

MIP(Miyazi ishikawa Project)

EclipseにおけるGitの使い方

2018 年 5 月 21 日

概要

EclipseにおけるGitの使い方について説明する。

目標

1. **Gitの操作ミス0:** このドキュメントを見ることで、Eclipse上でのGit操作ミスを0にする
2. **プロジェクト管理の規則を明確化:** このドキュメントによって、プロジェクト管理の運用規則がわかる

EclipseでGitを使う

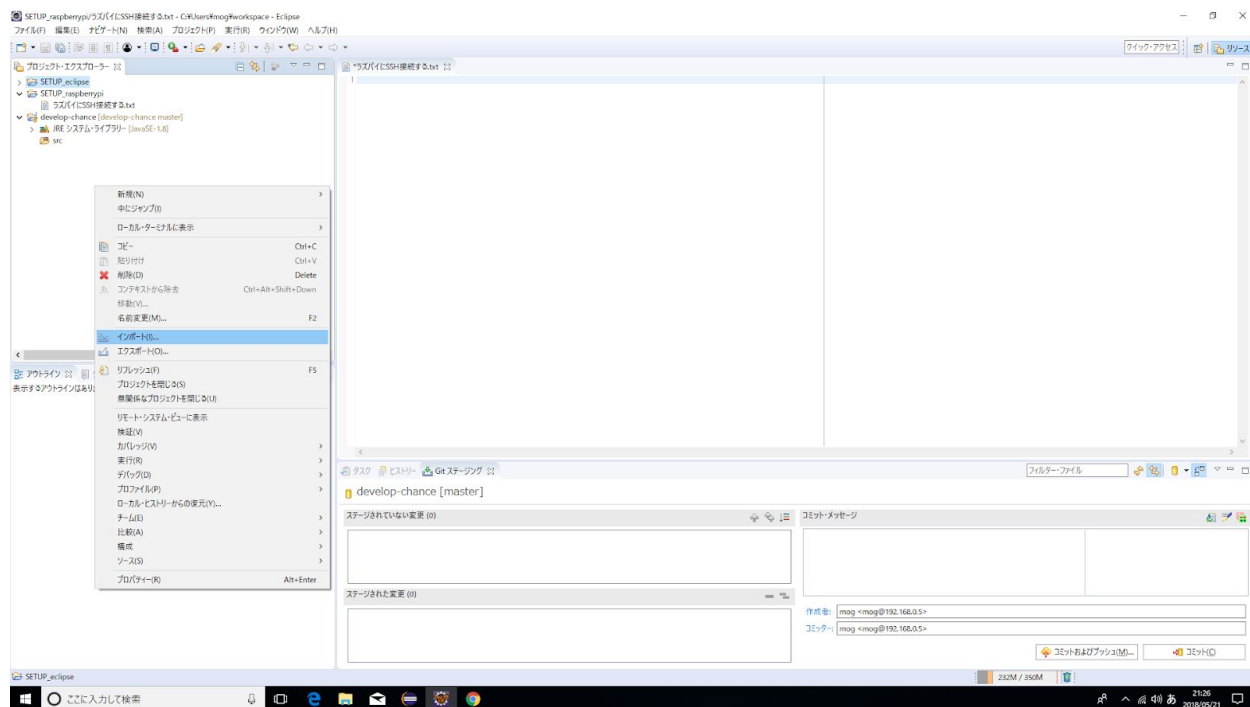
EclipseにGitを導入する方法が示されたサイトは以下。

URL:<https://git-manual.net/egit-eclipse-install/>

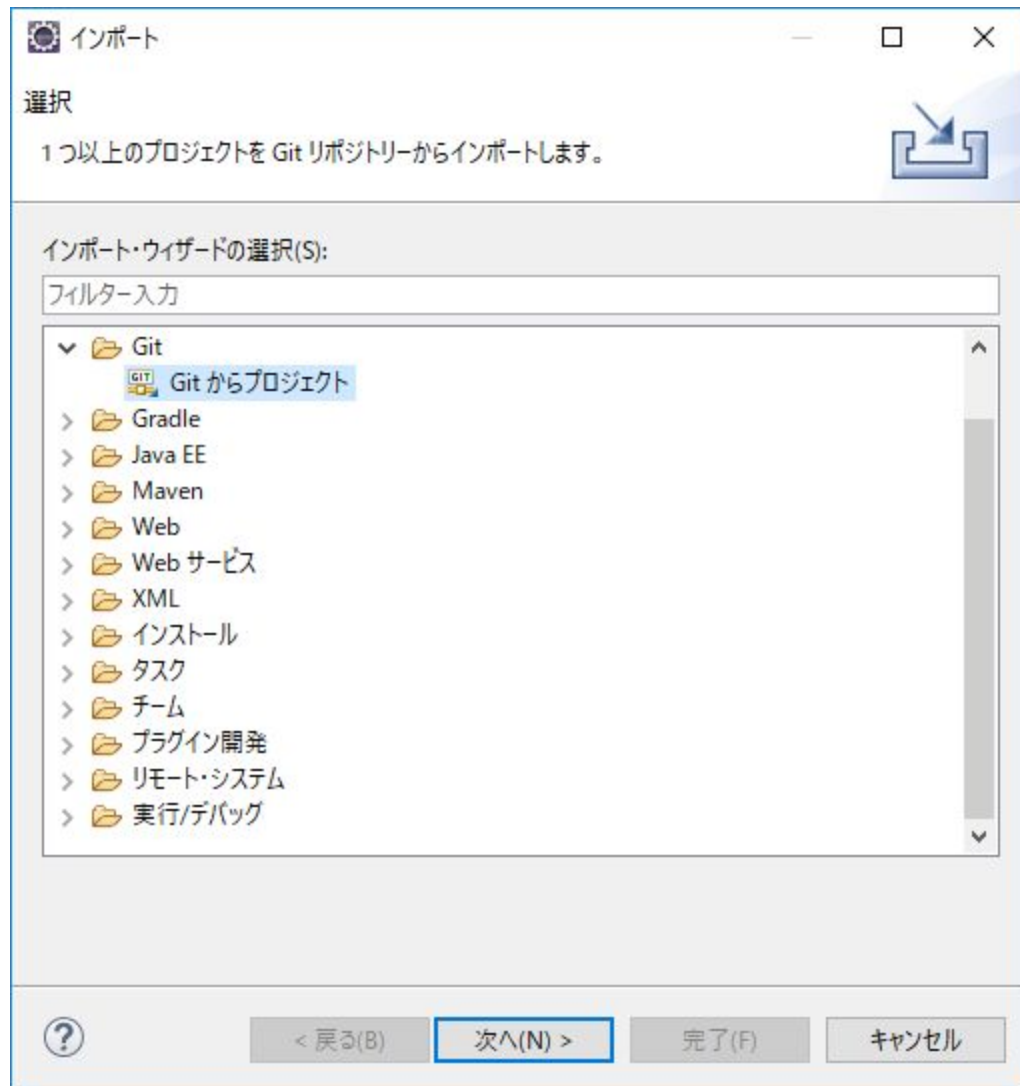
MogulはデフォルトでGitが入っているFull versionのEclipseをインストールしたため、この作業を行っていない。

Gitリポジトリをクローン

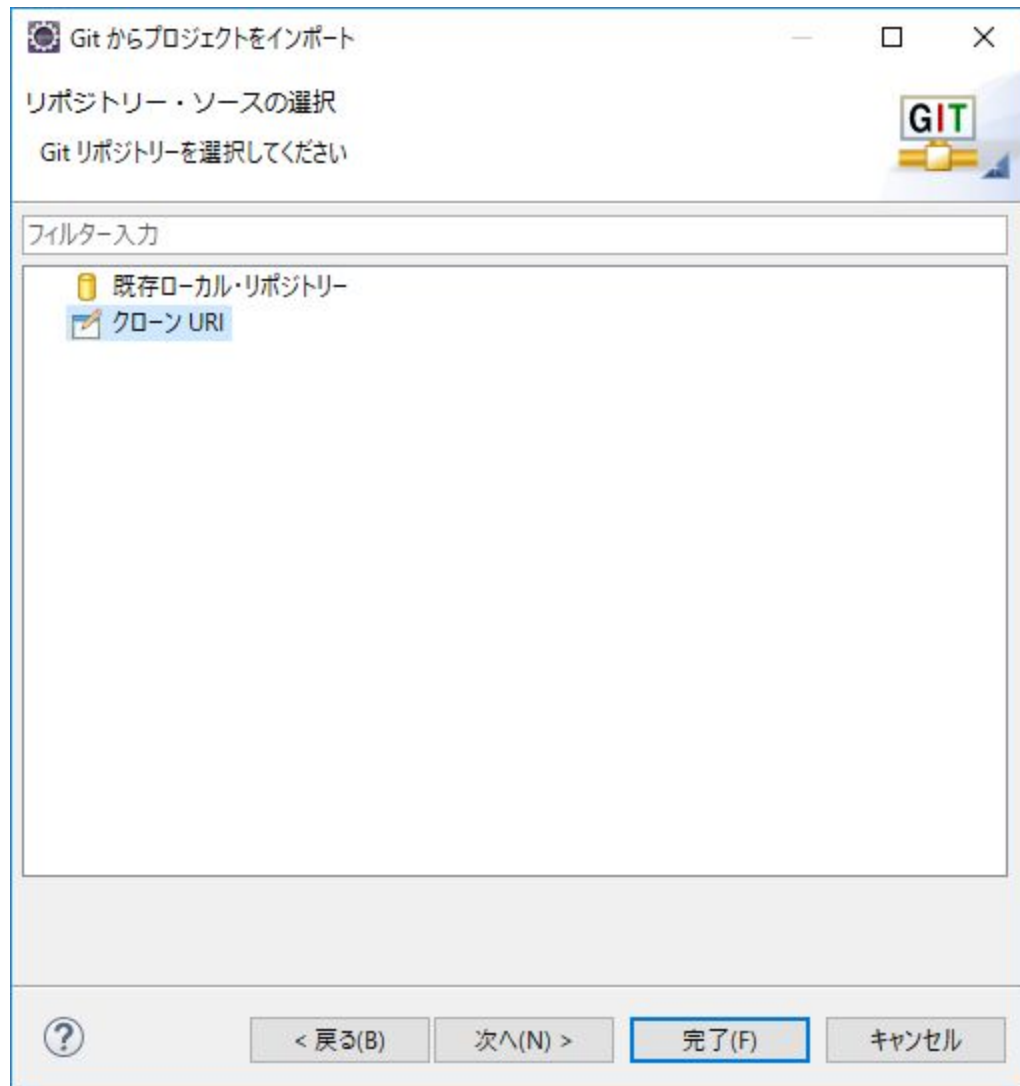
1. 右クリック>インポート



2. Gitからプロジェクト>次へ



3. クローンURI>次へ



4. リポジトリ情報入力>完了



Git からプロジェクトをインポート

ソース Git リポジトリ

ソース・リポジトリのロケーションを入力します。

ロケーション

URI(I): thub.com/MoguMoguMog/develop-chance.git ローカル・ファイル...

ホスト(H): github.com

リポジトリ・パス(R): /MoguMoguMog/develop-chance.git

接続

プロトコル(L): https

ポート(T):

認証

ユーザー(U): MoguMoguMog

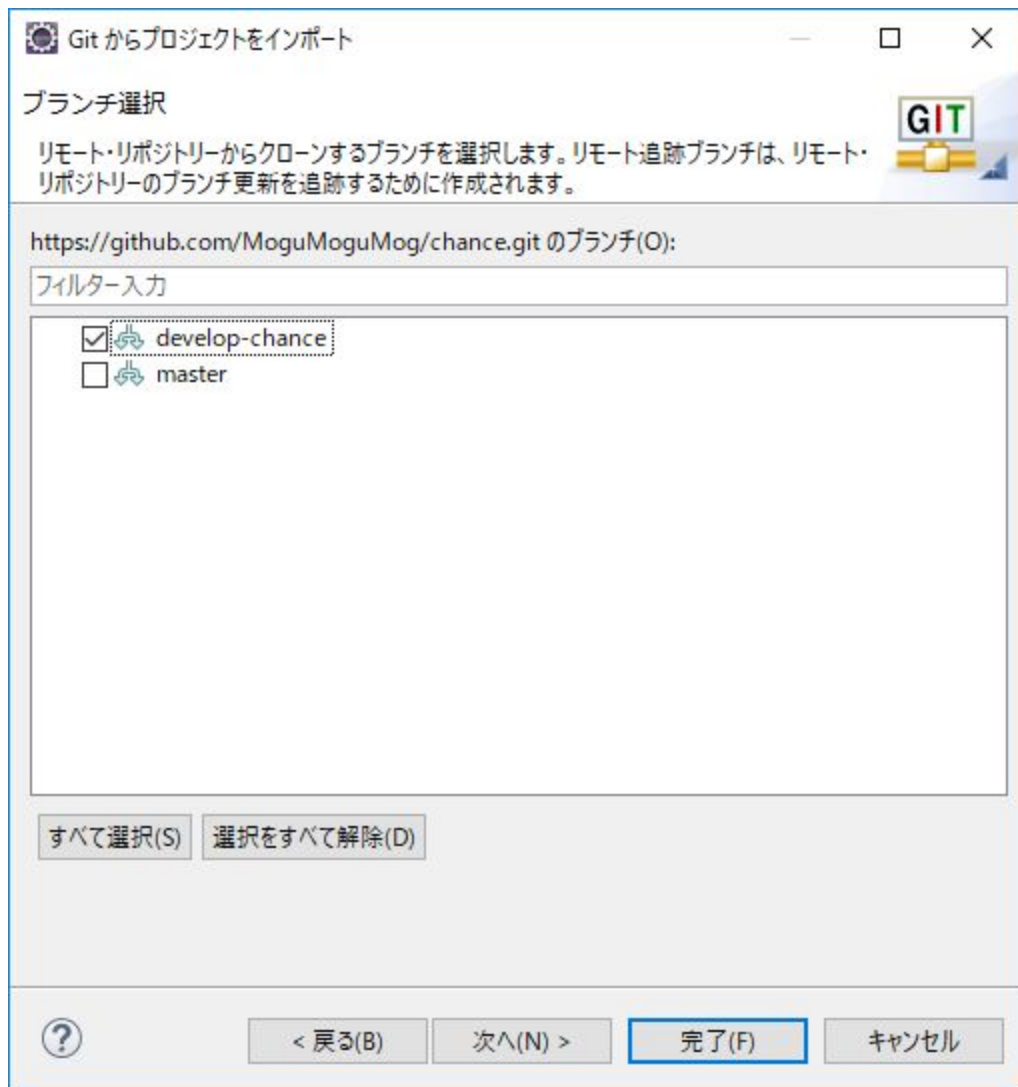
パスワード(P):

☒ セキュア・ストアに保管(S)

? < 戻る(B) 次へ(N) > 完了(F) キャンセル

- ・ URI -> https://github.com/MoguMoguMog/*.git (*:任意の名前)
- ・ ホスト、リポジトリパス->自動で入力されるはず
- ・ 認証:githubのユーザ名とパスワード

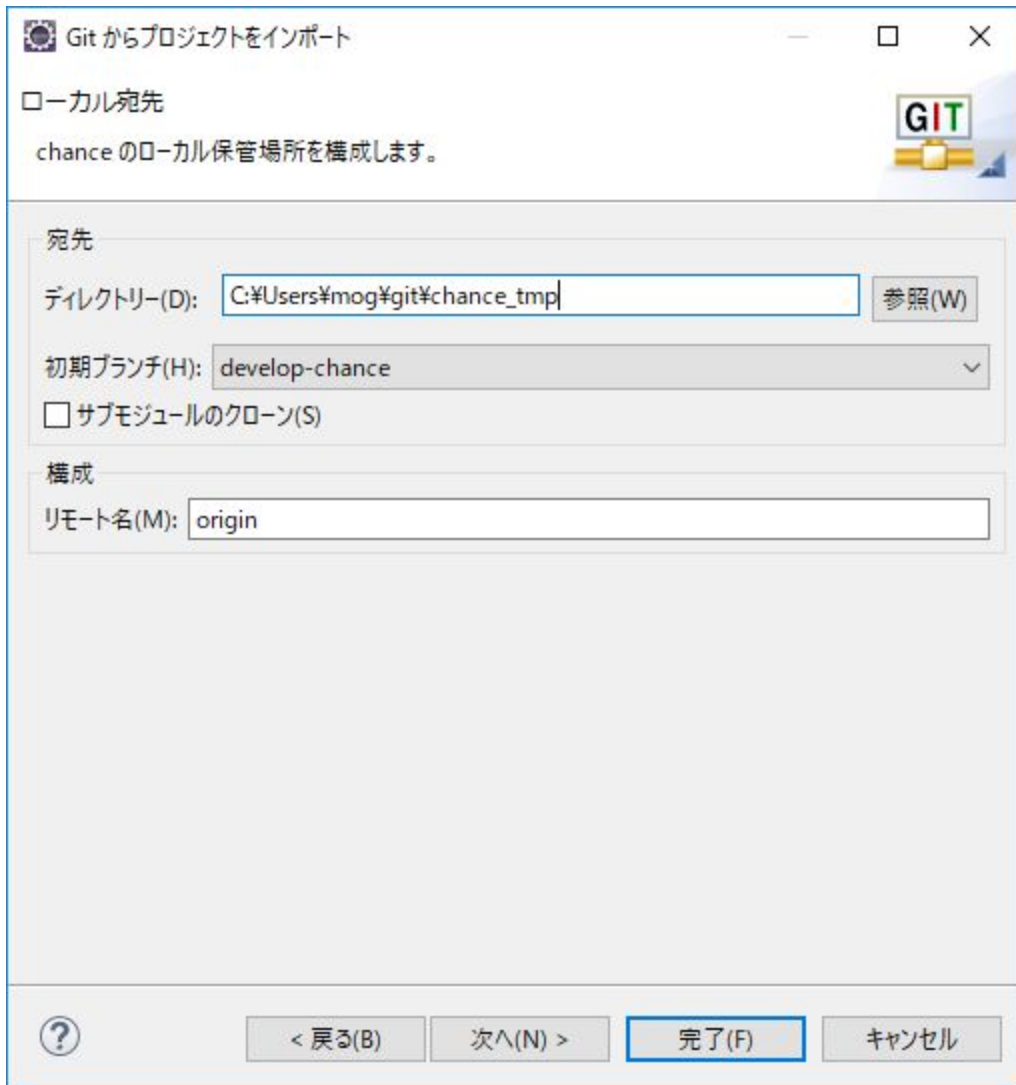
5. クローンするブランチを選択>次へ



masterブランチの派生がdevelop-*となっている。develop-*ブランチを派生させていく予定。
masterブランチは基本的に修正を加えない。修正を加えるのは石川がやる。

よって、通常masterをインポートすることはないのでチェックを外す。

6. 次へ



Git からプロジェクトをインポート

ローカル宛先

chance のローカル保管場所を構成します。

宛先

ディレクトリ(D): C:\Users\mog\git\chance_tmp 参照(W)

初期ブランチ(H): develop-chance

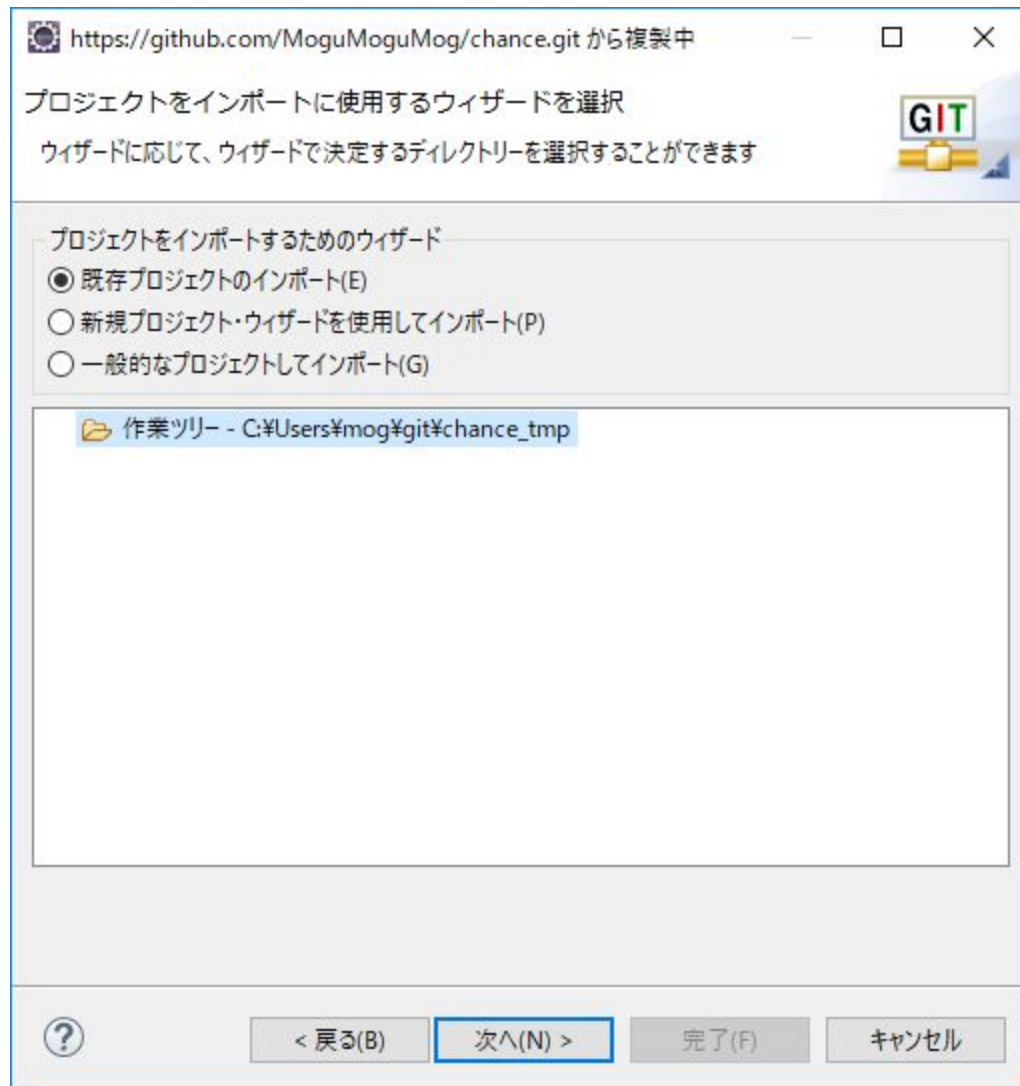
☐ サブモジュールのクローン(S)

構成

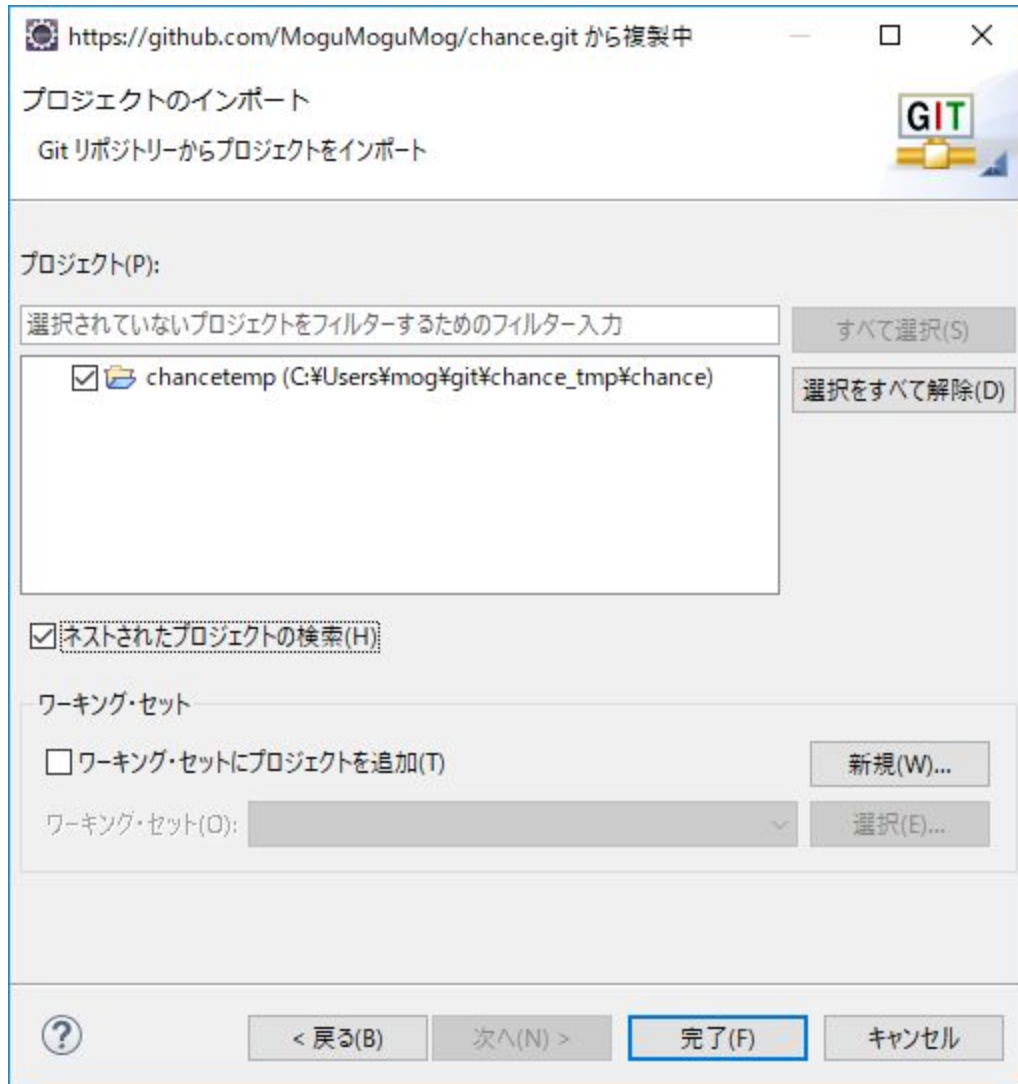
リモート名(M): origin

? < 戻る(B) 次へ(N) > 完了(F) キャンセル

7. 既存プロジェクトのインポート>次へ



8. 完了

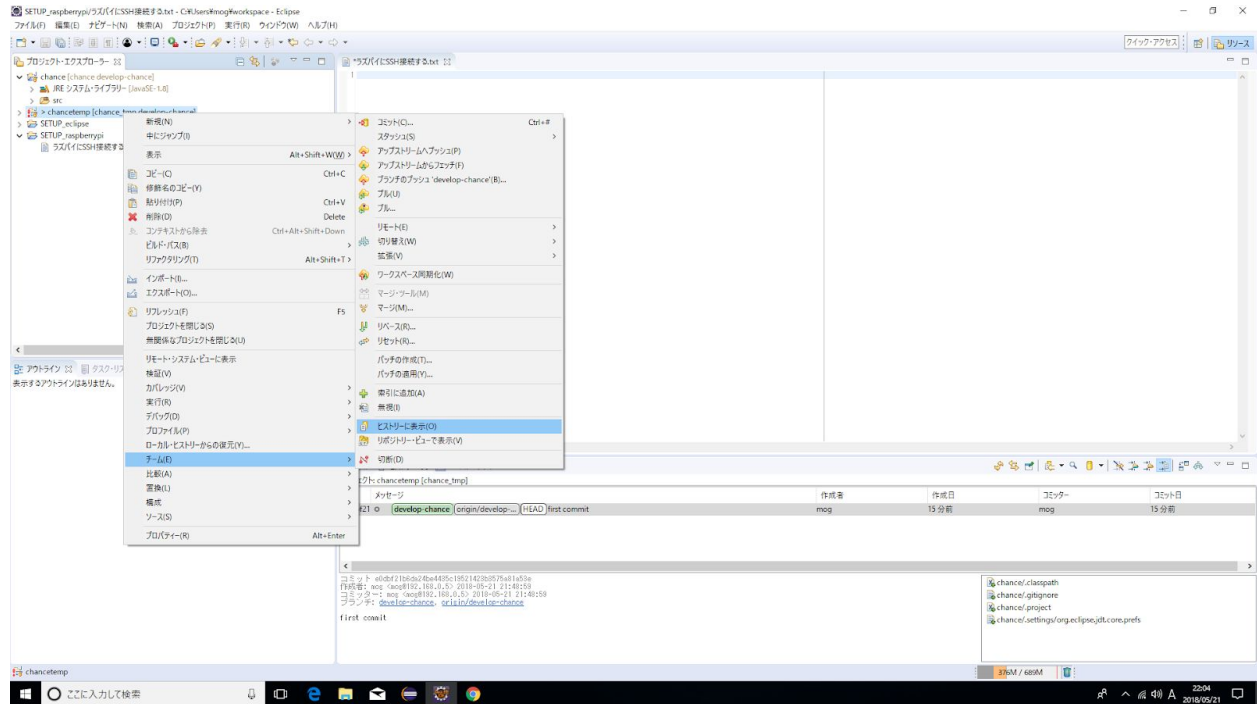


これでdevelop-*ブランチのインポートが完了した。

ブランチの作成

上記手順でdevelop-*ブランチをクローンした。しかし、develop-*に直接変更は加えない。develop-*ブランチからブランチを派生させ、そこに変更を加えていく。最後にすべての変更をdevelop-*ブランチにマージする。**develop-*ブランチにマージする作業は石川が行う。**

1. 複製したdevelop-*ブランチを右クリック>チーム>ヒストリーに表示



2. ヒストリーのdevelop-*ブランチを右クリック>チェックアウト>新規ローカルブランチとしてチェックアウト
3. ブランチ名を設定して完了

作成したブランチ上で変更を加え、コミットすることでローカルリポジトリに記憶される。ブランチをプッシュすることでgithub上に作成したブランチに変更が加わる。これはGitの特性である。（コミットしただけだとリモートリポジトリには変更反映されない）

ひとまずは以上。