# **RedmineでリポジトリとGitを連携方法**

**Redmineでリポジトリを作成する**

Redmineの画⾯からプロジェクトごとにリポジトリを作成したり、削除することができる。

### **Redmineでリポジトリの作成**

### 1. リポジトリタブの表⽰

プロジェクトの管理者でログインし、以下の画⾯を表⽰する。

プロジェクトの[設定]タブ **>** [リポジトリ]タブ



2. 新しいリポジトリ画⾯の表⽰

[新しいリポジトリ]をクリックする

### 3. リポジトリの作成

以下のとおり⼊⼒し、[作成]ボタンをクリックする。項⽬ 値

|  |  |
| --- | --- |
| バージョン管理システム | Gitを選択 |
| 新規リポジトリの作成 | ON |
| メインリポジトリ | 任意 |
| 識別⼦ | リポジトリを識別する名前  プロジェクト内で1つのリポジトリしか利⽤しない場合は⼊⼒不要 |
| パスのエンコーディング | Gitのみ  デフォルト値は「UTF-8」 |

ファイルとディレクトリの最新コミットを表⽰

Gitのみする

識別⼦を指定した場合、リポジトリ名は プロジェクトの識別⼦.リポジトリの識別⼦ となる。 例えば、プロジェクトの識別⼦を gui-g20c001-1-5、識別⼦を la8000\_gui\_testとした場合、リポジトリ名は gui-g20c001-1-5.la8000\_gui\_testとなる。

## **リポジトリの削除**

### 1. リポジトリタブの表⽰

プロジェクトの管理者でログインし、以下の画⾯を表⽰する。

プロジェクトの[設定]タブ > [リポジトリ]タブ

### 2. リポジトリの削除

削除するリポジトリの[削除]をクリックする。 外部リポジトリと連携している場合は、Redmineと連携するための設定だけが削除される。 「リポジトリの作成」の⼿順で作成したリポジトリは、設定だけでなく、リポジトリそのものも削除される。



## **Redmineで作成したリポジトリにアクセスする⽅法**

### **TortoiseGitをインストール、ローカルリポジトリ作成**

以下、Windowsを前提として記載する。

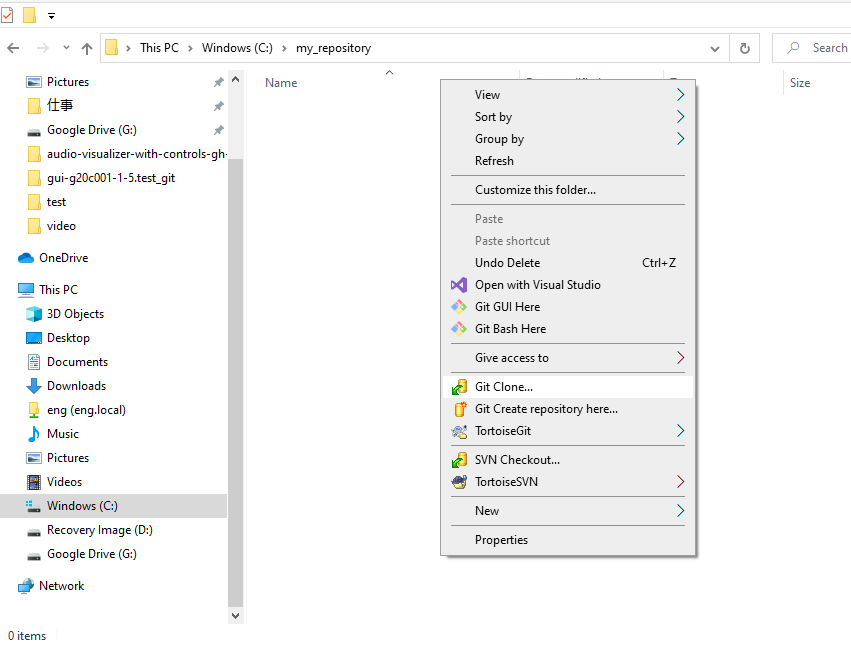
Link Dowload TortoiseGit: https://tortoisegit.org/download/

1. リポジトリを格納するフォルダを作成する

(C:)にリポジトリを格納するフォルダを作成する。

1. リポジトリをクローンする

. クローンダイアログの表⽰

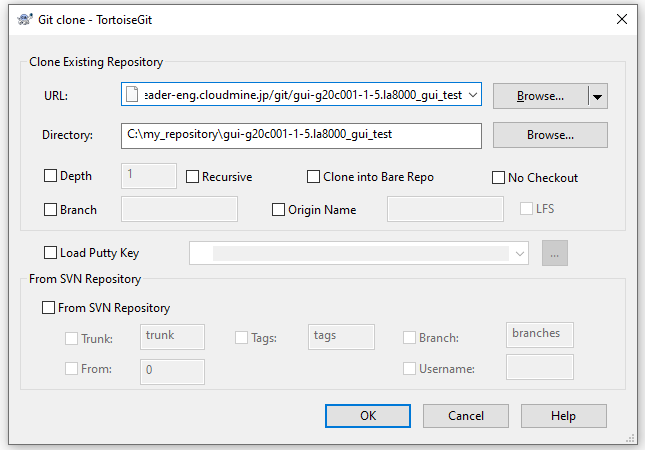
「１」で作成したフォルダをエクスプローラーで開く。 フォルダ内の任意の場所で右クリックし、[Git クローン]を選択する。

. クローン以下の情報を⼊⼒し、[OK]ボタンをクリックする。

項⽬ 値

|  |  |
| --- | --- |
| リポジトリのURL | https://サブドメイン.cloudmine.jp/git/プロジェクトの識別⼦.リポジトリの識別⼦ |
| チェックアウト先のディレクトリ | エクスプローラーで表⽰しているフォルダ |
| チェックアウトする深さ | 再帰的(デフォルト) |
| 外部参照を除外する | OFF(デフォルト) |

**例**：

 項⽬ 値

|  |  |
| --- | --- |
| サブドメイン | leader-eng |
| プロジェクトの識別⼦ | gui-g20c001-1-5 |
| リポジトリの識別⼦ | la8000\_gui\_test |
| クローン先のディレクトリ | https://leader-eng.cloudmine.jp/git/gui-g20c001-1-5.la8000\_gui\_test |
| クローンする深さ | デフォルト |

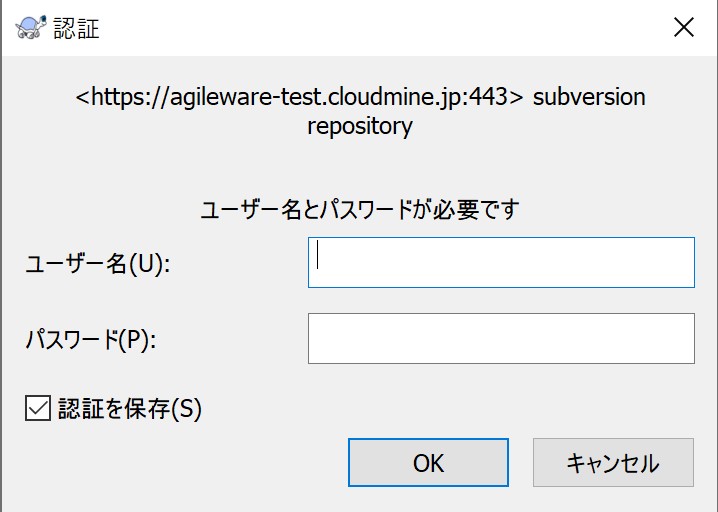
. 認証

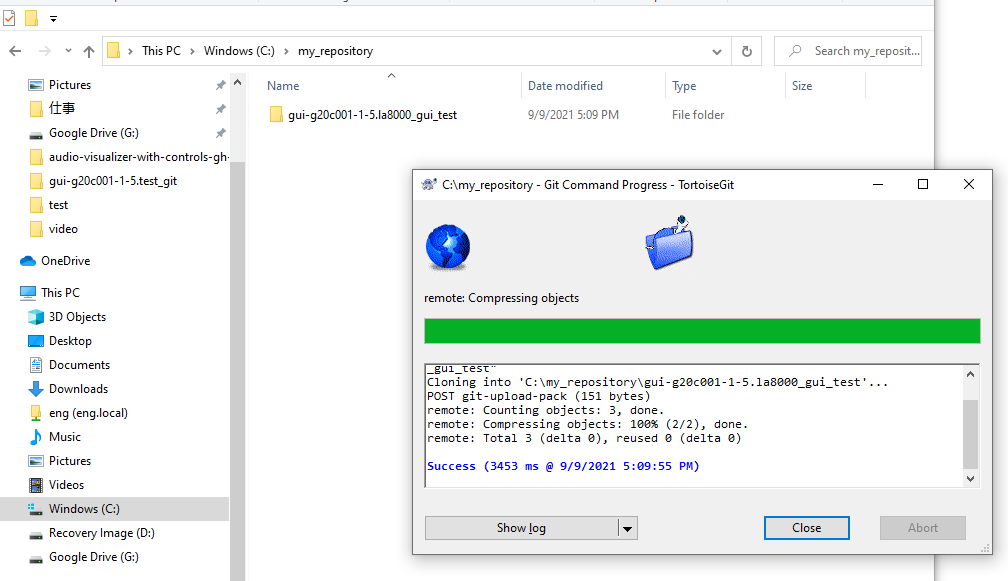
以下の情報を⼊⼒し、[OK]ボタンをクリックする。

項⽬ 値

|  |  |
| --- | --- |
| ユーザー名 | RedmineのログインID |
| パスワード | Redmineにログインする時のパスワード |

認証を保存 ON(デフォルト)

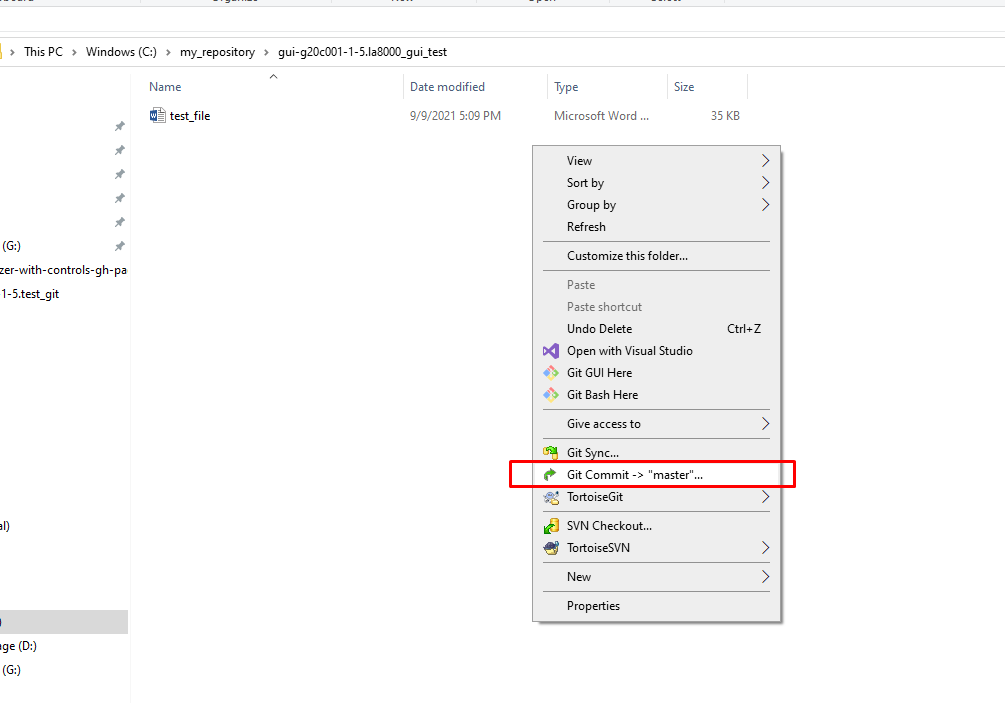
クローンの仕⽅は以下の通り。 リポジトリにアクセスした時に⼊⼒する認証情報は、Redmineの情報を⼊⼒する。**

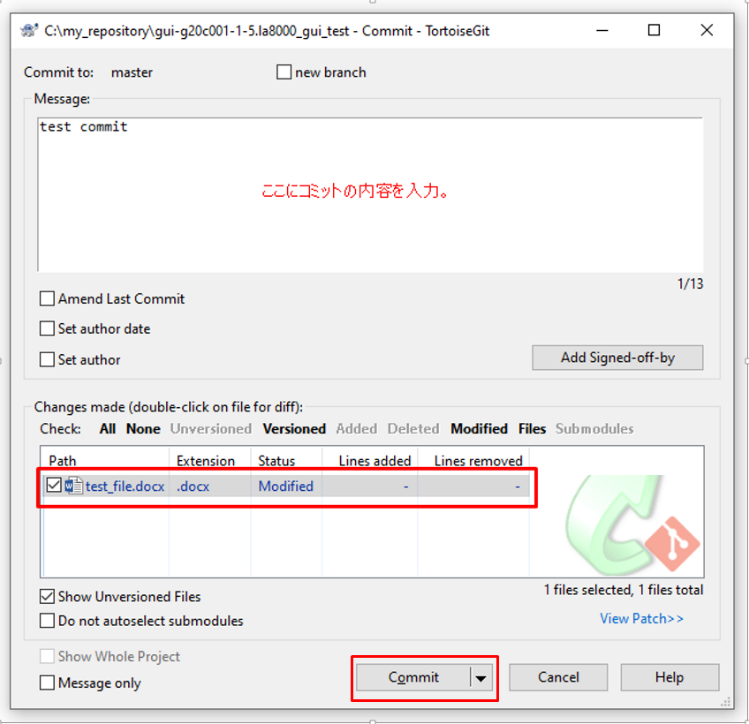
クローン出来たら、(C:)での作成したローカルリポジトリフォルダに、Redmineで作成したリモートリポジトリと同じを表示されます。

## **Gitのコミット（Commit）**

ローカルリポジトリに、ファイル作りたり、ファイル変更たり、Gitにコミット必要です。

コミット方法：ローカルリポジトリ内の任意の場所で右クリックし、[Git コミット]を選択する。

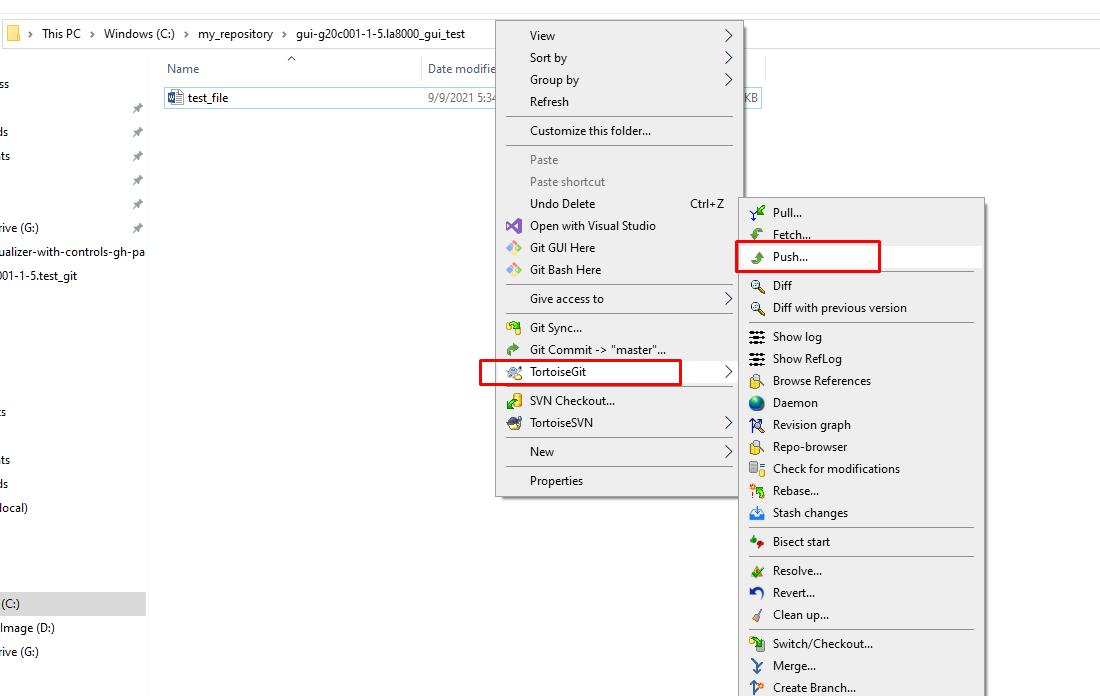


コミットの内容を入力してファイルを選択して、コミットボタンを選択。

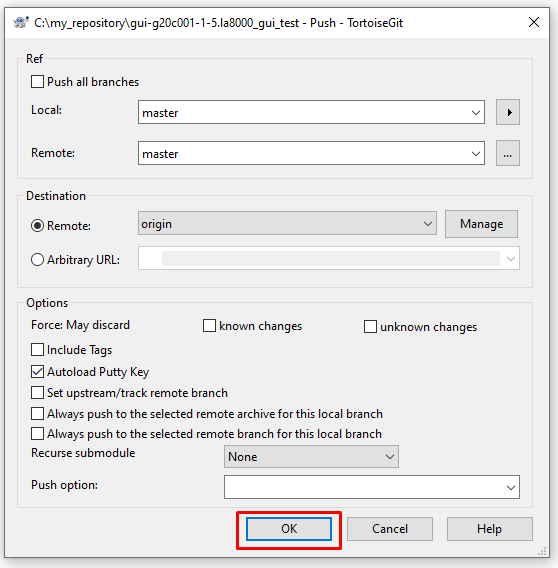
## **Gitのプッシュ(Push)**

リモートリポジトリで自分の手元のローカルリポジトリの変更履歴を共有するには、ローカルリポジトリ内の変更履歴をアップロードする必要があります。

プッシュ方法：ローカルリポジトリ内の任意の場所で右クリックし、[**TortoiseGit**]を選択して、表示するMenu内でプッシュ(Push)を選択。



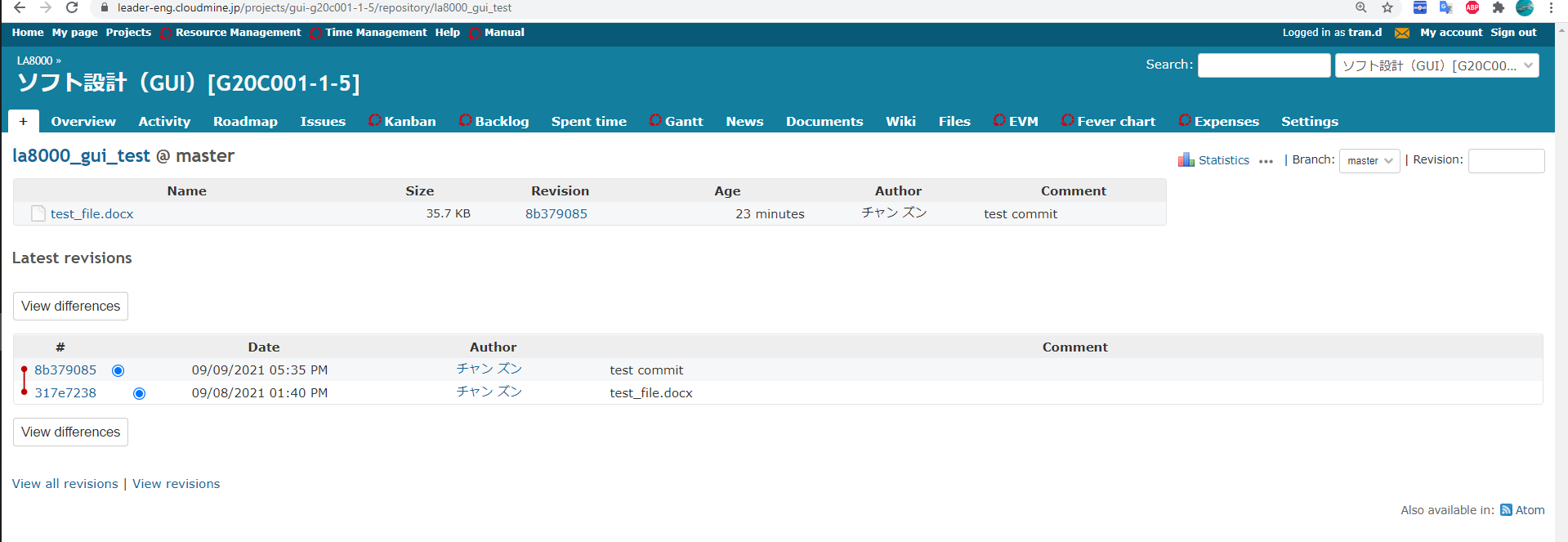
プッシュ選択したら、以下のような画面を表示する、そのままでOKボタンをクリックする。



## **リモートリポジトリからコミットしたリポジトリのログをチェック**

Redmineでプロジェクトの[設定]タブ > [リポジトリ]タブ　をアクセス

作ったリポジトリ名をクリックすると、以下のような画面を表示、ログをみえます。



## Git Pull

git pullは、リモート リポジトリからコンテンツをフェッチしてダウンロードし、そのコンテンツと一致するようローカル リポジトリを即時に更新するために使用されます。

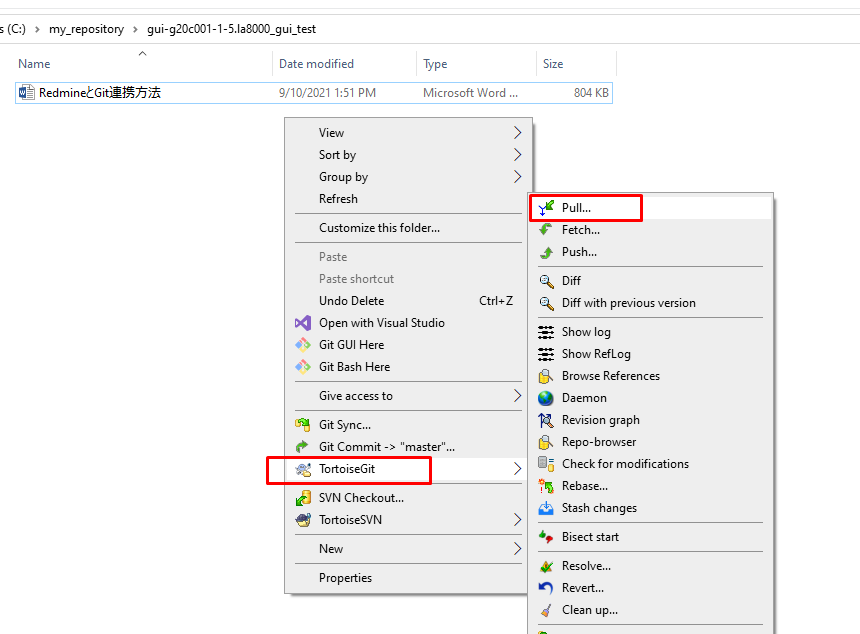
例：

User１のロカールリポジトリはコンテンツが変更して、コミットしてリモートリポジトリにPushする、

User2のロカールリポジトリはコンテンツがリモートリポジトリのコンテンツと一致するようにPullをします。

**やり方：**　ローカルリポジトリ内の任意の場所で右クリックし、[**TortoiseGit**]を選択して、表示するMenu内で

プル (Pull)を選択したら、Pullダイアログを表示する、OKボタンをクリックするとPull出来ます。



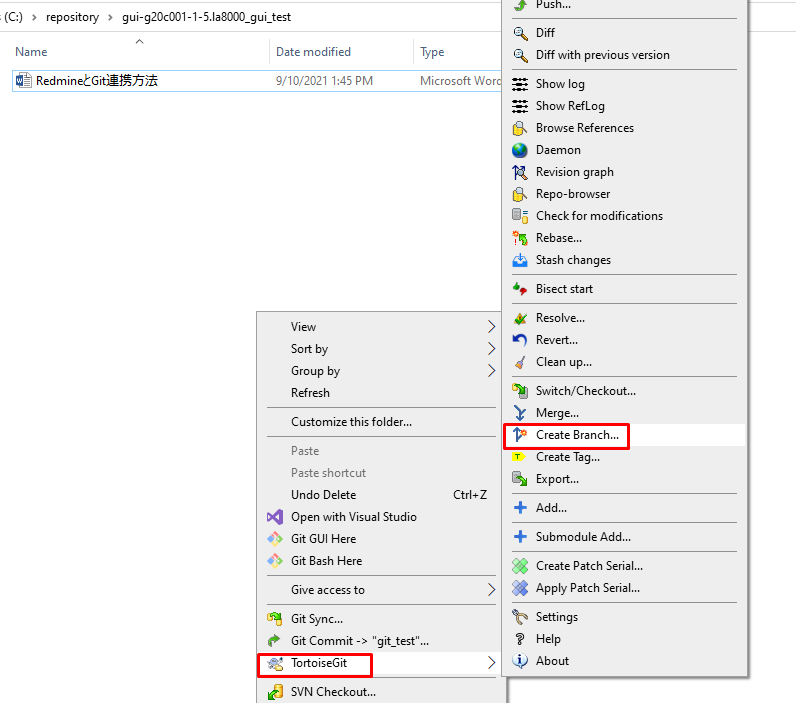
## **Git Branch 作り、消す方法**

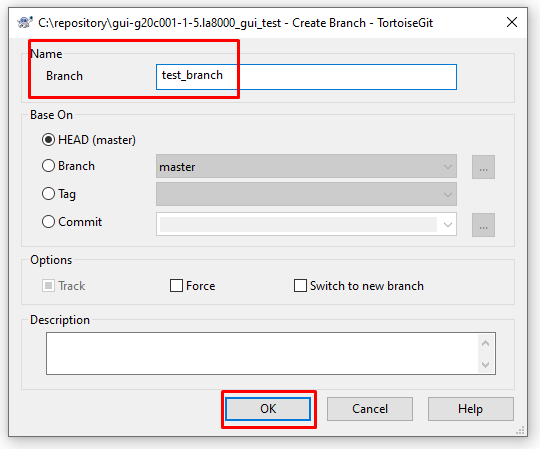
ブランチとは、履歴の流れを分岐して記録していくためのものです。分岐したブランチは他のブランチの影響を受けないため、同じリポジトリ中で複数の変更を同時に進めていくことができます。

**Branchを作り方法：**

ローカルリポジトリ内の任意の場所で右クリックし、[**TortoiseGit**]を選択して、表示するMenu内で

Create Branchを選択したら、Create Branchダイアログを表示する, Branch 名を入力してOKボタンをクリックすると新しいBranchを作成します。





### **Branchを消す方法：**

右クリックし、[**TortoiseGit**]を選択して、表示するMenu内で

Create Branchを選択したら、Browse referenceダイアログを表示する, 消すしたいBranch名で右クリックしてDelete branchをクリックするとBranchを消します。

