**TortoiseGitのhttp(s)通信**

**および**

**パスワードの記憶と破棄について**

2013年7月19日　初版

改訂履歴

|  |  |  |  |
| --- | --- | --- | --- |
| 版 | リリース | 担当 | 改訂内容 |
| 初版 | 2013年7月19日 | 板垣　衛 | (初版) |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

* 目次

[ 本書が扱うソフトウェアとバージョン 1](#_Toc361376734)

[ Gitリポジトリのクローン（http(s）通信) 1](#_Toc361376735)

[ ユーザーIDとパスワードを記憶する仕組み：wincredについて 3](#_Toc361376736)

[ 記憶しているユーザーIDとパスワードを破棄 3](#_Toc361376737)

[ Gitリポジトリと作業ツリーの新規作成 4](#_Toc361376738)

[ Gitリポジトリの同期先の追加（http(s)通信） 5](#_Toc361376739)

[ Bareリポジトリの新規作成（方法①）：GUI操作 7](#_Toc361376740)

[ Bareリポジトリの新規作成（方法②）：コマンドライン操作 8](#_Toc361376741)

# 本書が扱うソフトウェアとバージョン

・msysGit … Ver.1.8.3.msysgit.0

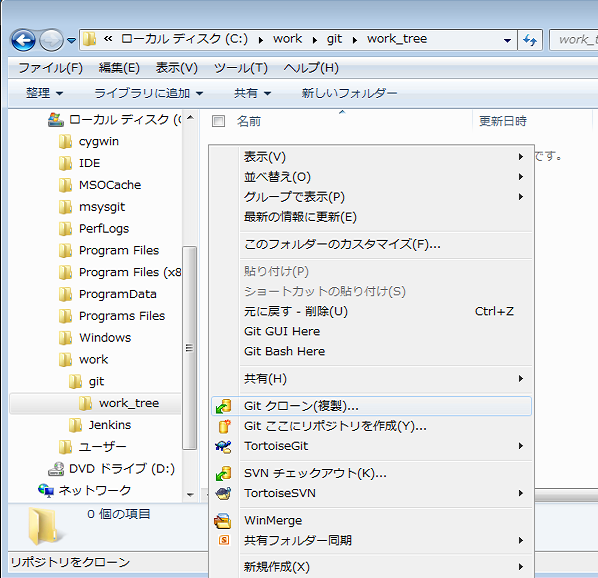
・TortoiseGit … Ver.1.8.3.0 – 64 Bit

# Gitリポジトリのクローン（http(s）通信)

Gitの共有リポジトリが既に存在する場合、かつ、そのサーバーとの通信がhttp通信もしくはhttps通信に対応している場合、下記の手順でリポジトリをクローンし、ローカルPCに作業ツリーとローカルリポジトリを作成する。

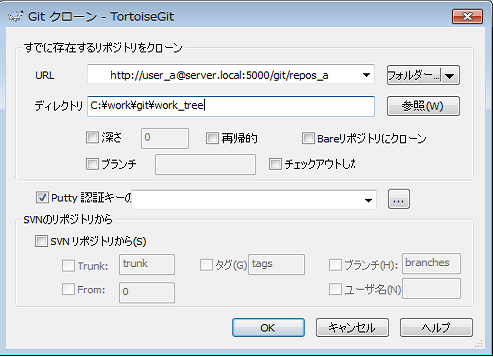
なお、できる限りhttp通信は用いない。社内での利用に留めるなら問題がないが、社外と連携するならhttp通信はNG。https通信を用いる。また、サーバーがSSH通信に対応しているなら、https通信よりも極力SSH通信を使用する。

**手順①：** エクスプローラーで右クリックして、コンテキストメニューから「Gitクローン(複製)…」を実行。



**手順②：** 「Gitクローン」のダイアログにて、共有リポジトリのURLと作業ツリー＆リポジトリを作成するローカルのディレクトリを指定する。

「Putty認証キーのロード」のチェックがデフォルトでONになっているのでOFFにする。これはSSH通信時にのみ必要な設定。他はデフォルトのまま。



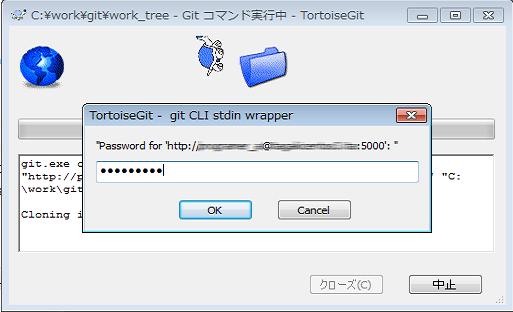
「Bareリポジトリにクローン」のチェックはONにすると、ローカルリポジトリだけ作られて作業ツリーが作られないので注意。

クローン元のURLには、http:// もしくは、https:// で始まるパスを指定する。

サーバー名の前に「ユーザーID@」を付けると、共有リポジトリアクセス時にユーザーIDの入力を省略できる。（URL例：<http://user_id@server.local/dir/>）

事前に、「[02]TortoiseGitセットアップ手順」に基づいて、https通信の証明書の認証チェック無効化設定と、ユーザーIDとパスワードの自動保存設定を行っておく事。

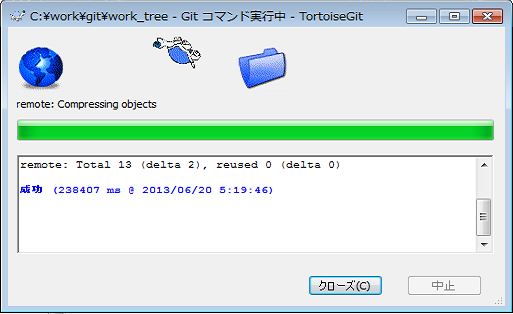
**手順③：** ［OK］ボタンを押してクローンを開始すると、ユーザーのパスワードを入力するダイアログボックスが表示されるので、パスワードを入力する。

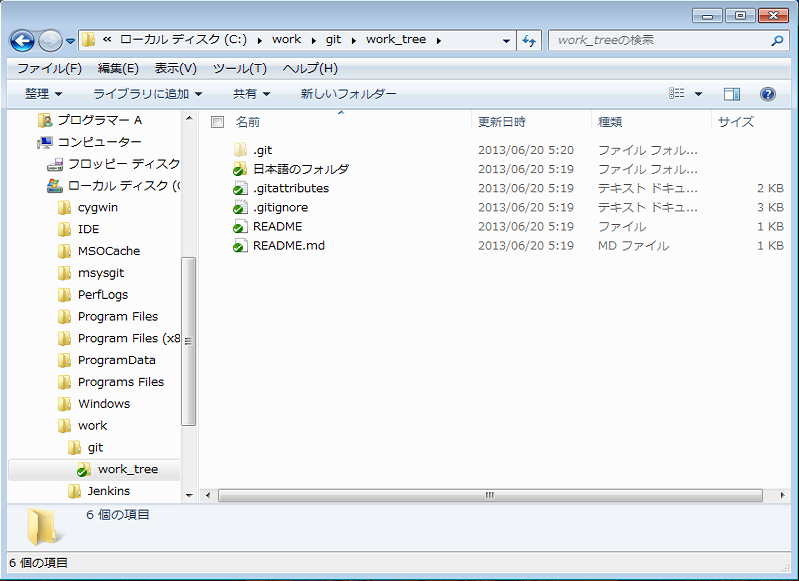


「手順②」にて、URLに「ユーザーID@」を付けていない場合、この場面でユーザーIDとパスワードをそれぞれ入力するように求められる。

また、サーバー側で認証を求めるように設定されていない場合、このダイアログは表示されない。

**手順④：** クローン完了。





クローンが完了すると、エクスプローラーに作業ツリー（マーク付きのフォルダ・アイコン）とローカルリポジトリ（.gitフォルダ）が作成される。

空っぽのリポジトリをクローンした場合は、.git フォルダが作成されるだけとなる。

この後の共有リポジトリとの同期操作（プッシュ、プル、フェッチ、リベース）については別紙参照。

# ユーザーIDとパスワードを記憶する仕組み：wincredについて

別紙の「[02]TortoiseGitセットアップ手順」にて、「■TortoiseGitの初期設定」の所で、Gitの「資格情報ヘルパー」として「wincred」を指定する方法を説明している。

これはTortoiseGit1.8.1以降に標準実装されている仕組みだが、これを使用すると、TortoiseSVNのように、認証したユーザーIDとパスワードを自動的に保存し、次回以降の共有リポジトリへのアクセスではユーザーIDとパスワードの入力を省略する事ができる。

なお、例えばActiveDirectory統合認証でアクセスするリポジトリがあり、一度アクセスした後にActiveDirectoryの方でユーザーのパスワードを変更するなどすると、TortoiseGitが記憶しているパスワードでユーザー認証ができなくなってしまう。

そのような場合、記憶しているパスワードをリセットする必要があるが、その方法については後述する。

# 記憶しているユーザーIDとパスワードを破棄

別紙の「[02]TortoiseGitセットアップ手順」に基づいて、ユーザーIDとパスワードの自動保存設定を行った場合、現時点では、それを破棄する手段が用意されておらず、パスワードを変更した場合などにパスワードを入力し直す事ができず、接続に失敗するようになってしまう。

その為、下記のツールを自作し、Git向けに記憶しているユーザーとパスワードを削除できるようにしている。

[Gitマニュアル・調査資料]

`-[Tools]

`-[git\_erase\_wincred\_all]

|-[32bit]

| `- git\_erase\_wincred\_all.exe

`-[64bit]

`- git\_erase\_wincred\_all.exe

32bit版と64bit版のそれぞれを用意しており、使用方法はどちらかOSの環境に合ったツールを実行するだけ。

乱暴なツールである為、実行確認もなにもなく、実行すると即座にGit用に記憶している全ユーザーとパスワードを削除する。

コマンドプロンプトに削除したURLとユーザーIDをリストアップしながら実行する。

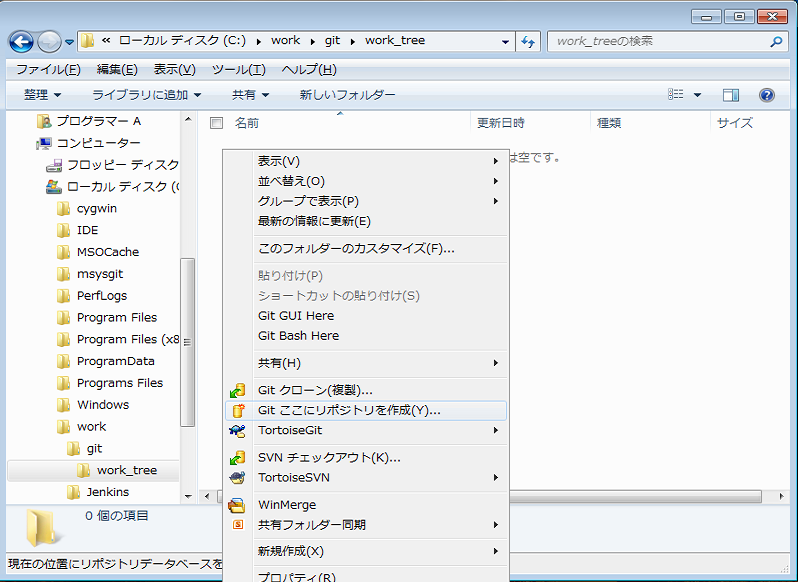
なお、TortoiseGitの設定画面には、TortoiseSVNと同様の、［保存されたデータ］→［認証データ］→［クリア］というボタンがあるが、TortoiseSVNと異なり、この操作を行ってもwincredで記録されたユーザー情報は削除されない。

# Gitリポジトリと作業ツリーの新規作成

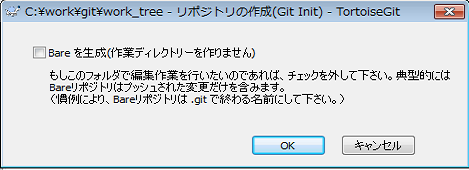
Gitの共有リポジトリがまだ存在しない場合、下記の手順でローカルPCに新規の作業ツリーとローカルリポジトリを作成する。

また、先にローカルリポジトリを作成してから既存の共有リポジトリと同期を取っても問題ない。

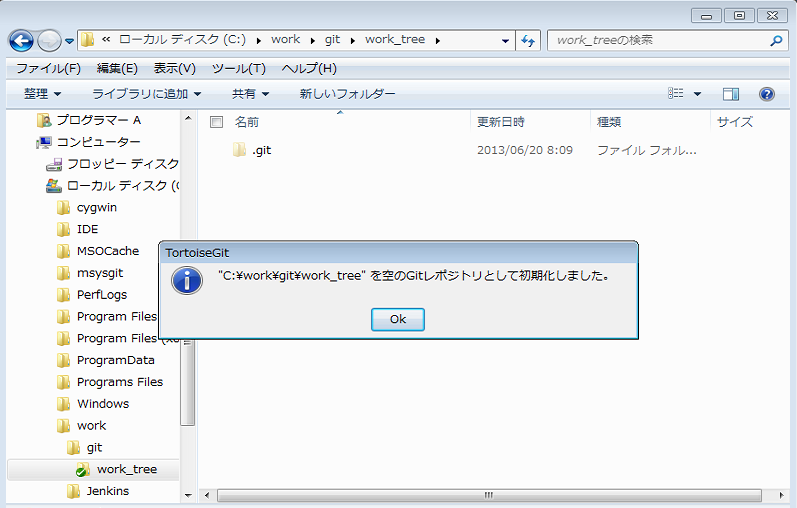
**手順①：** エクスプローラーで右クリックして、コンテキストメニューから「Gitここにリポジトリを作成…」を実行。



**手順②：** 「リポジトリの作成（Git Init）」のダイアログにて、「Bare を生成（作業ディレクトリーをつくりません）」のチェックがOFFのまま（デフォルトのまま）、［OK］ボタンを押す。



**手順③：** リポジトリと作業ツリーの新規作成完了。



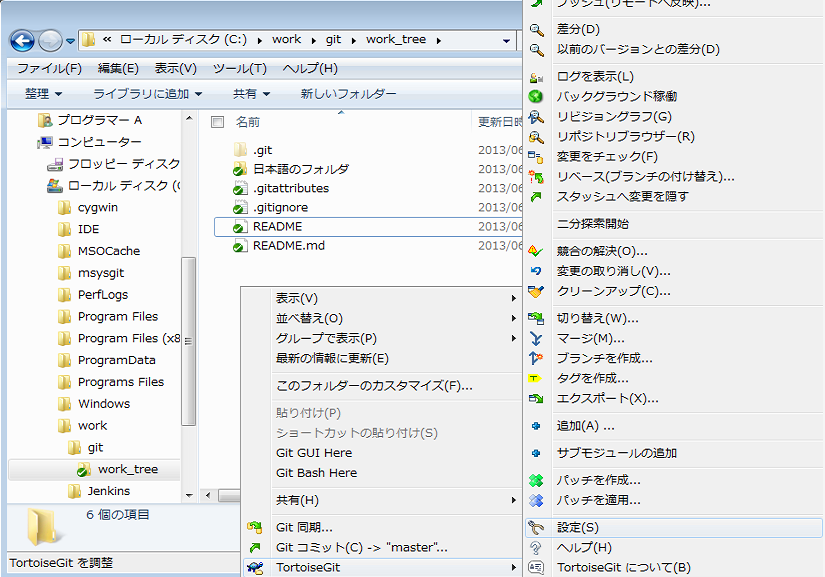
クローンが完了すると、エクスプローラーに空の作業ツリーとローカルリポジトリ（.gitフォルダ）が作成される。

この後の共有リポジトリとの同期操作（プッシュ、プル、フェッチ、リベース）については別紙参照。

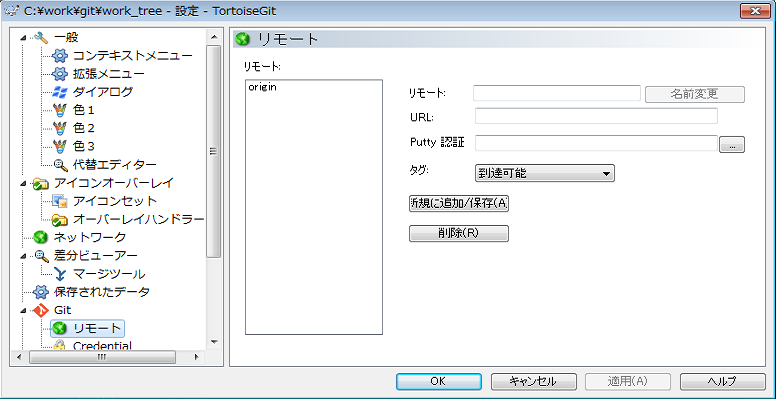
# Gitリポジトリの同期先の追加（http(s)通信）

共有リポジトリのクローンを行わずにローカルPCにローカルリポジトリを作成した後で共有リポジトリと同期したい場合、もしくは、新たに別のリポジトリと同期するようにしたい場合、下記の設定で同期先を追加登録する事ができる。（Gitには、複数の同期先を登録して使い分ける事ができる。）

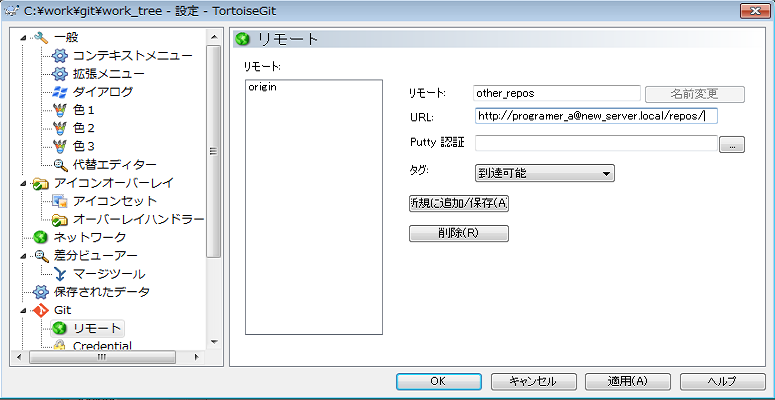
**手順①：** まずはTortoiseGitの「設定」画面を開く。



**手順②：** 設定画面の［Git］→［リモート］ページを表示。

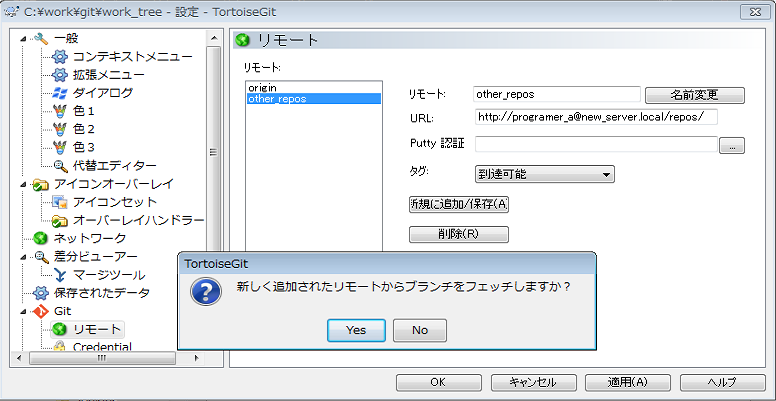


**手順③：** 新しいリモート同期先リポジトリの名称を「リモート」欄に、アクセスURLを「URL」欄に入力した上で、［新規に追加/保存］ボタンを押す。

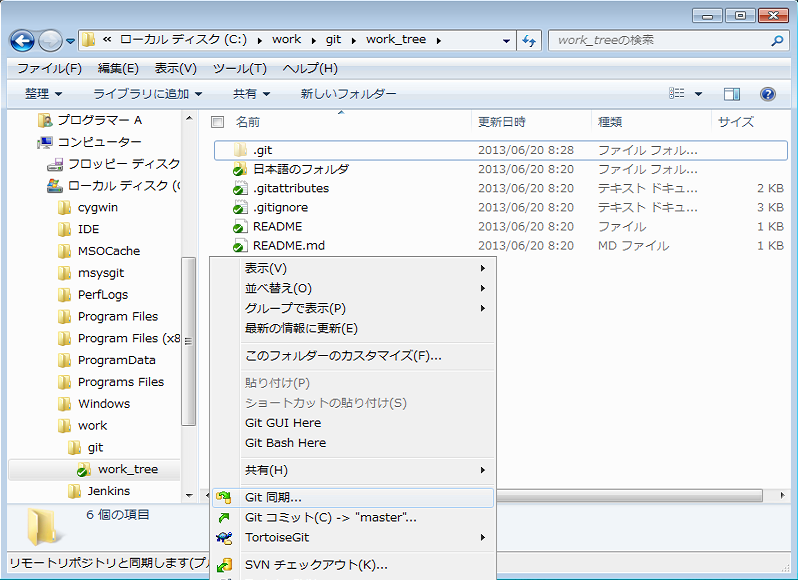


同期先がSSH通信の場合、「Putty認証」欄に秘密鍵のファイルを指定する必要があるが、それについては別紙参照。

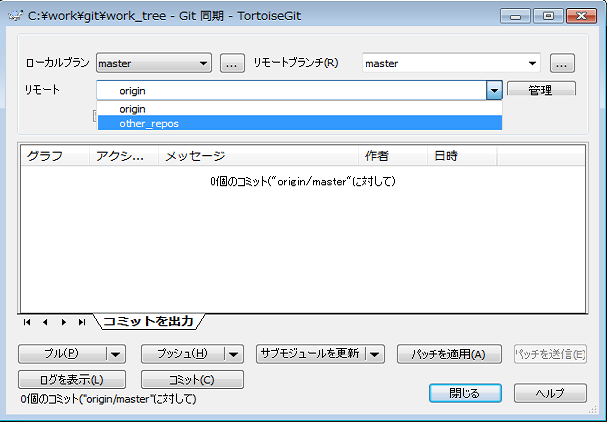
**手順④：** 「リモート」欄のリストに新たに同期先の名称が追加され、そのまま同期先から最新のファイルをフェッチするか問い合わせが出る。（実行してもしなくても良い。）



**手順⑤：** TortoiseGitの「同期」画面にも同様に新しい同期先が追加されている事を確認する。



**手順⑥：** 「同期」画面の「リモート」欄のリストに、新しい同期先が追加されている事を確認。



［管理］ボタンを押すと、「設定」画面が開き、接続先の追加・変更・削除が行える。

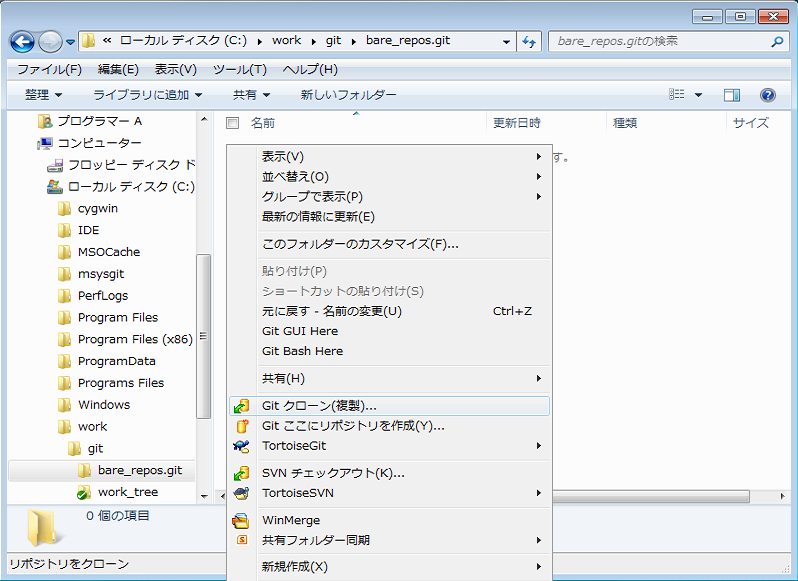
この後の共有リポジトリとの同期操作（プッシュ、プル、フェッチ、リベース）については別紙参照。

# Bareリポジトリの新規作成（方法①）：GUI操作

共有リポジトリを正規のGitサーバーにではなく、自分のPC上や共有フォルダ上に作る事もできる。その場合、作業ツリーを持たない「Bare（裸）リポジトリ」という形式で作成する。

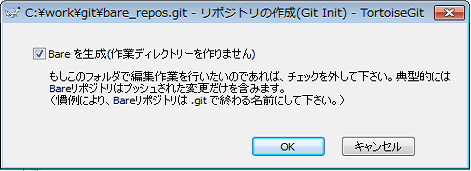
これにより、実際にサーバーがなくても、共有リポジトリとの同期操作を練習してみたり、個人用のバックアップや個人間のやりとりを行ってみたりといった事を行う事ができる。

**手順①：** エクスプローラーで右クリックして、コンテキストメニューから「Gitここにリポジトリを作成…」を実行。

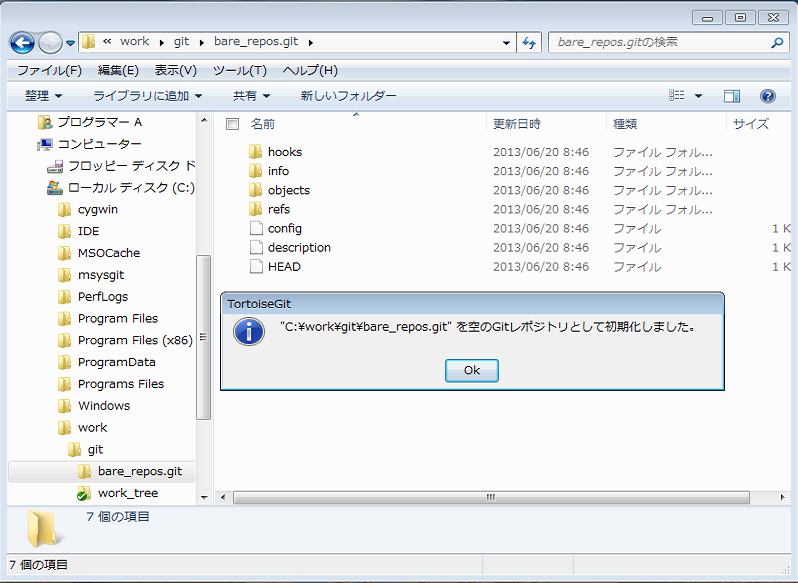


※慣例として、Bareリポジトリのフォルダ名には .git という拡張子を付ける事が多い。（必須ではない）

**手順②：** 「リポジトリの作成（Git Init）」のダイアログにて、「Bare を生成（作業ディレクトリーをつくりません）」のチェックをONにして（デフォルトはOFF）、［OK］ボタンを押す。



**手順③：** Bareリポジトリの新規作成完了。

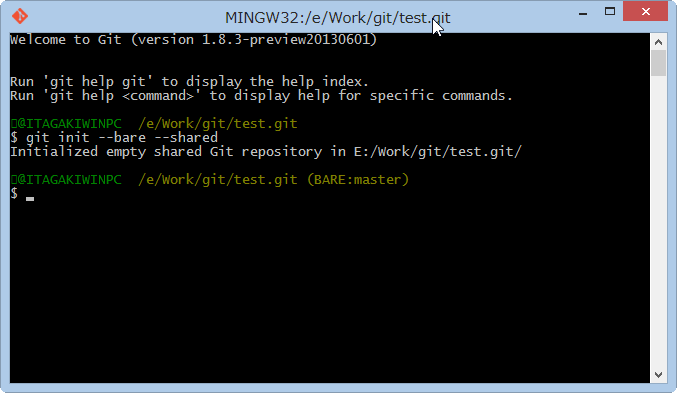


※作業ツリーが作成されていない点に注目。通常なら.gitフォルダ内に作成されるファイル・フォルダ群と同じ内容。

# Bareリポジトリの新規作成（方法②）：コマンドライン操作

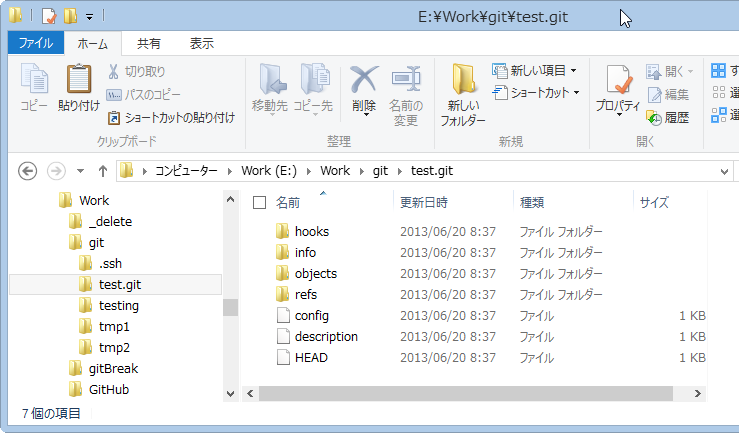
Bareリポジトリを、Linux上やmsysGit上のコマンドラインから作成する場合は、下記の手順で行う。

**手順①：** 対象フォルダに移動して、git init --bare --shared コマンドを実行する。



※慣例として、Bareリポジトリのフォルダ名には .git という拡張子を付ける事が多い。（必須ではない）

**手順②：** Bareリポジトリの新規作成完了。



※作業ツリーが作成されていない点に注目。通常なら.gitフォルダ内に作成されるファイル・フォルダ群と同じ内容。

|  |  |  |  |
| --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  | **TortoiseGitのhttp(s)通信およびパスワードの記憶と破棄について** |  | |

**以　上**