

```
Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance
$ git init
Initialized empty Git repository in C:/Users/Tanusha/Desktop/Cognizance/.git/

Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (master)
$ git add .

Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (master)
$ git commit -m "comment"
[master (root-commit) c31d5b4] comment
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Git_Commands_Task-1.pdf

Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (master)
$ git remote add origin https://github.com/Tanusha1408/Cognizance

Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (master)
$ git push origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 917.63 KiB | 11.62 MiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:   https://github.com/Tanusha1408/Cognizance/pull/new/master
remote:
To https://github.com/Tanusha1408/Cognizance
 * [new branch]      master -> master

Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (master)
$ git add Git_Commands_Task-1.pdf

Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (master)
$ git checkout Git_Commands_Task-1.pdf
Updated 0 paths from the index

Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (master)
$ git checkout -b Git_Commands_Task-1.pdf
Switched to a new branch 'Git_Commands_Task-1.pdf'

Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (Git_Commands_Task-1.pdf)
$ git commit
On branch Git_Commands_Task-1.pdf
nothing to commit, working tree clean
```



38°C
Cloudy



ENG
IN



14:44
09-06-2022

```
Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (Git_Commands_Task-1.pdf)
$ git commit
On branch Git_Commands_Task-1.pdf
nothing to commit, working tree clean
```

```
Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (Git_Commands_Task-1.pdf)
$ git commit -m "Task-1"
On branch Git_Commands_Task-1.pdf
nothing to commit, working tree clean
```

```
Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (Git_Commands_Task-1.pdf)
$ git commit -a
On branch Git_Commands_Task-1.pdf
nothing to commit, working tree clean
```

```
Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (Git_Commands_Task-1.pdf)
$ git fetch
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), 604 bytes | 40.00 KiB/s, done.
From https://github.com/Tanusha1408/Cognizance
* [new branch]      main       -> origin/main
```

```
Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (Git_Commands_Task-1.pdf)
$ 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> Git_Commands_Task-1.pdf
```

```
Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (Git_Commands_Task-1.pdf)
$ git push origin master
Everything up-to-date
```

```
Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (Git_Commands_Task-1.pdf)
$ git reset
```

```
Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (Git_Commands_Task-1.pdf)
$ git fork
git: 'fork' is not a git command. See 'git --help'.
```

The most similar commands are

fsck
hook

Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (Git_Commands_Task-1.pdf)

\$ git fsck

Checking object directories: 100% (256/256), done.

Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (Git_Commands_Task-1.pdf)

\$ git clone -b https://github.com/Tanusha1408/Cognizance.git

fatal: You must specify a repository to clone.

usage: git clone [<options>] [--] <repo> [<dir>]

| | |
|-----------------------------------|---|
| -v, --verbose | be more verbose |
| -q, --quiet | be more quiet |
| --progress | force progress reporting |
| --reject-shallow | don't clone shallow repository |
| -n, --no-checkout | don't create a checkout |
| --bare | create a bare repository |
| --mirror | create a mirror repository (implies bare) |
| -l, --local | to clone from a local repository |
| --no-hardlinks | don't use local hardlinks, always copy |
| -s, --shared | setup as shared repository |
| --recurse-submodules[=<pathspec>] | initialize submodules in the clone |
| --recursive ... | alias of --recurse-submodules |
| -j, --jobs <n> | number of submodules cloned in parallel |
| --template <template-directory> | directory from which templates will be used |
| --reference <repo> | reference repository |
| --reference-if-able <repo> | reference repository |
| --dissociate | use --reference only while cloning |
| -o, --origin <name> | use <name> instead of 'origin' to track upstream |
| -b, --branch <branch> | checkout <branch> instead of the remote's HEAD |
| -u, --upload-pack <path> | path to git-upload-pack on the remote |
| --depth <depth> | create a shallow clone of that depth |
| --shallow-since <time> | create a shallow clone since a specific time |
| --shallow-exclude <revision> | deepen history of shallow clone, excluding rev |
| --single-branch | clone only one branch, HEAD or --branch |
| --no-tags | don't clone any tags, and make later fetches not to follow them |
| --shallow-submodules | any cloned submodules will be shallow |

```
--separate-git-dir <gitdir>
    separate git dir from working tree
-c, --config <key=value>
    set config inside the new repository
--server-option <server-specific>
    option to transmit
-4, --ipv4
    use IPv4 addresses only
-6, --ipv6
    use IPv6 addresses only
--filter <args>
    object filtering
--also-filter-submodules
    apply partial clone filters to submodules
--remote-submodules
    any cloned submodules will use their remote-tracking branch
--sparse
    initialize sparse-checkout file to include only files at root
```

Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (Git_Commands_Task-1.pdf)

\$ git config --list --show-origin

```
file:C:/Program Files/Git/etc/gitconfig diff.astextplain.textconv=astextplain
file:C:/Program Files/Git/etc/gitconfig filter.lfs.clean=git-lfs clean -- %f
file:C:/Program Files/Git/etc/gitconfig filter.lfs.smudge=git-lfs smudge -- %f
file:C:/Program Files/Git/etc/gitconfig filter.lfs.process=git-lfs filter-process
file:C:/Program Files/Git/etc/gitconfig filter.lfs.required=true
file:C:/Program Files/Git/etc/gitconfig http.sslbackend=openssl
file:C:/Program Files/Git/etc/gitconfig http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
file:C:/Program Files/Git/etc/gitconfig core.autocrlf=true
file:C:/Program Files/Git/etc/gitconfig core.fscache=true
file:C:/Program Files/Git/etc/gitconfig core.symlinks=false
file:C:/Program Files/Git/etc/gitconfig pull.rebase=false
file:C:/Program Files/Git/etc/gitconfig credential.helper=manager-core
file:C:/Program Files/Git/etc/gitconfig credential.https://dev.azure.com.usehttppath=true
file:C:/Program Files/Git/etc/gitconfig init.defaultbranch=master
file:C:/Users/Tanusha/.gitconfig filter.lfs.required=true
file:C:/Users/Tanusha/.gitconfig filter.lfs.clean=git-lfs clean -- %f
file:C:/Users/Tanusha/.gitconfig filter.lfs.smudge=git-lfs smudge -- %f
file:C:/Users/Tanusha/.gitconfig filter.lfs.process=git-lfs filter-process
file:C:/Users/Tanusha/.gitconfig user.name=TANUSHA GUDISe
file:C:/Users/Tanusha/.gitconfig user.email=75236583+Tanusha1408@users.noreply.github.com
file:.git/config core.repositoryformatversion=0
file:.git/config core.filemode=false
file:.git/config core.bare=false
file:.git/config core.logallrefupdates=true
file:.git/config core.symlinks=false
file:.git/config core.ignorecase=true
file:.git/config remote.origin.url=https://github.com/Tanusha1408/Cognizance
file:.git/config remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
```

Tanusha@DESKTOP-PVIU9AS MINGW64 ~/Desktop/Cognizance (Git_Commands_Task-1.pdf)

\$