```
SKI@DESKTOP-LG785IS MINGW64 /e/git
$ git init
Initialized empty Git repository in E:/git/.git/

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ ls-a
bash: ls-a: command not found

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ ls -a
./  ../  .git/  communication.pptx

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$
```

```
SKI@DESKTOP-LG785IS MINGW64 /e/git
$ git init
Initialized empty Git repository in E:/git/.git/

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ ls-a
bash: ls-a: command not found

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ ls -a
./  ../  .git/  communication.pptx

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git config --global user.name "VikasB07"

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git config --global user.email "vikasboopathy26@gmail.com"

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git add "communication^C


SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git add .

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   communication.pptx


SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ |
```

```
SKI@DESKTOP-LG785IS MINGW64 /e/git (vikas)
$ git status
On branch vikas
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:   sample1.txt

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

SKI@DESKTOP-LG785IS MINGW64 /e/git (vikas)
$ git add "sample1.txt"

SKI@DESKTOP-LG785IS MINGW64 /e/git (vikas)
$ git commit -m "college"
[vikas 45ebbc5] college
 1 file changed, 1 insertion(+), 1 deletion(-)

SKI@DESKTOP-LG785IS MINGW64 /e/git (vikas)
$ git status
On branch vikas
nothing to commit, working tree clean

SKI@DESKTOP-LG785IS MINGW64 /e/git (vikas)
$ git add "sample1.txt"

SKI@DESKTOP-LG785IS MINGW64 /e/git (vikas)
$ git status
On branch vikas
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   sample1.txt


SKI@DESKTOP-LG785IS MINGW64 /e/git (vikas)
$ git commit -m "college"
[vikas 96ac1e4] college
 1 file changed, 1 insertion(+), 1 deletion(-)

SKI@DESKTOP-LG785IS MINGW64 /e/git (vikas)
$ git merge vikas
Already up to date.

SKI@DESKTOP-LG785IS MINGW64 /e/git (vikas)
$ git merge master
Already up to date.

SKI@DESKTOP-LG785IS MINGW64 /e/git (vikas)
$
```

```
SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git branch
* master

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git branch vikas

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git branch
* master
  vikas

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git chechout vikas
git: 'chechout' is not a git command. See 'git --help'.

The most similar command is
        checkout

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git checkout vikas
M       sample1.txt
Switched to branch 'vikas'
```

```
SKI@DESKTOP-LG785IS MINGW64 /e/git (vikas)
$ git remote add origin https://github.com/VikasB07/workspace.git

SKI@DESKTOP-LG785IS MINGW64 /e/git (vikas)
$ git checkout master
D        communication.pptx
D        ~$communication.pptx
Switched to branch 'master'

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git remote add origin https://github.com/VikasB07/workspace.git
error: remote origin already exists.

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git remote remove origin

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git remote add origin https://github.com/VikasB07/workspace.git

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git add .

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git commit -m "college"
[master 0f66c36] college
 2 files changed, 0 insertions(+), 0 deletions(-)
 delete mode 100644 communication.pptx
 delete mode 100644 ~$communication.pptx

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git push -u origin master
info: please complete authentication in your browser...
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 8 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (13/13), 1.73 MiB | 103.00 KiB/s, done.
Total 13 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/VikasB07/workspace.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

```
SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git push -u origin vikas
Enumerating objects: 16, done.
Counting objects: 100% (16/16), done.
Delta compression using up to 8 threads
Compressing objects: 100% (12/12), done.
Writing objects: 100% (14/14), 1.24 KiB | 140.00 KiB/s, done.
Total 14 (delta 6), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (6/6), completed with 1 local object.
remote:
remote: Create a pull request for 'vikas' on GitHub by visiting:
remote:        https://github.com/VikasB07/workspace/pull/new/vikas
remote:
To https://github.com/VikasB07/workspace.git
 * [new branch]       vikas -> vikas
branch 'vikas' set up to track 'origin/vikas'.

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git pull
remote: Enumerating objects: 7, done.
remote: Counting objects: 100% (7/7), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (5/5), 1.79 KiB | 131.00 KiB/s, done.
From https://github.com/VikasB07/workspace
   0f66c36..f81786a  master     -> origin/master
Updating 0f66c36..f81786a
Fast-forward
 sample1.txt   |    2 +-
 vik new       |    1 +
 vikas 1.pptx | Bin 0 -> 1841141 bytes
 3 files changed, 2 insertions(+), 1 deletion(-)
 create mode 100644 vik new
 create mode 100644 vikas 1.pptx
```

```
SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ touch sample2.txt

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git add .

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git commit -m "meg"
[master fa78e95] meg
 1 file changed, 4 insertions(+)
 create mode 100644 sample2.txt

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git add .

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git commit -m "meg1"
[master c6bd87a] meg1
 1 file changed, 2 insertions(+), 1 deletion(-)

SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git log
commit c6bd87aa6005a3cc94ae4b6b6a2ea49ace6f5e05 (HEAD -> master)
Author: VikasB07 <vikasboopathy26@gmail.com>
Date:   Thu Nov 20 16:04:11 2025 +0530

    meg1

commit fa78e95fa6e390bfd81280f243b2d56da1e619bb
Author: VikasB07 <vikasboopathy26@gmail.com>
Date:   Thu Nov 20 16:02:57 2025 +0530

    meg

commit f81786abf2880727e40db9f53bf4471019e9c64b (origin/master, origin/HEAD)
Author: VikasB07 <vikasboopathy26@gmail.com>
Date:   Thu Nov 20 15:50:58 2025 +0530

    Create vik new

commit cb1020d90f948263b73176de7c4d6ce0068ccb4c
```

```
SKI@DESKTOP-LG785IS MINGW64 /e/git (master)
$ git checkout fa78e95fa6e390bfd81280f243b2d56da1e619bb
Note: switching to 'fa78e95fa6e390bfd81280f243b2d56da1e619bb'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

  git switch -c <new-branch-name>

Or undo this operation with:

  git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at fa78e95 meg

SKI@DESKTOP-LG785IS MINGW64 /e/git ((fa78e95...))
$
```