Microsoft Windows [Version 10.0.18362.535] (c) 2019 Microsoft Corporation. All rights reserved. D:\Temp\Hello>git init Initialized empty Git repository in D:/Temp/Hello/.git/ D:\Temp\Hello>git status On branch master No commits yet Untracked files: (use "git add <file>..." to include in what will be committed) hh.txt nothing added to commit but untracked files present (use "git add" to track) D:\Temp\Hello>git add . D:\Temp\Hello>git commit -m "Initial Commit" \*\*\* Please tell me who you are. Run git config --global user.email "you@example.com" git config --global user.name "Your Name" to set your account's default identity. Omit --global to set the identity only in this repository. fatal: unable to auto-detect email address (got 'Ashitosh Dhapate@Ashuu.(none)') D:\Temp\Hello>git config --global user.email "shodan-ad" D:\Temp\Hello>git config --global user.email "dhapateashu.ad@gmail.com" D:\Temp\Hello>git config --global user.name "shodan-ad" D:\Temp\Hello>git config --list diff.astextplain.textconv=astextplain filter.lfs.clean=git-lfs clean -- %f filter.lfs.smudge=git-lfs smudge -- %f filter.lfs.process=git-lfs filter-process filter.lfs.required=true http.sslbackend=openssl core.autocrlf=true core.fscache=true core.symlinks=false credential.helper=manager user.email=dhapateashu.ad@gmail.com user.name=shodan-ad core.repositoryformatversion=0 core.filemode=false core.bare=false core.logallrefupdates=true core.symlinks=false core.ignorecase=true D:\Temp\Hello>git commit -m "Initial Commit" [master (root-commit) 53f008a] Initial Commit 1 file changed, 0 insertions(+), 0 deletions(-) create mode 100644 hh.txt D:\Temp\Hello>git status On branch master Untracked files: (use "git add <file>..." to include in what will be committed) jj.txt nothing added to commit but untracked files present (use "git add" to track) D:\Temp\Hello>git status On branch master Changes not staged for commit: (use "git add <file>..." to update what will be committed) (use "git restore <file>..." to discard changes in working directory) modified: hh.txt Untracked files: (use "git add <file>..." to include in what will be committed) jj.txt no changes added to commit (use "git add" and/or "git commit -a") D:\Temp\Hello>git add hh.txt D:\Temp\Hello>git status On branch master Changes to be committed: (use "git restore --staged <file>..." to unstage) modified: hh.txt Untracked files: (use "git add <file>..." to include in what will be committed) jj.txt D:\Temp\Hello>git commit Aborting commit due to empty commit message.

D:\Temp\Hello>git commit -m "second commit"

[master 83b1119] second commit

1 file changed, 1 insertion(+)

D:\Temp\Hello>git status

On branch master

Untracked files:

(use "git add <file>..." to include in what will be committed)

jj.txt

nothing added to commit but untracked files present (use "git add" to track)

D:\Temp\Hello>git add jj.txt

D:\Temp\Hello>git status

On branch master

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

new file: jj.txt

D:\Temp\Hello>git commit -m "third commit"

[master a9648b2] third commit

1 file changed, 0 insertions(+), 0 deletions(-)

create mode 100644 jj.txt

D:\Temp\Hello>git status

On branch master

nothing to commit, working tree clean

D:\Temp\Hello>git logs

git: 'logs' is not a git command. See 'git --help'.

The most similar command is

log

D:\Temp\Hello>log

'log' is not recognized as an internal or external command,

operable program or batch file.

D:\Temp\Hello>git --help

usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]

[--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]

[-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]

[--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]

<command> [<args>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)

clone Clone a repository into a new directory

init Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)

add Add file contents to the index

mv Move or rename a file, a directory, or a symlink

restore Restore working tree files

rm Remove files from the working tree and from the index

examine the history and state (see also: git help revisions)

bisect Use binary search to find the commit that introduced a bug

diff Show changes between commits, commit and working tree, etc

grep Print lines matching a pattern

log Show commit logs

show Show various types of objects

status Show the working tree status

grow, mark and tweak your common history

branch List, create, or delete branches

commit Record changes to the repository

merge Join two or more development histories together

rebase Reapply commits on top of another base tip

reset Reset current HEAD to the specified state

switch Switch branches

tag Create, list, delete or verify a tag object signed with GPG

collaborate (see also: git help workflows)

fetch Download objects and refs from another repository

pull Fetch from and integrate with another repository or a local branch

push Update remote refs along with associated objects

'git help -a' and 'git help -g' list available subcommands and some

concept guides. See 'git help <command>' or 'git help <concept>'

to read about a specific subcommand or concept.

See 'git help git' for an overview of the system.

D:\Temp\Hello>git log

commit a9648b25cc4c6bc25e9166c8fd5eeae330237a80 (HEAD -> master) Author: shodan-ad <dhapateashu.ad@gmail.com> Date: Wed Jan 8 09:45:47 2020 +0530 third commit commit 83b1119151a3aa743dbb46f54836ca9de0107fa3 Author: shodan-ad <dhapateashu.ad@gmail.com> Date: Wed Jan 8 09:45:25 2020 +0530

second commit

commit 53f008aa4fbe98f1d38fae5857bca6f2887987e2

Author: shodan-ad <dhapateashu.ad@gmail.com>

Date: Wed Jan 8 09:43:32 2020 +0530

Initial Commit

D:\Temp\Hello>git rm hh.txt

rm 'hh.txt'

D:\Temp\Hello>git status

On branch master

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

deleted: hh.txt

D:\Temp\Hello>ls

'ls' is not recognized as an internal or external command,

operable program or batch file.

D:\Temp\Hello>git status

On branch master

Changes to be committed:

(use "git restore --staged <file>..." to unstage)

deleted: hh.txt

Untracked files:

(use "git add <file>..." to include in what will be committed)

kk.txt

D:\Temp\Hello>git add .

D:\Temp\Hello>git commit -m "third commit"

[master d0f3087] third commit

2 files changed, 1 deletion(-)

delete mode 100644 hh.txt

create mode 100644 kk.txt

D:\Temp\Hello>git remote add origin https://github.com/shodan-ad/Demo.git

D:\Temp\Hello>git push

fatal: The current branch master has no upstream branch.

To push the current branch and set the remote as upstream, use

git push --set-upstream origin master

D:\Temp\Hello>git push --set-upstream origin master

To https://github.com/shodan-ad/Demo.git

! [rejected] master -> master (fetch first)

error: failed to push some refs to 'https://github.com/shodan-ad/Demo.git'

hint: Updates were rejected because the remote contains work that you do

hint: not have locally. This is usually caused by another repository pushing

hint: to the same ref. You may want to first integrate the remote changes

hint: (e.g., 'git pull ...') before pushing again.

hint: See the 'Note about fast-forwards' in 'git push --help' for details.

D:\Temp\Hello>git push --set-upstream origin master

To https://github.com/shodan-ad/Demo.git

! [rejected] master -> master (fetch first)

error: failed to push some refs to 'https://github.com/shodan-ad/Demo.git'

hint: Updates were rejected because the remote contains work that you do

hint: not have locally. This is usually caused by another repository pushing

hint: to the same ref. You may want to first integrate the remote changes

hint: (e.g., 'git pull ...') before pushing again.

hint: See the 'Note about fast-forwards' in 'git push --help' for details.

D:\Temp\Hello>git remote add origin https://github.com/shodan-ad/Demo.git

fatal: remote origin already exists.

D:\Temp\Hello>git pull

warning: no common commits

remote: Enumerating objects: 3, done.

remote: Counting objects: 100% (3/3), done.

remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

Unpacking objects: 100% (3/3), done.

From https://github.com/shodan-ad/Demo

\* [new branch] master -> origin/master

There is no tracking information for the current branch.

Please specify which branch you want to merge with.

See git-pull(1) for details.

git pull <remote> <branch>

If you wish to set tracking information for this branch you can do so with:

git branch --set-upstream-to=origin/<branch> master

D:\Temp\Hello>git push

fatal: The current branch master has no upstream branch.

To push the current branch and set the remote as upstream, use

git push --set-upstream origin master

D:\Temp\Hello>git push --set-upstream origin master

To https://github.com/shodan-ad/Demo.git

! [rejected] master -> master (non-fast-forward)

error: failed to push some refs to 'https://github.com/shodan-ad/Demo.git'

hint: Updates were rejected because the tip of your current branch is behind

hint: its remote counterpart. Integrate the remote changes (e.g.

hint: 'git pull ...') before pushing again.

hint: See the 'Note about fast-forwards' in 'git push --help' for details.

D:\Temp\Hello>

D:\Temp\Hello>git remote remove origin

D:\Temp\Hello>git status

On branch master

nothing to commit, working tree clean

D:\Temp\Hello>git remote add origin https://github.com/shodan-ad/Demo.git

D:\Temp\Hello>git pull

From https://github.com/shodan-ad/Demo

\* [new branch] master -> origin/master

There is no tracking information for the current branch.

Please specify which branch you want to merge with.

See git-pull(1) for details.

git pull <remote> <branch>

If you wish to set tracking information for this branch you can do so with:

git branch --set-upstream-to=origin/<branch> master

D:\Temp\Hello>git pull master

fatal: 'master' does not appear to be a git repository

fatal: Could not read from remote repository.

Please make sure you have the correct access rights

and the repository exists.

D:\Temp\Hello>git pull origin https://github.com/shodan-ad/Demo.git master

fatal: invalid refspec 'https://github.com/shodan-ad/Demo.git'

D:\Temp\Hello>git status

On branch master

nothing to commit, working tree clean

D:\Temp\Hello>git add .

D:\Temp\Hello>git commit -m "gg"

On branch master

nothing to commit, working tree clean

D:\Temp\Hello>git push

fatal: The current branch master has no upstream branch.

To push the current branch and set the remote as upstream, use

git push --set-upstream origin master

D:\Temp\Hello>git push --set-upstream origin master

To https://github.com/shodan-ad/Demo.git

! [rejected] master -> master (non-fast-forward)

error: failed to push some refs to 'https://github.com/shodan-ad/Demo.git'

hint: Updates were rejected because the tip of your current branch is behind

hint: its remote counterpart. Integrate the remote changes (e.g.

hint: 'git pull ...') before pushing again.

hint: See the 'Note about fast-forwards' in 'git push --help' for details.

D:\Temp\Hello>git pull

There is no tracking information for the current branch.

Please specify which branch you want to merge with.

See git-pull(1) for details.

git pull <remote> <branch>

If you wish to set tracking information for this branch you can do so with:

git branch --set-upstream-to=origin/<branch> master

D:\Temp\Hello>git pull origin master --allow-undeleted-histories

error: unknown option `allow-undeleted-histories'

usage: git pull [<options>] [<repository> [<refspec>...]]

-v, --verbose be more verbose

-q, --quiet be more quiet

--progress force progress reporting

--recurse-submodules[=<on-demand>]

control for recursive fetching of submodules

Options related to merging

-r, --rebase[=(false|true|merges|preserve|interactive)]

incorporate changes by rebasing rather than merging

-n do not show a diffstat at the end of the merge

--stat show a diffstat at the end of the merge

--log[=<n>] add (at most <n>) entries from shortlog to merge commit message

--signoff[=...] add Signed-off-by:

--squash create a single commit instead of doing a merge

--commit perform a commit if the merge succeeds (default)

--edit edit message before committing

--cleanup <mode> how to strip spaces and #comments from message

--ff allow fast-forward

--ff-only abort if fast-forward is not possible

--verify-signatures verify that the named commit has a valid GPG signature

--autostash automatically stash/stash pop before and after rebase

-s, --strategy <strategy>

merge strategy to use

-X, --strategy-option <option=value>

option for selected merge strategy

-S, --gpg-sign[=<key-id>]

GPG sign commit

--allow-unrelated-histories

allow merging unrelated histories

Options related to fetching

--all fetch from all remotes

-a, --append append to .git/FETCH\_HEAD instead of overwriting

--upload-pack <path> path to upload pack on remote end

-f, --force force overwrite of local branch

-t, --tags fetch all tags and associated objects

-p, --prune prune remote-tracking branches no longer on remote

-j, --jobs[=<n>] number of submodules pulled in parallel

--dry-run dry run

-k, --keep keep downloaded pack

--depth <depth> deepen history of shallow clone

--unshallow convert to a complete repository

--update-shallow accept refs that update .git/shallow

--refmap <refmap> specify fetch refmap

-4, --ipv4 use IPv4 addresses only

-6, --ipv6 use IPv6 addresses only

--show-forced-updates

check for forced-updates on all updated branches

--set-upstream set upstream for git pull/fetch

D:\Temp\Hello>git pull origin master --allow-unrelated-histories

From https://github.com/shodan-ad/Demo

\* branch master -> FETCH\_HEAD

Merge made by the 'recursive' strategy.

README.md | 1 +

1 file changed, 1 insertion(+)

create mode 100644 README.md

D:\Temp\Hello>git push --set-upstream origin master

Enumerating objects: 13, done.

Counting objects: 100% (13/13), done.

Delta compression using up to 4 threads

Compressing objects: 100% (8/8), done.

Writing objects: 100% (12/12), 1.13 KiB | 192.00 KiB/s, done.

Total 12 (delta 0), reused 0 (delta 0)

To https://github.com/shodan-ad/Demo.git

9fe60f1..a9be15a master -> master

Branch 'master' set up to track remote branch 'master' from 'origin'.

D:\Temp\Hello>git status

On branch master

Your branch is up to date with 'origin/master'.

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git restore <file>..." to discard changes in working directory)

modified: kk.txt

no changes added to commit (use "git add" and/or "git commit -a")

D:\Temp\Hello>git add .

D:\Temp\Hello>git commit -m "jjjj"

[master 10c9579] jjjj

1 file changed, 1 insertion(+)

D:\Temp\Hello>git push

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Delta compression using up to 4 threads

Compressing objects: 100% (2/2), done.

Writing objects: 100% (3/3), 251 bytes | 83.00 KiB/s, done.

Total 3 (delta 1), reused 0 (delta 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/shodan-ad/Demo.git

a9be15a..10c9579 master -> master

D:\Temp\Hello>git status

On branch master

Your branch is up to date with 'origin/master'.

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git restore <file>..." to discard changes in working directory)

modified: kk.txt

no changes added to commit (use "git add" and/or "git commit -a")

D:\Temp\Hello>git add kk.txt

D:\Temp\Hello>git commit -m "Next"

[master c8f518c] Next

1 file changed, 1 insertion(+), 1 deletion(-)

D:\Temp\Hello>git push

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Delta compression using up to 4 threads

Compressing objects: 100% (2/2), done.

Writing objects: 100% (3/3), 258 bytes | 129.00 KiB/s, done.

Total 3 (delta 1), reused 0 (delta 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/shodan-ad/Demo.git

10c9579..c8f518c master -> master

D:\Temp\Hello>