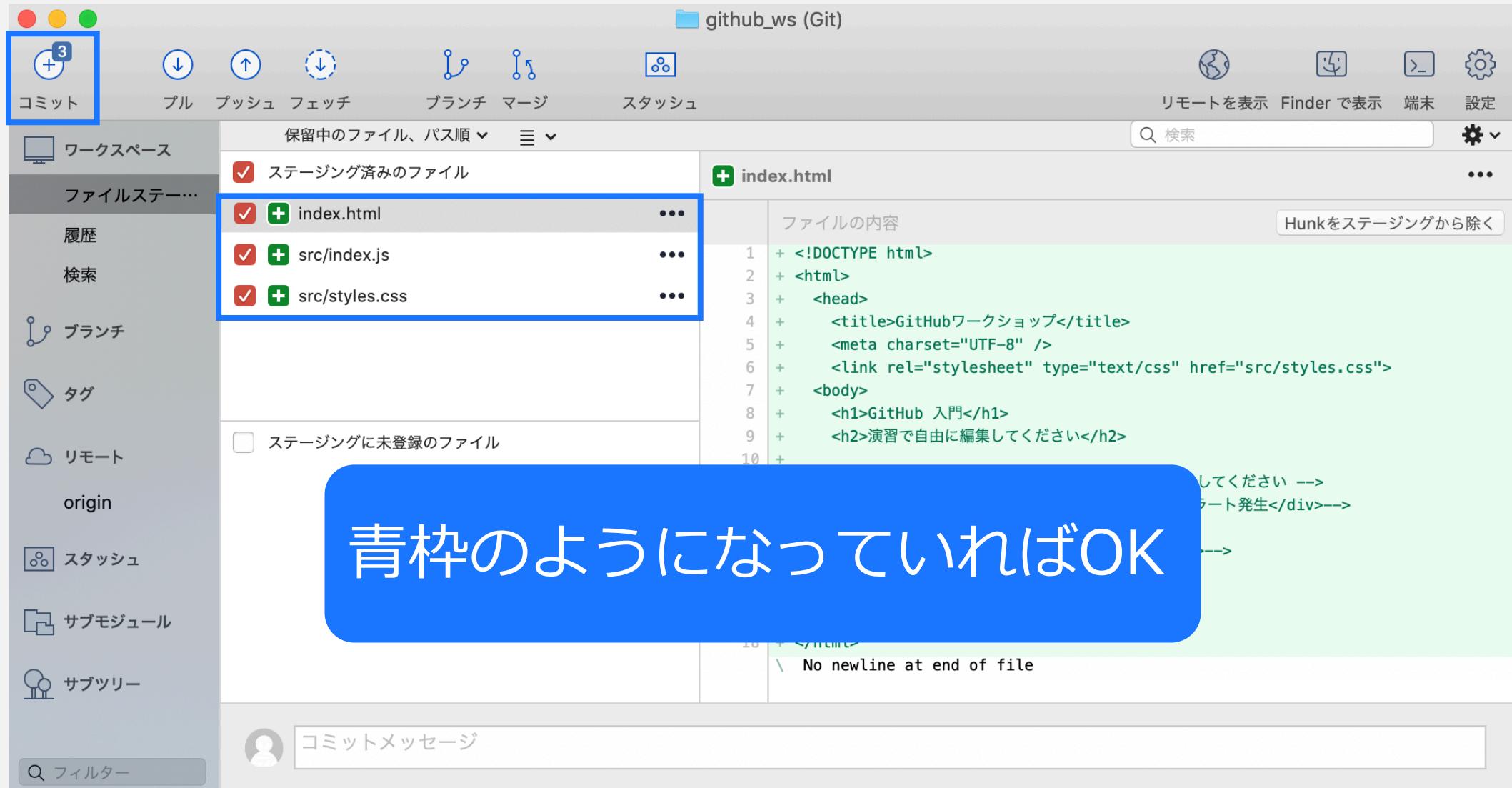
- ・チェックする
- ・上にドラッグ&ドロップする

のいずれかの操作で add される

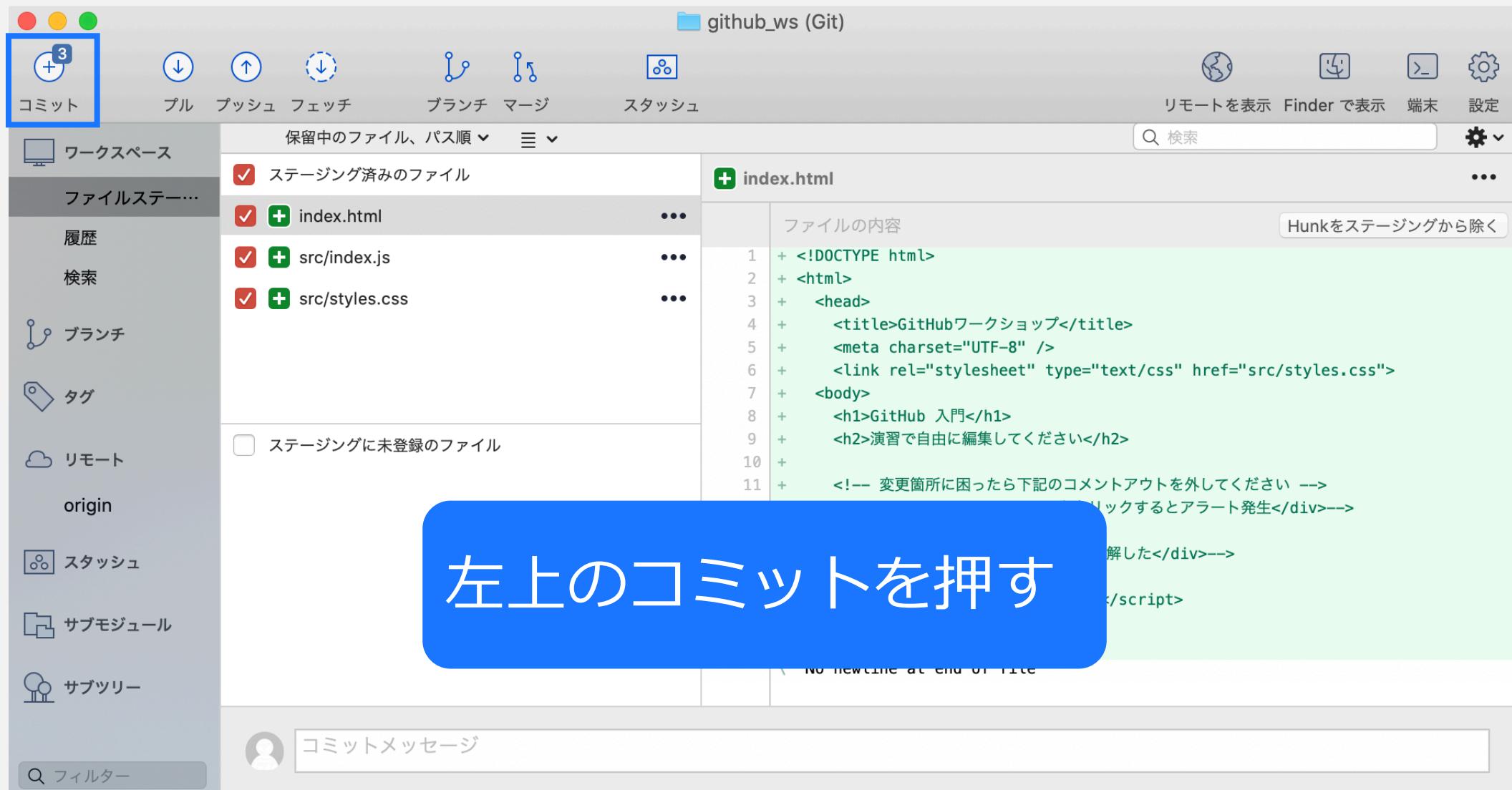
The screenshot shows the GitHub desktop application interface. On the left, there's a sidebar with various options like 'コミット' (Commit), 'プル' (Pull), 'プッシュ' (Push), 'ブランチ' (Branch), 'マージ' (Merge), 'ワークスペース' (Workspace), 'ファイルステータス' (File Status) which is selected, '履歴' (History), '検索' (Search), 'ブランチ' (Branch), 'タグ' (Tag), 'リモート' (Remote), 'origin', 'スタッッシュ' (Stash), 'サブモジュール' (Submodule), and 'サブツリー' (Subtree). At the bottom, there's a 'フィルター' (Filter) button and a 'コミットメッセージ' (Commit Message) input field with a user icon. The main area shows a list of files in the staging area: 'index.html' (selected), 'src/index.js', and 'src/styles.css'. To the right, the file content of 'index.html' is shown:

```
<h2>演習で自由に編集してください</h2>
<!-- 変更箇所に困ったら下記のコメントアウトを外してください -->
<!--<div id="target">ここをクリックするとアラート発生</div>-->
<!--<div id="ok">GitHub完全に理解した</div>-->
<script src=".src/index.js"></script>
</body>
</html>
```

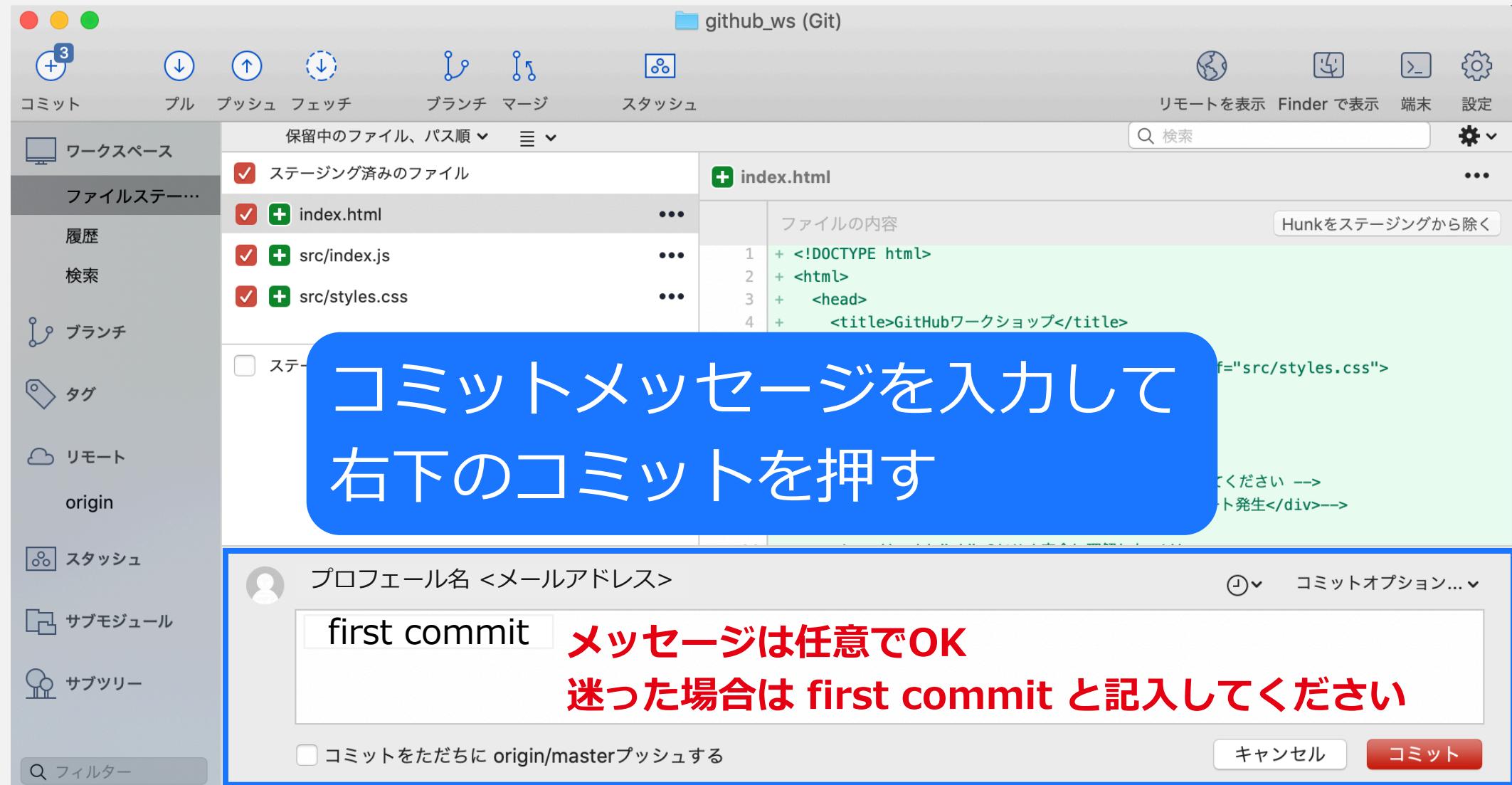
GitHubへの疎通確認 (add)



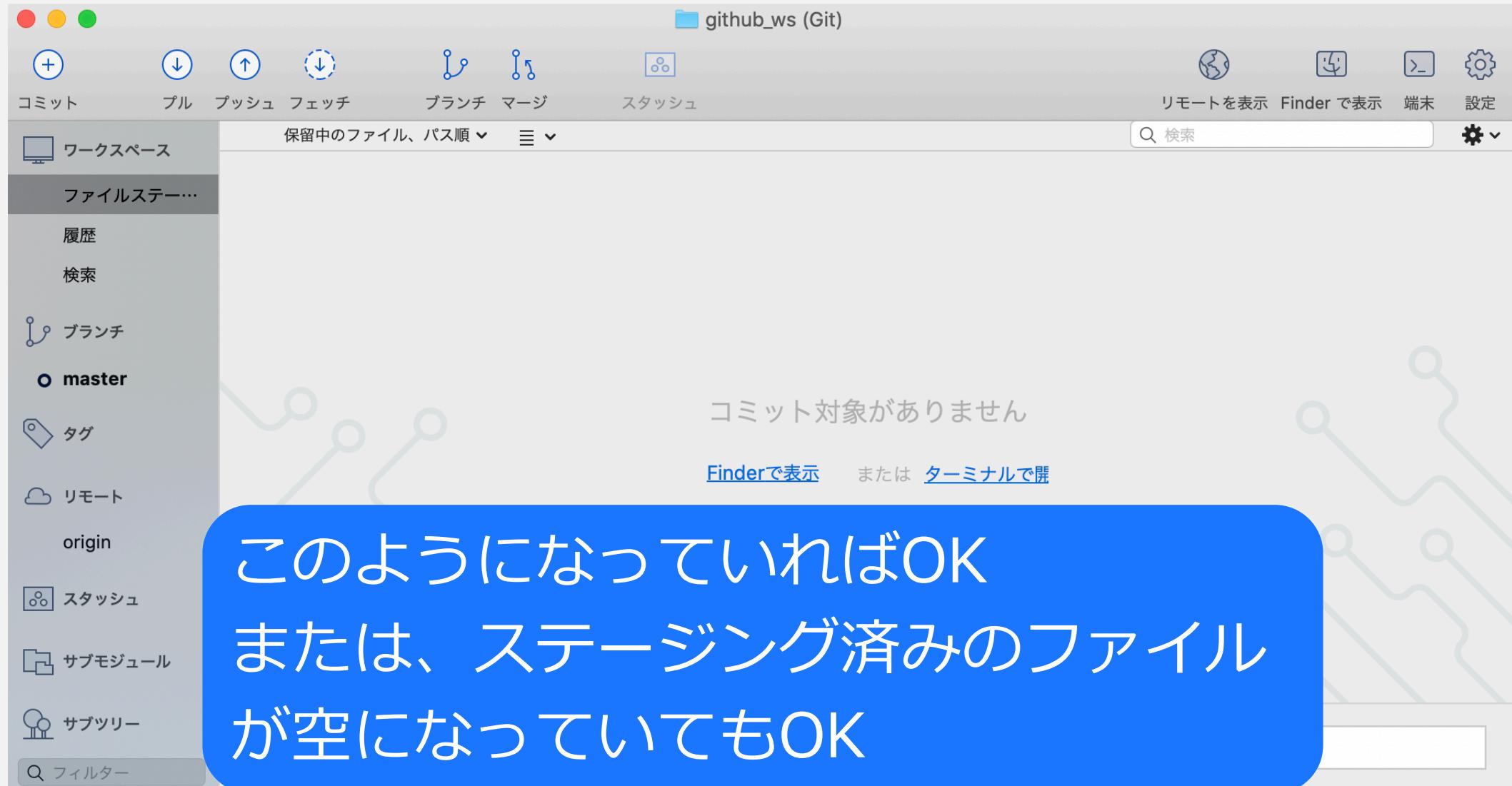
GitHubへの疎通確認 (commit)



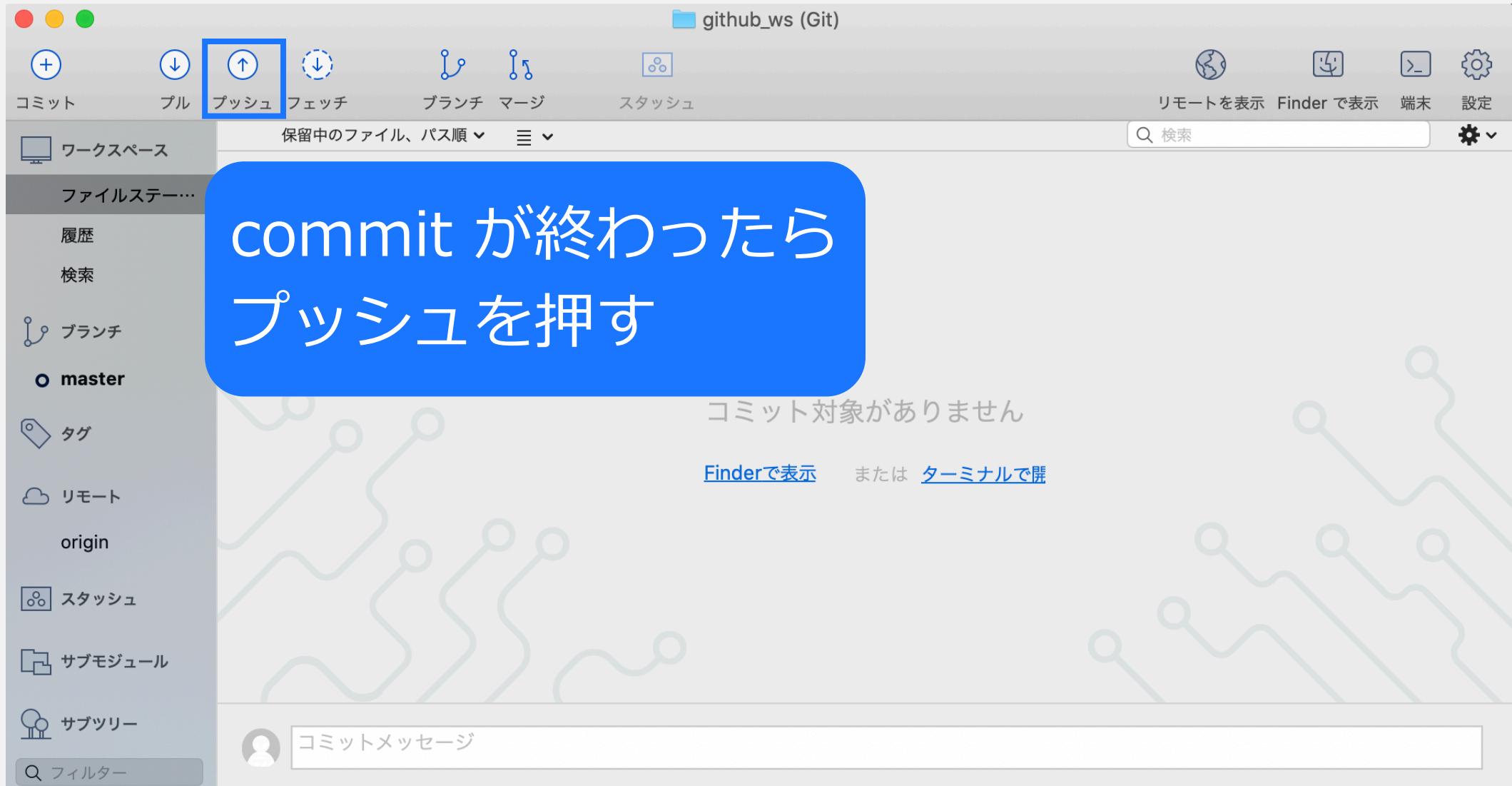
GitHubへの疎通確認 (commit)



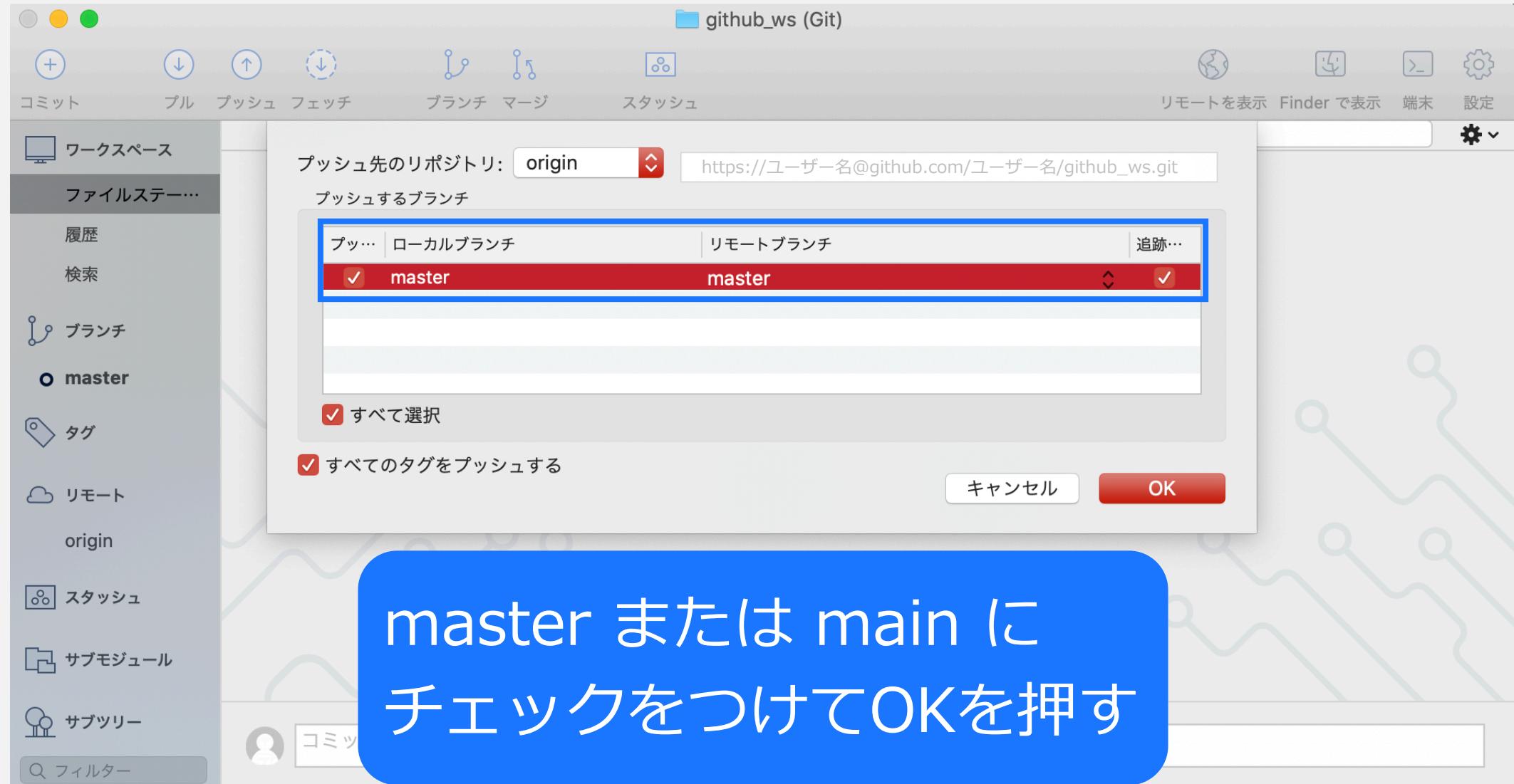
GitHubへの疎通確認 (commit)



GitHubへの疎通確認 (push)



GitHubへの疎通確認 (push)



GitHub に反映されているか確認

Screenshot of a GitHub repository page for 'github_ws'.

The repository details:

- Owner: ユーザー名 / [github_ws](#)
- Code: master branch, 1 branch, 0 tags
- Last commit: 50cdc45 14 minutes ago (1 commits)
- Files added: src (Add index.html & src), index.html (Add index.html & src)
- Contributors: 14 minutes ago
- Commits: 14 minutes ago

Repository status:

- Unwatched (1 watch)
- Starred (0 stars)
- Forked (0 forks)

About section:

No description, website, or topics provided.

Releases section:

No releases published

[Create a new release](#)

Call to action:

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

[Add a README](#)

Pushがうまくいかない時①

- 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

Pushがうまくいかない時②

- Push時のURLの問題

- SourceTreeの[設定] -> [リモート]からoriginのURLを確認し、URLの「github.com」の前が「ユーザー名@」となっていた場合、この「ユーザー名@」を消すとうまくいく場合があります
- 変更後のURL : https://github.com/{自分のユーザー名}/github_ws
- 上記をやってもうまくいかない場合、「Sourcetree と Github を連携」でやったOauth認証の部分をBasic認証に変えて試してみましょう



事前準備は以上となります
お疲れさまでした