

<前提>

この手順を使用する前提条件

- ・Git管理するための
 - ・Git Bashのインストール
 - ・Eclipseのインストール
 - ・対象のプロジェクト（ディレクトリ）
 - ・リモートリポジトリ（GitHub）

の準備が出来ていること

<目次>

■初回コミット準備

- ☐ ローカルリポジトリの作成（Eclipseで実施）
- ☐ 既存のローカルGitリポジトリを追加（Eclipseで実施）
- ☐ パーソナルアクセストークンの発行

■コミット

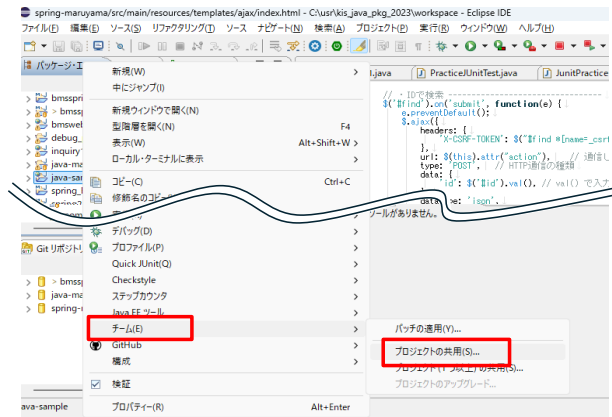
- ☐ 「ステージされていない変更」欄からaddする
- ☐ 「コミットメッセージ」を入力する
- ☐ コミットおよびプッシュ

■初回コミット準備

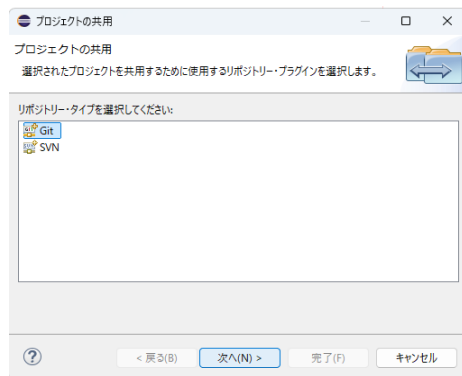
□ ローカルリポジトリの作成 (Eclipseで実施)

(※CUIを含め、**まだ一度も初回コミットを行っていない場合**。実行済の場合は飛ばして次の「既存のリポジトリ～」に進む)

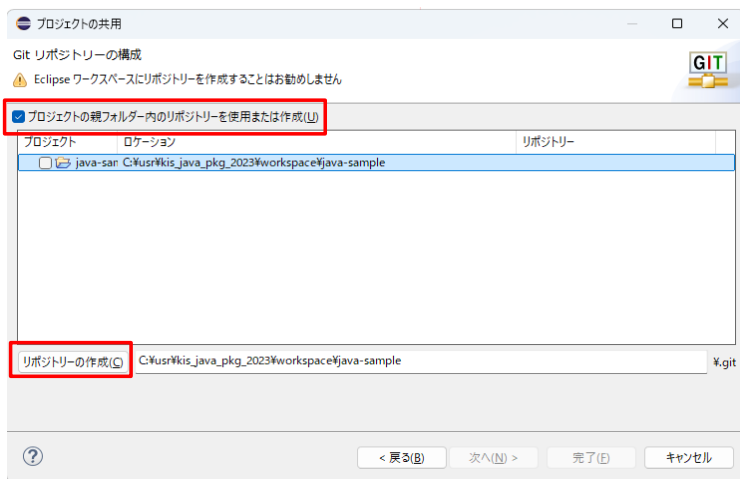
- ・プロジェクト名で右クリック>チーム>プロジェクトの共有



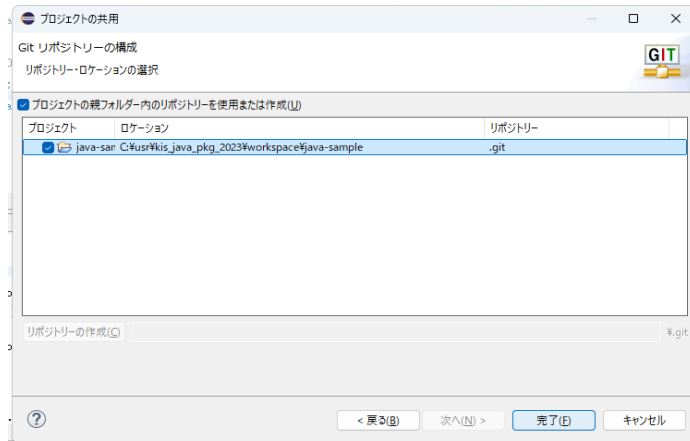
- ・「Git」を選択して「次へ」



- ・「プロジェクトの親フォルダ～」にチェックを入れ、「リポジトリの作成」をクリック

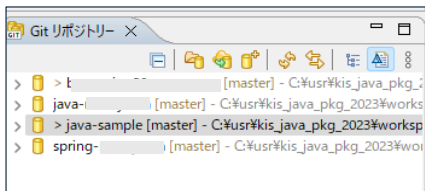


- ・下図のような表示になったら「完了」をクリック



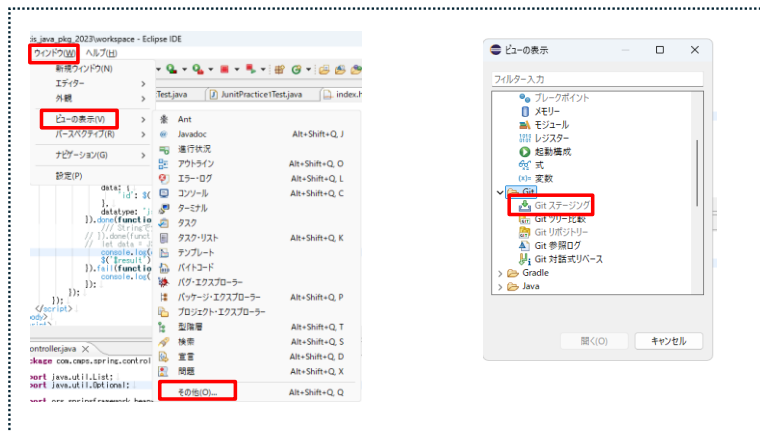
- ・「Gitリポジトリ」ビューに追加されたことが確認できます

※他にプロジェクトがない場合は、先ほど選択した自身のプロジェクトのみが表示されている状態



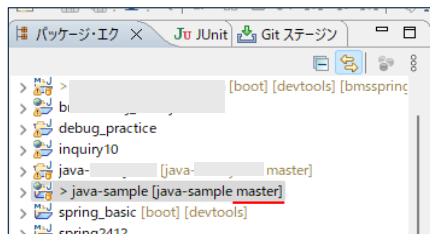
※「Gitリポジトリ」ビューが表示されていない場合は、以下で開くこともできます

ウィンドウ>ビューの表示>その他>Gitの中から選択して開く



- ・「パッケージ・エクスプローラー」ビューのプロジェクト名の右端にも

master (main) と表示されていることが確認できます。



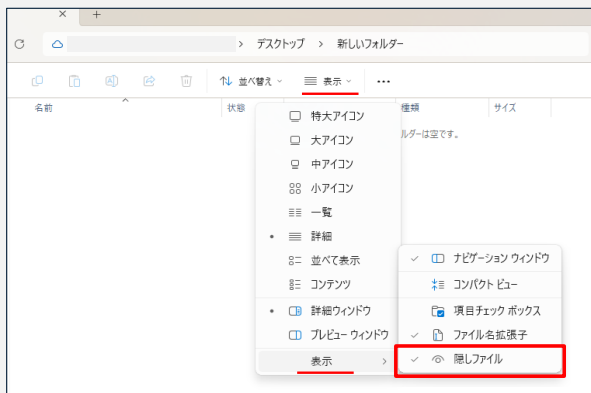
□ 既存のローカルGitリポジトリを追加（Eclipseで実施）

（※CUIを含め、初回コミットを行ったことがある場合。初めて行う場合は飛ばして次の「パーソナルアクセス〜」に進む）

（準備）

- ・エクスプローラーを開き、表示 > 表示 > 隠しファイル の順でクリックして開き、「隠しファイル」にチェックが付いていることを確認する。

チェックがついていない場合はクリックし、チェックが付いている状態にする。

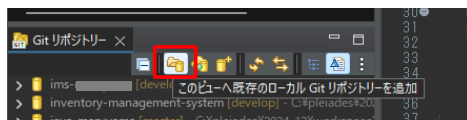


- ・「Gitリポジトリ」ビューを開く。

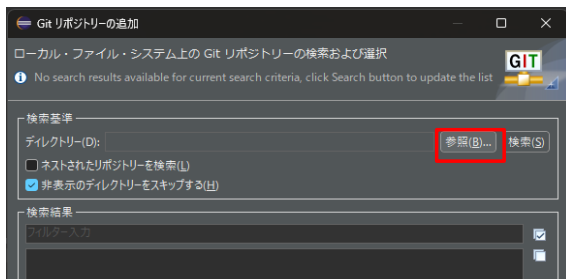
開いていない場合は全項目の「※「Gitリポジトリ」ビューが表示されていない場合〜」を参照して開く。

（以下、手順）

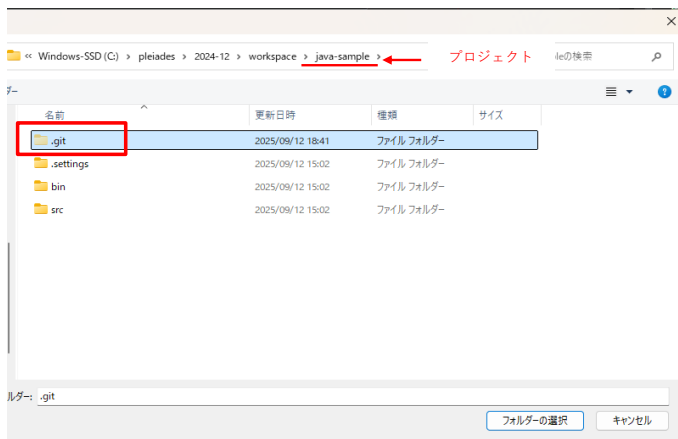
- ・下記のアイコン、または「既存のローカルGitリポジトリを追加」をクリックする。



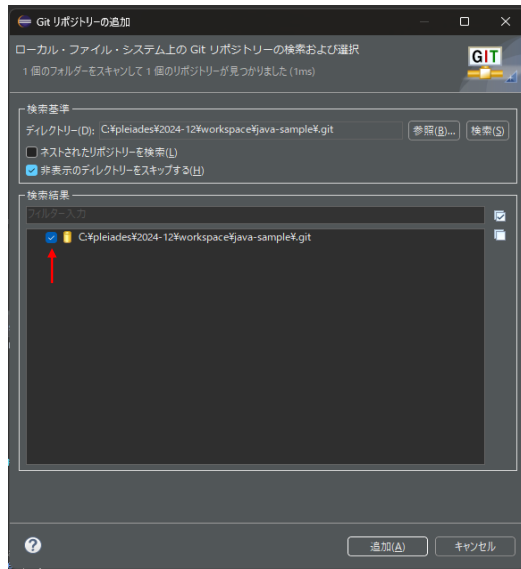
- ・「参照」をクリック



- ・対象プロジェクトのフォルダを開き、「.git」フォルダを選択して、「フォルダーの選択」をクリック



- ・「検索結果」に表示されたフォルダにチェックを入れて、「追加」をクリック

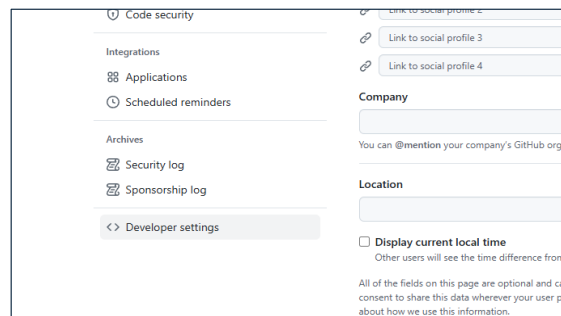
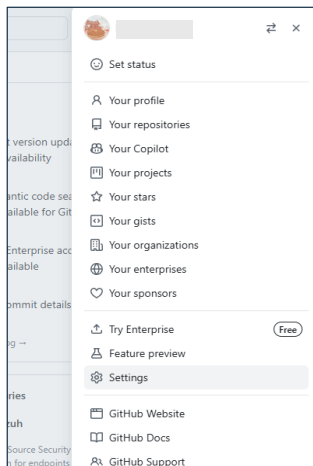


以上で「Gitリポジトリ」ビューにローカルリポジトリが追加されます。

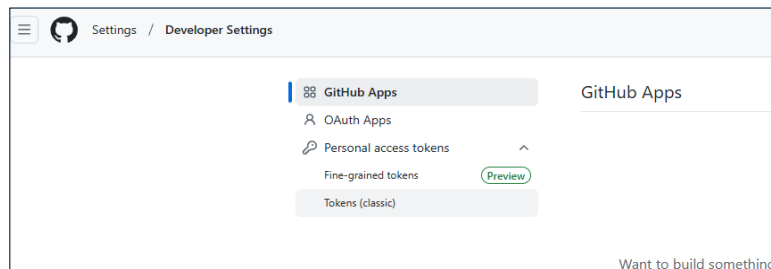
□ パーソナルアクセストークンの発行

※トークンの有効期限が切れたら再度発行する必要あり

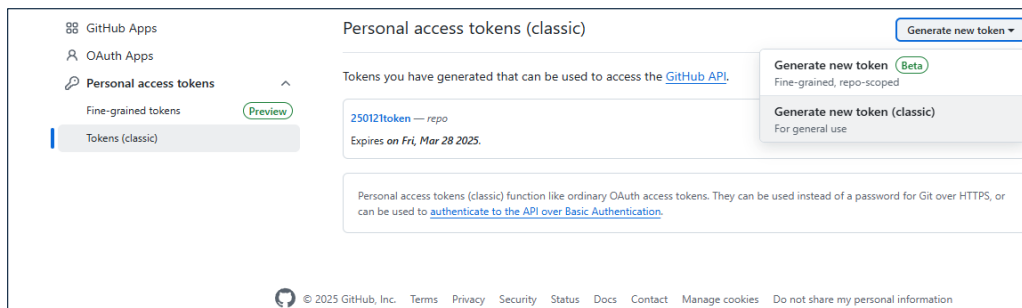
- ・ブラウザでGitHubを開く
- ・GitHubのトップページから右上のメニューを開く > Settings > 次の画面の下部にある「Developer settings」



- ・ Personal access tokens > Tokens (classic)



- ・ Generate new token > Generate new token (classic)



- ・ 以下を入力・選択

Note: パーソナルアクセストークンの利用用途（名称）を記述

→ 自分が分かればよいので今日の日付 + eclipse等でOK

Expiration: パーソナルアクセストークンの利用期限を設定

→ 特にこだわらなければ90daysでOK

Select scopes: このトークンがもつ権限。Eclipseからリポジトリ操作を行いたいので repo にチェック

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

20250314eclipse

What's this token for?

Expiration *

90 days The token will expire on Thu, Jun 12 2025

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes](#).

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input type="checkbox"/> repo:status	Access commit status
<input type="checkbox"/> repo_deployment	Access deployment status
<input type="checkbox"/> public_repo	Access public repositories
<input type="checkbox"/> repo:invite	Access repository invitations
<input type="checkbox"/> security_events	Read and write security events

- ・ 下記のようにトークンが発行されるので、コピーして控えておく。

Personal access tokens (classic) Generate new token

Tokens you have generated that can be used to access the [GitHub API](#).

Make sure to copy your personal access token now. You won't be able to see it again!

ghp_CL3... Copy Delete

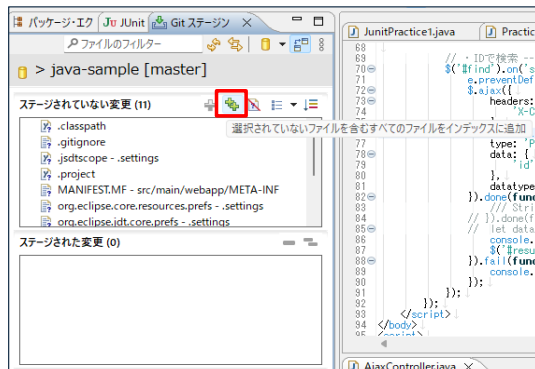
■コミット

- ・「Gitステージング」ビューを開く
→※初回・開いていない場合は、プロジェクトを右クリック>チーム>コミット

□「ステージされていない変更」欄からaddする（=今回のコミットに含めたいファイルを選択）

→※初回は全件addする

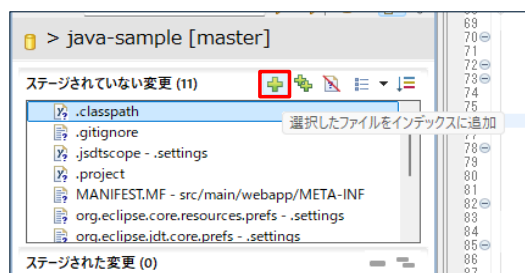
【全件add】 「++」マークをクリックする



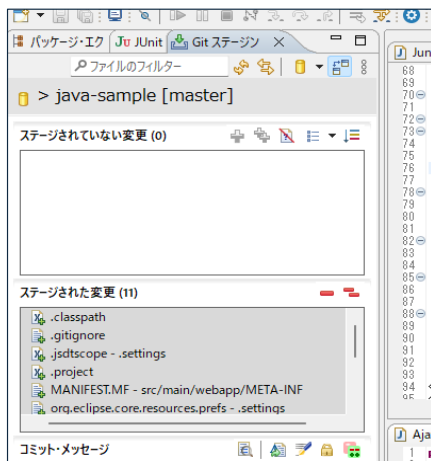
→2回目以降は全件でも個別でもOK

個別のファイルのみaddしたい場合は以下の通り

【個別add】 ファイルを選択>「+」をクリック



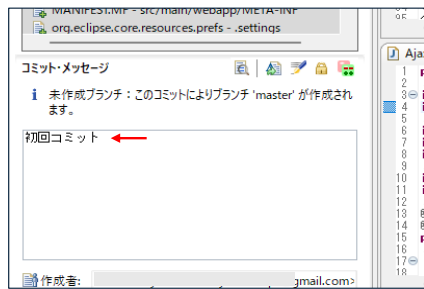
☆addすると、ファイル名が「ステージされた変更」に移動されるのが確認できる



□「コミットメッセージ」を入力する

→※初回の場合は「初回コミット」

→2回目以降は「単元名」など、適切な内容。記録として残せるタイトルになるので、分かりやすく記載する

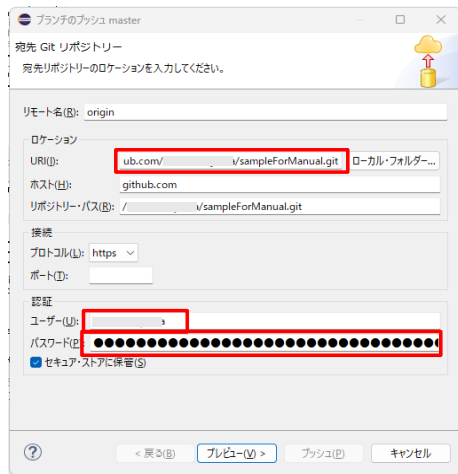


□コミットおよびプッシュ

- ・「コミット及びプッシュ」をクリック ※初回でエラーが出た場合は続けて「HEADのプッシュ」をクリックする

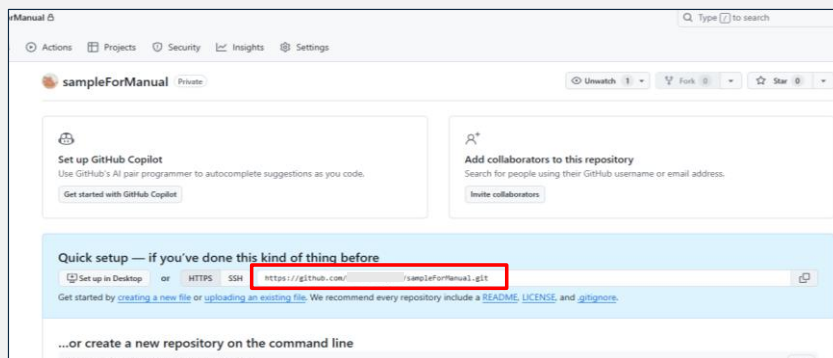
→※初回の場合

「宛先Gitリポジトリ」が表示される



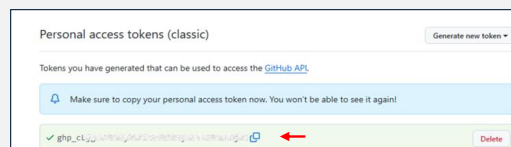
上記の赤枠の欄を入力する

- ・URI →GitHubのリモートリポジトリのURIをコピーして貼り付け



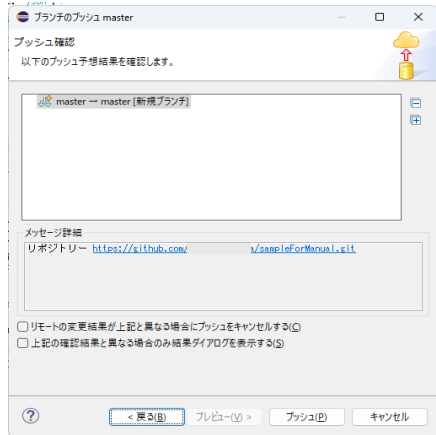
- ・ユーザー →GitHubユーザー名

- ・パスワード →控えておいた「パーソナルアクセストークン」



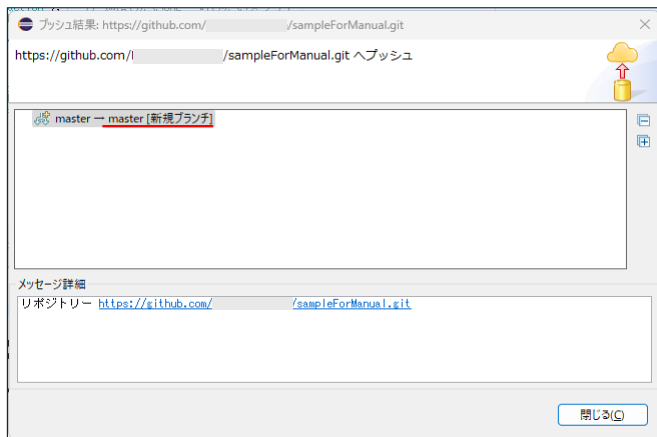
↓

入力できたら、「次へ」または「プレビュー」で進めていき、下記のような表示になったら「プッシュ」をクリック

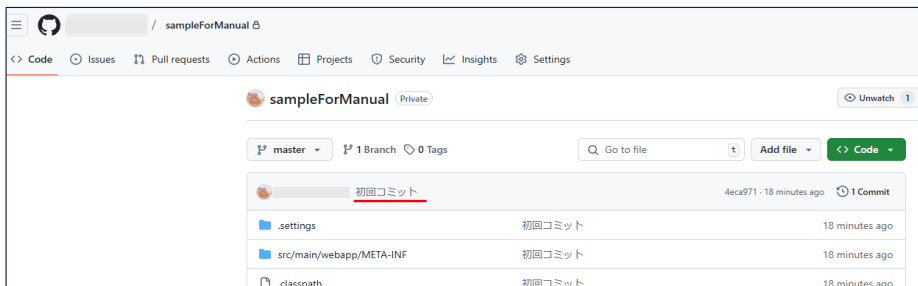


↓

成功の場合はこのような表示



GitHubのリポジトリを更新すると、以下のような画面
赤線には最新のコミットメッセージが表示されます



→2回目以降の場合

下記のように表示されていればプッシュ成功

赤線：コミットのハッシュ値（数字＋英字の文字列）とコミットメッセージ

