

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
■ Git リポジトリのクローン (http(s) 通信)	1
■ 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
■ 記憶しているユーザーID とパスワードを破棄.....	3
■ Git リポジトリと作業ツリーの新規作成	4
■ Git リポジトリの同期先の追加 (http(s)通信)	5
■ Bare リポジトリの新規作成 (方法①) : GUI 操作	7
■ Bare リポジトリの新規作成 (方法②) : コマンドライン操作	8

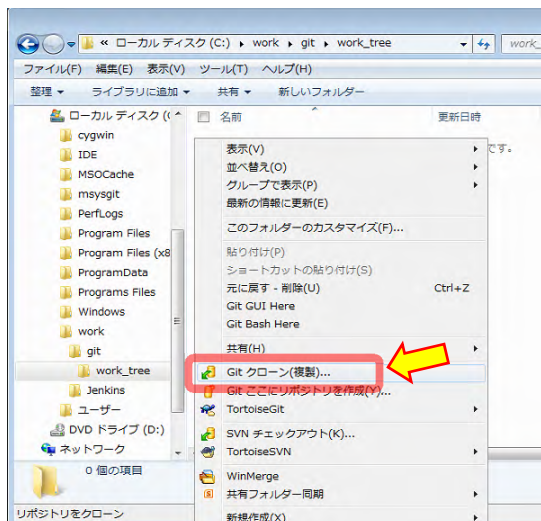
■ 本書が扱うソフトウェアとバージョン

- 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)

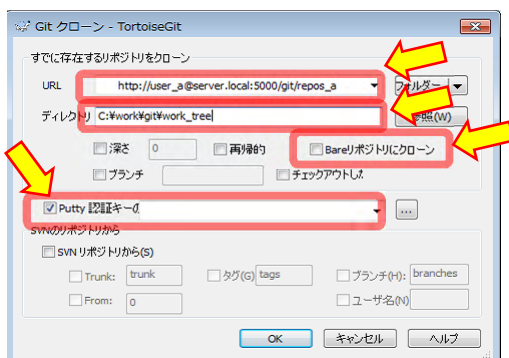
Git の共有リポジトリが既に存在する場合、かつ、そのサーバーとの通信が **http** 通信もしくは **https** 通信に対応している場合、下記の手順でリポジトリをクローンし、ローカル PC に作業ツリーとローカルリポジトリを作成する。
 なお、できる限り **http** 通信は用いない。社内での利用に留めるなら問題がないが、**社外と連携するなら http 通信は NG**。**https** 通信を用いる。また、サーバーが **SSH** 通信に対応しているなら、**https** 通信よりも極力 **SSH** 通信を使用する。

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。



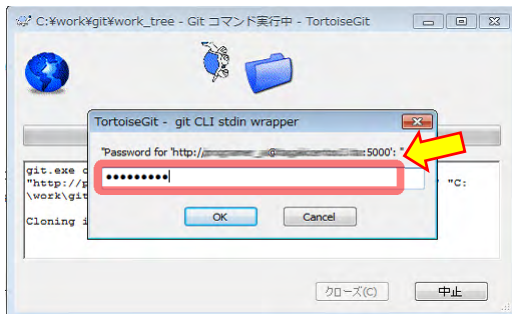
「Bare リポジトリにクローン」のチェックは ON にすると、ローカルリポジトリだけ作られて作業ツリーが作られないので注意。

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。

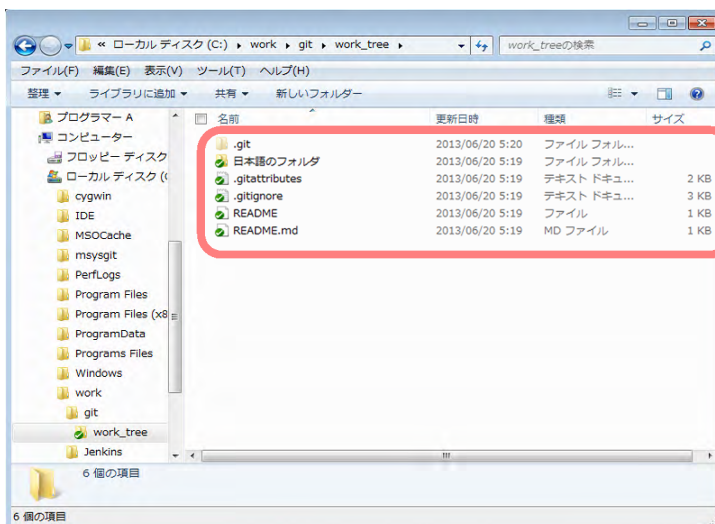
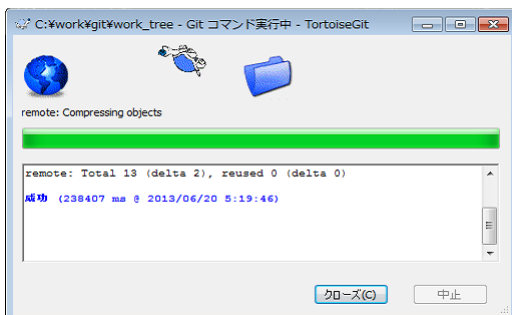
手順③: [OK] ボタンを押してクローンを開始すると、ユーザーのパスワードを入力するダイアログボックスが表示されるので、パスワードを入力する。



「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。

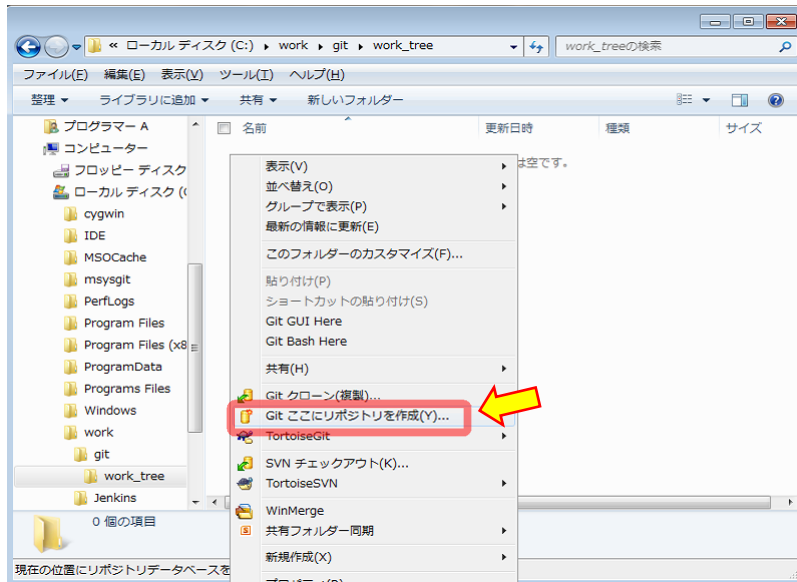
なお、TortoiseGit の設定画面には、TortoiseSVN と同様の、[保存されたデータ] → [認証データ] → [クリア] というボタンがあるが、TortoiseSVN と異なり、この操作を行っても wincred で記録されたユーザー情報は削除されない。

■ Git リポジトリと作業ツリーの新規作成

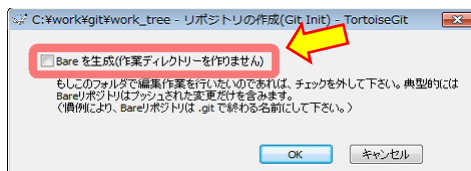
Git の共有リポジトリがまだ存在しない場合、下記の手順でローカル PC に新規の作業ツリーとローカルリポジトリを作成する。

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。

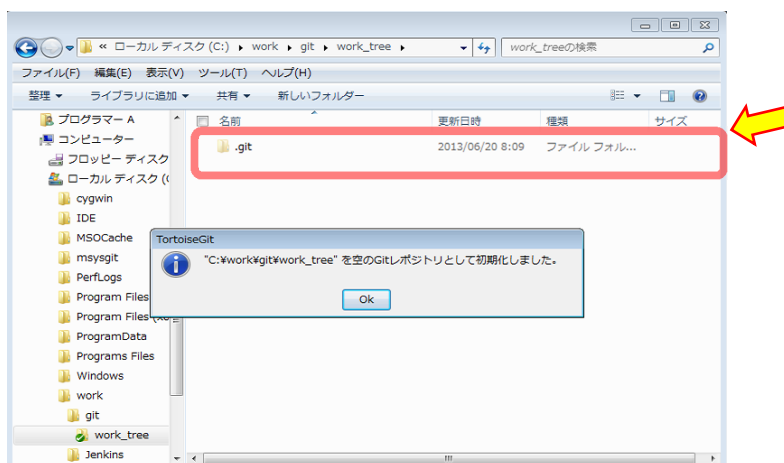
手順①： エクスプローラーで右クリックして、コンテキストメニューから「Git ここにリポジトリを作成...」を実行。



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。



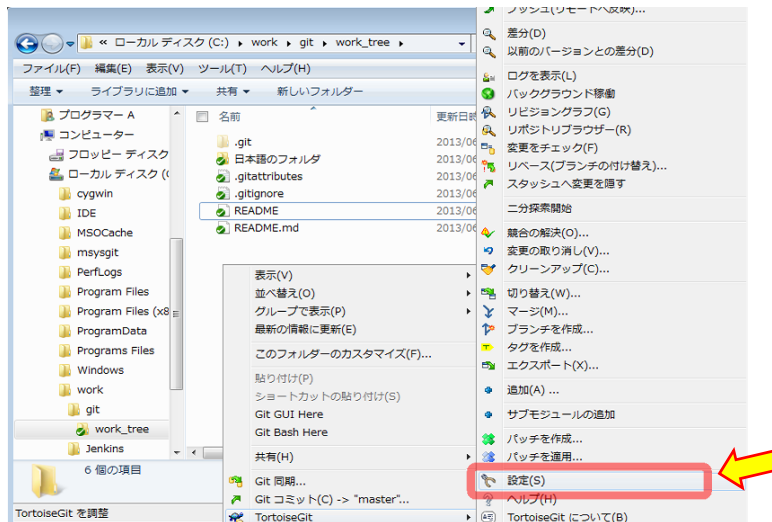
クローンが完了すると、エクスプローラーに空の作業ツリーとローカルリポジトリ (.git フォルダ) が作成される。

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。

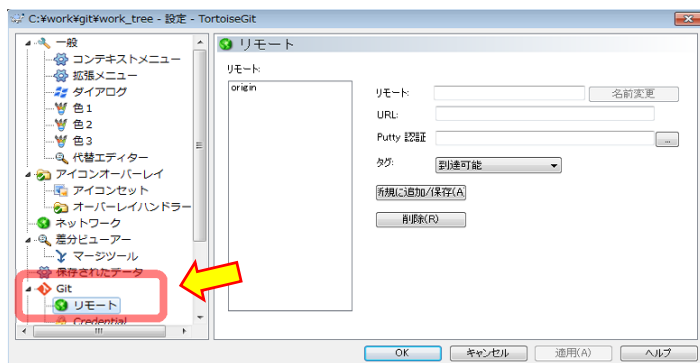
■ Git リポジトリの同期先の追加 (http(s)通信)

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。)

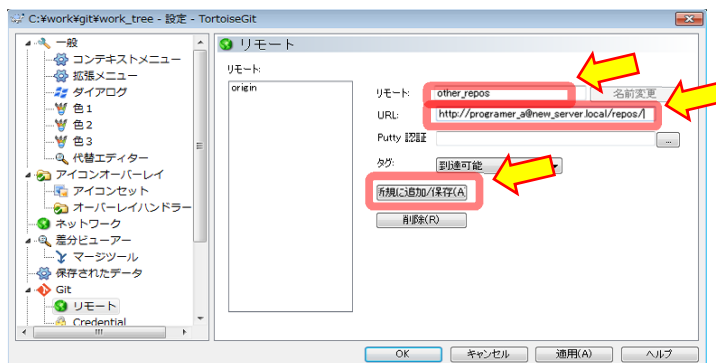
手順①： まずは TortoiseGit の「設定」画面を開く。



手順②： 設定画面の [Git] → [リモート] ページを表示。

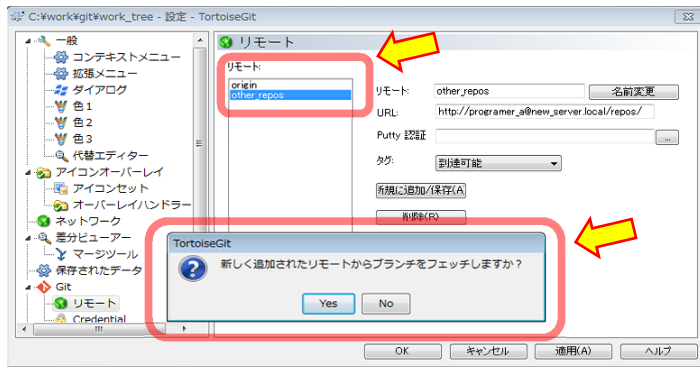


手順③： 新しいリモート同期先リポジトリの名称を「リモート」欄に、アクセス URL を「URL」欄に入力した上で、[新規に追加/保存] ボタンを押す。

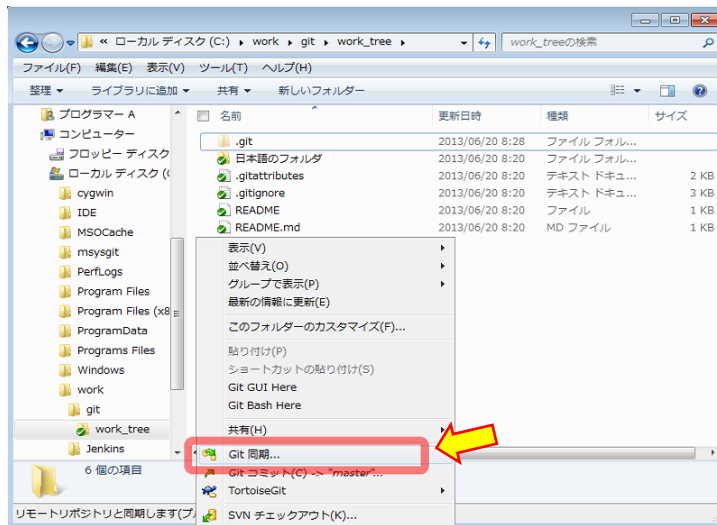


同期先が SSH 通信の場合、「Putty 認証」欄に秘密鍵のファイルを指定する必要があるが、それについては別紙参照。

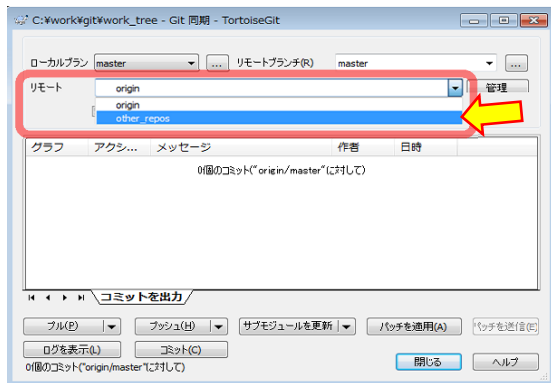
手順④：「リモート」欄のリストに新たに同期先の名称が追加され、そのまま同期先から最新のファイルをフェッチするか問い合わせが出る。(実行してもしなくても良い。)



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。

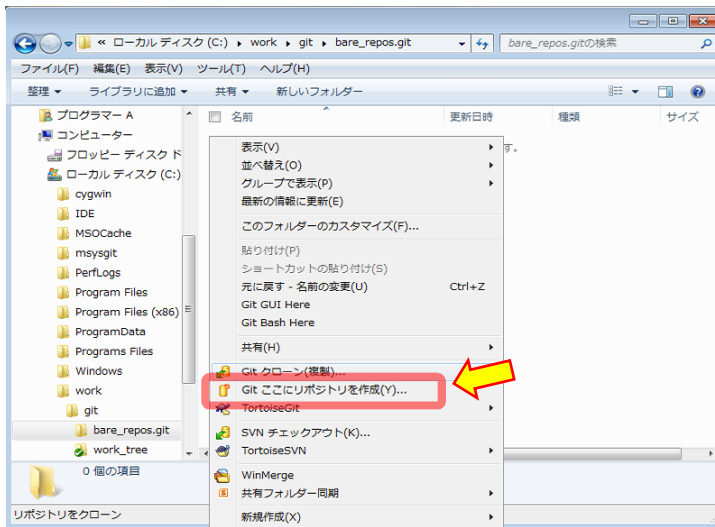
この後の共有リポジトリとの同期操作（プッシュ、プル、フェッチ、リベース）については別紙参照。

■ 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。

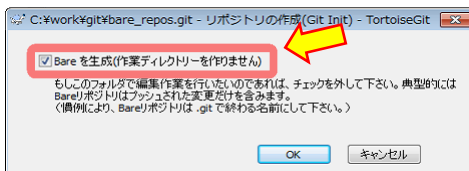
これにより、実際にサーバーがなくても、共有リポジトリとの同期操作を練習してみたり、個人用のバックアップや個人間のやりとりを行ってみたりといった事を行う事ができる。

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。

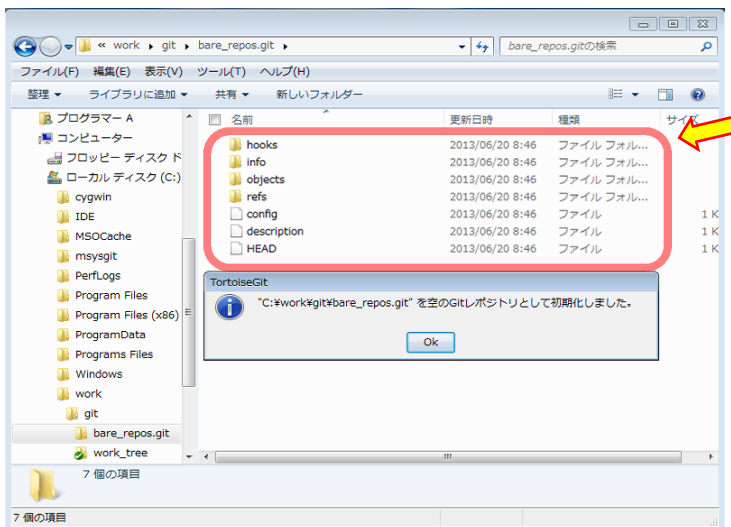


※慣例として、Bare リポジトリのフォルダ名には .git という拡張子を付ける事が多い。（必須ではない）

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。

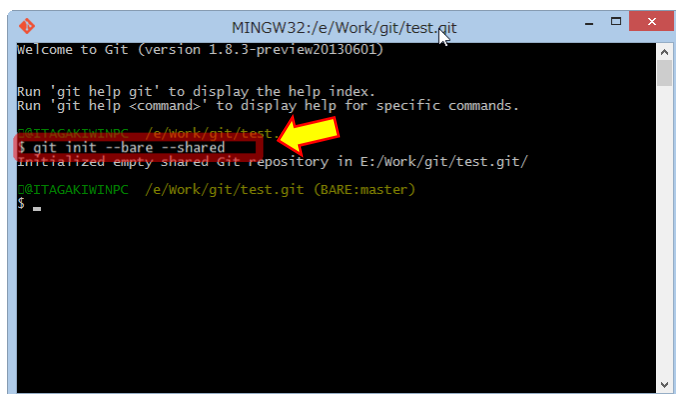


※作業ツリーが作成されていない点に注目。通常なら.git フォルダ内に作成されるファイル・フォルダ群と同じ内容。

■ 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。



```

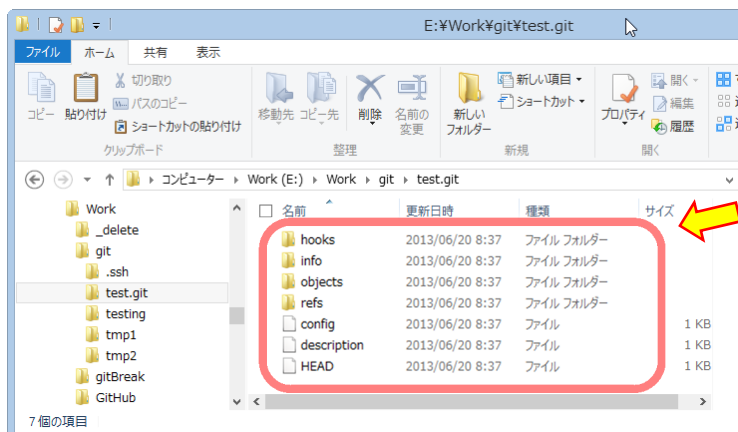
MINGW32:/e:/Work/git/test.git
Welcome to Git (version 1.8.3-preview20130601)

Run 'git help git' to display the help index.
Run 'git help <command>' to display help for specific commands.

$ git init --bare --shared
Initialized empty shared Git repository in E:/Work/git/test.git/
$
  
```

※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