

1) Sudo apt-get install git

Installing git application

```
ubuntu@ip-172-31-39-107:~$ sudo apt-get install git
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  git
0 upgraded, 1 newly installed, 0 to remove and 26 not upgraded.
Need to get 3679 kB of archives.
After this operation, 22.2 MB of additional disk space will be used.
Get:1 http://us-east-1-ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 git amd64 1:2.43.0-1ubuntu7.1 [3679 kB]
Fetched 3679 kB in 0s (48.4 MB/s)
Selecting previously unselected package git.
(Reading database ... 66859 files and directories currently installed.)
Preparing to unpack .../git_1%3a2.43.0-1ubuntu7.1_amd64.deb ...
Unpacking git (1:2.43.0-1ubuntu7.1) ...
Setting up git (1:2.43.0-1ubuntu7.1) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-39-107:~$
```

2) Git init

Initializing git repository

```
ubuntu@ip-172-31-39-107:~/gitassignment$ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/ubuntu/gitassignment/.git/
ubuntu@ip-172-31-39-107:~/gitassignment$ ls -a
.  ..  .git
ubuntu@ip-172-31-39-107:~/gitassignment$
```

3) Configuring

Configuring user details

```
git config --global user.name "shruthi"
```

```
git config --global user.mail "shruthir1017@gmail.com"
```

4) add file and commit, status

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ touch file1
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git add file1
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git commit -m "file1 added"
[master (root-commit) 0e52f41] file1 added
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file1
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git status
On branch master
nothing to commit, working tree clean
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git config --global user.name "shruthi"
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git config --global user.mail "shruthir1017@gmail.com"
```

5) Git log

```
ubuntu@ip-172-31-39-107:~/gitassignment$ git log
commit cdd88fb1bffc947d384ed4627edf2d095b0a2a54 (HEAD -> master)
Author: shruthi <ubuntu@ip-172-31-39-107.ec2.internal>
Date:   Wed Jul 31 08:07:38 2024 +0000

    file 1 added
ubuntu@ip-172-31-39-107:~/gitassignment$ |
```

Git log --oneline

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git log --oneline
9c3c8d9 (HEAD -> master) f3 created
6ac07dd f2 created
aa612e8 f1 created
```

Git log <file-name>

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git log f1
commit 1db42512d72730d01e27236889f4a74e1e9fc0de (HEAD -> master)
Author: shruthi <shruthir1017@gmail.com>
Date:   Fri Aug 2 12:44:36 2024 +0530

    f1 updated

commit aa612e80e8c579236743e5260276e63d8d74ffc3
Author: shruthi <shruthir1017@gmail.com>
Date:   Fri Aug 2 12:41:28 2024 +0530

    f1 created
```

Git log --author="user-name"

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git log --author="shruthi" --oneline
1db4251 (HEAD -> master) f1 updated
9c3c8d9 f3 created
6ac07dd f2 created
aa612e8 f1 created
```

6) Git blame filename-display who modified the file line by line

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git blame index.txt
^fa6a2b9 (shruthi 2024-07-31 23:41:26 +0530 1)  hello
2e0c2fef (shruthi 2024-07-31 23:43:58 +0530 2)  hello
```

7) Git restore <filename>

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   f1

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    deleted:    f2
    deleted:    f3

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git restore f3

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   f1

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    deleted:    f2
```

Git restore . (all files will be restored)

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   f1

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    deleted:    f2

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git restore .

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   f1

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ |
```

Git restore --staged <file-name>

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   disha1
    new file:   t1
    new file:   t2
    new file:   t3

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git restore --staged t1

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   disha1
    new file:   t2
    new file:   t3

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    t1
```

Git restore --staged . (All files which are in staging area)

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   disha1
    new file:   t2
    new file:   t3

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    t1

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git restore --staged .

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    disha1
    t1
    t2
    t3

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

8) .gitignore

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ touch shru{1..5}


Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ ls
f1 f2 f3 shru1 shru2 shru3 shru4 shru5

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ vi .gitignore

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .gitignore
        shru4
        shru5

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

Vi .gitignore (include the files which you want to untrack)

 MINGW64:/f/New folder

```
shru1
shru2
shru3|
```

9) Git commit --amend

To edit recent commit message

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ touch file2

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git add file2

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git commit -m "file2 created"
[master e5378ff] file2 created
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file2

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git commit --amend
[master d2be7b4] file2 updated
Date: Thu Aug 1 00:02:56 2024 +0530
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file2
```

file2 created

```
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# Date:      Thu Aug 1 00:02:56 2024 +0530
#
# On branch master
# Changes to be committed:
#       new file:   file2
#
~
```

10) Git checkout <commit-id>

```
commit cdd88fb1bffc947d384ed4627edf2d095b0a2a54 (HEAD -> master)
Author: shruthi <ubuntu@ip-172-31-39-107.ec2.internal>
Date:   Wed Jul 31 08:07:38 2024 +0000

    file 1 added

ubuntu@ip-172-31-39-107:~/gitassignment$ git checkout cdd88fb1bffc947d384ed4627edf2d095b0a2a54
Note: switching to 'cdd88fb1bffc947d384ed4627edf2d095b0a2a54'.

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 cdd88fb file 1 added
ubuntu@ip-172-31-39-107:~/gitassignment$ git branch
* (HEAD detached at cdd88fb)
  master
```

11) Git branch

Git branch feture1

Git checkout feture1

Git checkout -b feture2

Git branch -d feture1

Git branch -m feture1 branch1

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git branch b1

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git checkout -b b2
Switched to a new branch 'b2'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (b2)
$ git branch
  b1
* b2
  master

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (b2)
$ git branch -m b2 b3

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (b3)
$ git branch
  b1
* b3
  master

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (b3)
$ git branch b1
fatal: a branch named 'b1' already exists

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (b3)
$ git branch -d b1
Deleted branch b1 (was 6081abb).

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (b3)
$ git branch
* b3
  master
```


12) Git tag

Git tag tagname

Git checkout tagname

git tag -d tagname

git show tagname

git tag tagname <commit-id>

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ touch t1

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git add t1

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git commit -m "t1 created"
[master (root-commit) e92c79b] t1 created
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 t1

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git log --oneline
e92c79b (HEAD -> master) t1 created

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git tag v.1

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git log --oneline
e92c79b (HEAD -> master, tag: v.1) t1 created

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ touch t2

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git add t2

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git commit -m "t2 created"
[master 83742da] t2 created
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 t2
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git tag v.2
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git tag
v.1
v.2
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git show v.1
commit e92c79b4a2dfcb5af7cc01f530d0e40ad1913a04 (tag: v.1)
Author: shruthi <shruthir1017@gmail.com>
Date: Sat Aug 3 23:38:46 2024 +0530
```

```
t1 created
```

```
diff --git a/t1 b/t1
new file mode 100644
index 0000000..e69de29
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git tag -d v.1
Deleted tag 'v.1' (was e92c79b)
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git tag
v.2
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ touch t3
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git add t3
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git commit -m "t3 created"
[master 6081abb] t3 created
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 t3
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git log --oneline
6081abb (HEAD -> master) t3 created
83742da (tag: v.2) t2 created
e92c79b t1 created
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git tag v.3 6081abb
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git log --oneline
6081abb (HEAD -> master, tag: v.3) t3 created
83742da (tag: v.2) t2 created
e92c79b t1 created
```

13) Git merge

==> Assignment :

try to create master branch
in master branch create a file biggest3.c
in master branch create a file fact.c and give tag name
under master you create one more branch --> branch 1
in branch 1 create a file with code of reverse.c
again create a branch 2 in master branch
in branch 2 create a file fibonacci.called
merge branch 1 and branch 2 in master branch

Git merge <branchname>

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ vi big.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git add big.c
warning: in the working copy of 'big.c', LF will be replaced by CRLF the next time Git touches it

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git commit -m "biggest"
[master f373f3d] biggest
1 file changed, 1 insertion(+)
create mode 100644 big.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ vi fact.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git add fact.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git commit -m "factorial"
[master b0ad6f7] factorial
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 fact.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git tag v.1

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git checkout -b b1
Switched to a new branch 'b1'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (b1)
$ vi rev.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (b1)
$ vi add rev.c
```

```

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (b1)
$ git add rev.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (b1)
$ git commit -m "reverese"
[b1 e94dd08] reverese
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 rev.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (b1)
$ git checkout master
Switched to branch 'master'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git checkout -b b2
Switched to a new branch 'b2'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (b2)
$ vi fib.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (b2)
$ git add fib.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (b2)
$ git commit -m "fibinocci"
[b2 cc7b8cd] fibinocci
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 fib.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (b2)
$ git checkout master
Switched to branch 'master'

```

When we try to do git merge branch2 Merge commit message will open in vi editor do the cahnges if required and then save commit message
One extra commit added for merge in master branch with existing commits

```

Merge branch 'b2'
# Please enter a commit message to explain why this merge is necessary,
# especially if it merges an updated upstream into a topic branch.
#
# Lines starting with '#' will be ignored, and an empty message aborts
# the commit.
~
~
~

```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git merge b1
Updating b0ad6f7..e94dd08
Fast-forward
 rev.c | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 rev.c
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git log --oneline
e94dd08 (HEAD -> master, b1) reverse
b0ad6f7 (tag: v.1) factorial
f373f3d biggest
1342f98 bug fixed
0e52f41 file1 added
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git merge b2
Merge made by the 'ort' strategy.
 fib.c | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 fib.c
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git log --oneline
925e7a7 (HEAD -> master) Merged branch 'b2'
cc7b8cd (b2) fibonacci
e94dd08 (b1) reverse
b0ad6f7 (tag: v.1) factorial
f373f3d biggest
1342f98 bug fixed
0e52f41 file1 added
```

14) Git rebase

try to create master branch
in master branch create a file biggest3.c
in master branch create a file fact.c and give tag name
under master you create one more branch --> branch 1
in branch 1 create a file with code of reverse.c
again create a branch 2 in master branch
in branch 2 create a file fibbonacci.called

Git rebase branch-name

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (master)
$ touch big.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (master)
$ git add big.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (master)
$ git commit -m "biggest"
[master (root-commit) 3f2cdaa] biggest
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 big.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (master)
$ touch fact.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (master)
$ git add fact.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (master)
$ git commit -m "factorial"
[master ae7f24e] factorial
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 fact.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (master)
$ git tag v.2

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (master)
$ git checkout -b b1
Switched to a new branch 'b1'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (b1)
$ touch rev.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (b1)
$ git add rev.c
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (b1)
$ git commit -m "reverse"
[b1 d4bd613] reverse
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 rev.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (b1)
$ git checkout master
Switched to branch 'master'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (master)
$ git checkout -b b2
Switched to a new branch 'b2'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (b2)
$ touch fib.c
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (b2)
$ git add fib.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (b2)
$ git commit -m "fibonacci"
[b2 312d767] fibonacci
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 fib.c

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (b2)
$ git checkout master
Switched to branch 'master'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (master)
$ git log --oneline
ae7f24e (HEAD -> master, tag: v.2) factorial
3f2cdaa biggest

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (master)
$ git rebase b1
Successfully rebased and updated refs/heads/master.

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (master)
$ git log --oneline
d4bd613 (HEAD -> master, b1) reverse
ae7f24e (tag: v.2) factorial
3f2cdaa biggest

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (master)
$ git rebase b2
Successfully rebased and updated refs/heads/master.

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2) (master)
$ git log --oneline
c49dd0c (HEAD -> master) reverse
312d767 (b2) fibonacci
ae7f24e (tag: v.2) factorial
3f2cdaa biggest
```

15) Git cherry-pick <commit-id>

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (branch1)
$ git log
commit 74b193f495dc6444894216ac790892b138d428ed (HEAD -> branch1)
Author: shruthi <shruthir1017@gmail.com>
Date: Sat Aug 3 22:41:58 2024 +0530

    t2 and t3 added

commit 0e52f41f6d8f8788f02c3dff001ced00d793a46b (master)
Author: shruthi <shruthir1017@gmail.com>
Date: Sat Aug 3 22:26:49 2024 +0530

    file1 added

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (branch1)
$ git checkout -b branch2
Switched to a new branch 'branch2'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (branch2)
$ touch bugfix.txt

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (branch2)
$ git add bugfix.txt

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (branch2)
$ git commit -m "bug fixed"
[branch2 9af9ff1] bug fixed
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 bugfix.txt

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (branch2)
$ git log --oneline
9af9ff1 (HEAD -> branch2) bug fixed
74b193f (branch1) t2 and t3 added
0e52f41 (master) file1 added
```


Git cherry-pick

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (branch2)
$ git checkout branch1
Switched to branch 'branch1'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (branch1)
$ git cherry-pick 9af9ff1
[branch1 b62c024] bug fixed
Date: Sat Aug 3 22:43:14 2024 +0530
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 bugfix.txt

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (branch1)
$ git log --oneline
b62c024 (HEAD -> branch1) bug fixed
74b193f t2 and t3 added
0e52f41 (master) file1 added

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (branch1)
$ git checkout master
Switched to branch 'master'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git cherry-pick 9af9ff1
[master 1342f98] bug fixed
Date: Sat Aug 3 22:43:14 2024 +0530
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 bugfix.txt

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (master)
$ git log --oneline
1342f98 (HEAD -> master) bug fixed
0e52f41 file1 added
```

16) Git stash

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ vi index.txt

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git add .
warning: in the working copy of 'index.txt', LF will be replaced by CRLF the next time Git touches it

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git commit -m "index created"
[master (root-commit) fa6a2b9] index created
1 file changed, 1 insertion(+)
create mode 100644 index.txt

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git log --oneline
fa6a2b9 (HEAD -> master) index created

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git checkout -b branch1
Switched to a new branch 'branch1'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (branch1)
$ ls
index.txt
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (branch1)
$ git checkout master
Switched to branch 'master'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ vi index

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ vi index.txt

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git add .
warning: in the working copy of 'index.txt', LF will be replaced by CRLF the next time Git touches it

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git commit -m "index updated"
[master 2e0c2fe] index updated
2 files changed, 1 insertion(+)
create mode 100644 index

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git checkout branch1
Switched to branch 'branch1'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (branch1)
$ vi index.txt
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (branch1)
$ git checkout master
error: Your local changes to the following files would be overwritten by checkout:
    index.txt
Please commit your changes or stash them before you switch branches.
Aborting

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (branch1)
$ git stash
Saved working directory and index state WIP on branch1: fa6a2b9 index created

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (branch1)
$ git checkout master
Switched to branch 'master'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git checkout branch1
Switched to branch 'branch1'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (branch1)
$ git stash list
stash@{0}: WIP on branch1: fa6a2b9 index created

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (branch1)
$ git stash pop
On branch branch1
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:   index.txt

no changes added to commit (use "git add" and/or "git commit -a")
Dropped refs/stash@{0} (0b5a66b0e2579f315824f8565301b999cab5c8de)

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (branch1)
$ git stash list
```

17) Git squash-reduce number of commits by combining them

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git rebase -i HEAD~2
[detached HEAD 59d0a74] This is a combination of 2 commits.
Date: Wed Jul 31 23:43:58 2024 +0530
3 files changed, 1 insertion(+)
create mode 100644 file2
create mode 100644 index
Successfully rebased and updated refs/heads/master.

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git log
commit 59d0a74913c594dbc55b1ecf52ba815cad3ca744 (HEAD -> master)
Author: shruthi <shruthir1017@gmail.com>
Date: Wed Jul 31 23:43:58 2024 +0530

    This is a combination of 2 commits.

    index updated

    file2 updated
```

```
pick 2e0c2fe index updated
squash| d2be7b4 file2 updated

# Rebase fa6a2b9..d2be7b4 onto fa6a2b9 (2 commands)
#
# Commands:
# p, pick <commit> = use commit
# r, reword <commit> = use commit, but edit the commit message
# e, edit <commit> = use commit, but stop for amending
# s, squash <commit> = use commit, but meld into previous commit
# f, fixup [-C | -c] <commit> = like "squash" but keep only the previous
#                               commit's log message, unless -C is used, in which case
#                               keep only this commit's message; -c is same as -C but
#                               opens the editor
```

18) Git revert

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git revert e3a9fba
[master 1728b79] Revert "files added"
7 files changed, 2 insertions(+)
rename file1 => index (100%)
create mode 100644 index.txt
rename file3 => shru11 (100%)
rename file4 => shru12 (100%)
rename file5 => shru13 (100%)
create mode 100644 shru14
create mode 100644 shru15

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (2)/project2 (master)
$ git log
commit 1728b791b5cb030b615dfd98c0d9f29533b1c121 (HEAD -> master)
Author: shruthi <shruthir1017@gmail.com>
Date: Thu Aug 1 00:24:31 2024 +0530

    Revert "files added"

    This reverts commit e3a9fba26f1481b3c6dc1c011f71158b04193150.

commit e3a9fba26f1481b3c6dc1c011f71158b04193150
Author: shruthi <shruthir1017@gmail.com>
Date: Thu Aug 1 00:23:49 2024 +0530

    files added
```

Revert "files added"

This reverts commit e3a9fba26f1481b3c6dc1c011f71158b04193150.

```
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Changes to be committed:
#   renamed:   file1 -> index
#   new file:   index.txt
#   renamed:   file3 -> shru11
#   renamed:   file4 -> shru12
#   renamed:   file5 -> shru13
#   new file:   shru14
#   new file:   shru15
#
```

19) Git reset

Git reset --hard HEAD~2

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git log --oneline
f57bcbe (HEAD -> master) shru8 created
81276a7 shru7 created
b057046 shru6 created
9b0e297 shru5 created
adca836 shru4 created
9dc3a67 shru3 created
1801d63 shru2 created
ea3dc7e shru1 created

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git reset --hard HEAD~2
HEAD is now at b057046 shru6 created
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git log --oneline
b057046 (HEAD -> master) shru6 created
9b0e297 shru5 created
adca836 shru4 created
9dc3a67 shru3 created
1801d63 shru2 created
ea3dc7e shru1 created
```

Git reset --mixed HEAD~2

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git log --oneline
adca836 (HEAD -> master) shru4 created
9dc3a67 shru3 created
1801d63 shru2 created
ea3dc7e shru1 created

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git reset --mixed HEAD~2

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    shru3
    shru4
    shru5
    shru6

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

Git reset --soft HEAD~2

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git log --oneline
b057046 (HEAD -> master) shru6 created
9b0e297 shru5 created
adca836 shru4 created
9dc3a67 shru3 created
1801d63 shru2 created
ea3dc7e shru1 created

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git reset --soft HEAD~2

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git log --oneline
adca836 (HEAD -> master) shru4 created
9dc3a67 shru3 created
1801d63 shru2 created
ea3dc7e shru1 created

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   shru5
        new file:   shru6
```

20) Git show

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ touch file5

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git add file5

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git commit -m "file5 created"
[master 4ec39cf] file5 created
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file5

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git show
commit 4ec39cfb8c4259a447fbbdd17de5260144bbeb5e (HEAD -> master)
Author: shruthi <shruthir1017@gmail.com>
Date: Fri Aug 2 13:50:25 2024 +0530

    file5 created

diff --git a/file5 b/file5
new file mode 100644
index 0000000..e69de29
```

Git show <commit-id>

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git log --oneline
4ec39cf (HEAD -> master) file5 created
1801d63 shru2 created
ea3dc7e shru1 created

Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git show 4ec39cf
commit 4ec39cfb8c4259a447fbbdd17de5260144bbeb5e (HEAD -> master)
Author: shruthi <shruthir1017@gmail.com>
Date: Fri Aug 2 13:50:25 2024 +0530

    file5 created

diff --git a/file5 b/file5
new file mode 100644
index 0000000..e69de29
```

Git show - -stat (display latest commit details)

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/New folder/New folder (master)
$ git show --stat
commit 4ec39cfb8c4259a447fbbdd17de5260144bbeb5e (HEAD -> master)
Author: shruthi <shruthir1017@gmail.com>
Date: Fri Aug 2 13:50:25 2024 +0530

    file5 created

file5 | 0
1 file changed, 0 insertions(+), 0 deletions(-)
```


21) Git diff

Git diff --staged

(difference between working directory and staging area)

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (master)
$ git diff --staged
diff --git a/file2 b/file2
new file mode 100644
index 0000000..e69de29

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (master)
$ vi file2

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (master)
$ git add file2
warning: in the working copy of 'file2', LF will be replaced by CRLF the next time Git touches it

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (master)
$ git diff

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (master)
$ git diff --staged
diff --git a/file2 b/file2
new file mode 100644
index 0000000..64fe046
--- /dev/null
+++ b/file2
@@ -0,0 +1,2 @@
+ hello
+
```

Git diff <commit1> <commit2> <filename>

Changes between 2 commits for specific file

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (master)
$ git diff b214f88 e75c0de file2

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (master)
$ git diff b214f88 e75c0de file1
diff --git a/file1 b/file1
index e69de29..ce01362 100644
--- a/file1
+++ b/file1
@@ -0,0 +1 @@
+hello
```

Git diff <commit1> <commit2>

Changes between 2 commits

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (master)
$ git log --oneline
43bba66 (HEAD -> master) file 2 created
e75c0de file1 updated
b214f88 file1 created

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (master)
$ git diff b214f88 43bba66
diff --git a/file1 b/file1
index e69de29..ce01362 100644
--- a/file1
+++ b/file1
@@ -0,0 +1 @@
+hello
diff --git a/file2 b/file2
new file mode 100644
index 0000000..64fe046
--- /dev/null
+++ b/file2
@@ -0,0 +1,2 @@
+ hello
+
```

Git diff - -cached

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (master)
$ ls
file1 file2 file3 file4 file5

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    file5

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

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (master)
$ git add file5

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (master)
$ git diff --cached
diff --git a/file5 b/file5
new file mode 100644
index 0000000..e69de29
```

22) Git bisect

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (branch2)
$ git log --oneline
a5da2d6 (HEAD -> branch2, master) file5 updated
270bebe file 3 4 created
43bba66 file 2 created
e75c0de file1 updated
b214f88 file1 created

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (branch2)
$ git bisect start
status: waiting for both good and bad commits

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (branch2|BISECTING)
$ git bisect bad
status: waiting for good commit(s), bad commit known

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (branch2|BISECTING)
$ git bisect good 43bba66
Bisecting: 0 revisions left to test after this (roughly 0 steps)
[270bebe998beb2d875ce3d87da1edb4be72a554] file 3 4 created

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project ((270bebe...)|BISECTING)
$ git bisect bad
270bebe998beb2d875ce3d87da1edb4be72a554 is the first bad commit
commit 270bebe998beb2d875ce3d87da1edb4be72a554 (HEAD)
Author: shruthi <shruthir1017@gmail.com>
Date:   Fri Aug 2 14:15:17 2024 +0530

    file 3 4 created

file3 | 0
file4 | 0
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file3
create mode 100644 file4

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project ((270bebe...)|BISECTING)
$ git bisect good
270bebe998beb2d875ce3d87da1edb4be72a554 was both good and bad

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project ((270bebe...)|BISECTING)
$ git bisect reset
Previous HEAD position was 270bebe file 3 4 created
Switched to branch 'branch2'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project (branch2)
$ |
```

23) Git clone

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder
$ git clone git@github.com:shruthir1017/project1.git
Cloning into 'project1'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (5/5), done.
Receiving objects: 100% (8/8), done.
remote: Total 8 (delta 0), reused 0 (delta 0), pack-reused 0

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder
$ ls
project1/
```

24) Git push

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder
$ ls
project1/

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder
$ cd project1/


Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project1 (main)
$ ls
README.md  file1.txt

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project1 (main)
$ vi file2

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project1 (main)
$ git add file2
warning: in the working copy of 'file2', LF will be replaced by CRLF the next time Git touches it


Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project1 (main)
$ git commit -m "file2 added"
[main 40735be] file2 added
1 file changed, 2 insertions(+)
create mode 100644 file2




Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project1 (main)
$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 313 bytes | 313.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:shruthir1017/project1.git
5e7bd3a..40735be main -> main
```




project1
Public
Pin
Unwatch 1

main
3 Branches
0 Tags

Add file
Code


shruthir1017
file2 added
40735be · 1 minute ago
3 Commits


| | | |
|---|------------------|--------------|
|  README.md | Initial commit | 2 weeks ago |
|  file1.txt | Create file1.txt | 2 weeks ago |
|  file2 | file2 added | 1 minute ago |






[README](#)


25) Git pull

main
project1 /

Add file
...


shruthir1017
Create file3
d923695 · now
History

| Name | Last commit message | Last commit date |
|---|---------------------|------------------|
|  README.md | Initial commit | 2 weeks ago |
|  file1.txt | Create file1.txt | 2 weeks ago |
|  file2 | file2 added | 2 minutes ago |
|  file3 | Create file3 | now |

```

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project1 (main)
$ git pull
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 914 bytes | 83.00 KiB/s, done.
From github.com:shruthir1017/project1
   40735be..d923695  main       -> origin/main
Updating 40735be..d923695
Fast-forward
 file3 | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 file3

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder/project1 (main)
$ ls
README.md  file1.txt  file2      file3

```

26) Git remote

Git remote add origin <url-repo>

To add new remote repo

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3)
$ git init
Initialized empty Git repository in F:/devops class/New folder (3)/.git/

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote add origin git@github.com:shruthir1017/june-2024.git

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ ls

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote -v
origin  git@github.com:shruthir1017/june-2024.git (fetch)
origin  git@github.com:shruthir1017/june-2024.git (push)
```

Git remote show origin

To display details about remote repo

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote show origin
* remote origin
Fetch URL: git@github.com:shruthir1017/example.git
Push URL: git@github.com:shruthir1017/example.git
HEAD branch: main
Remote branch:
  main new (next fetch will store in remotes/origin)
```

Git remote add upstream <repo-url>

To additional remotes

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote add upstream git@github.com:shruthir1017/disha.git

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote -v
origin  git@github.com:shruthir1017/example.git (fetch)
origin  git@github.com:shruthir1017/example.git (push)
upstream      git@github.com:shruthir1017/disha.git (fetch)
upstream      git@github.com:shruthir1017/disha.git (push)
```

Git remote remove origin

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote remove origin

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote -v
upstream      git@github.com:shruthir1017/disha.git (fetch)
upstream      git@github.com:shruthir1017/disha.git (push)
```

Git remote -v (display url of remote repo)

Git remote (list all remotes)

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote -v
origin  git@github.com:shruthir1017/OPQ.git (fetch)
origin  git@github.com:shruthir1017/OPQ.git (push)
upstream      git@github.com:shruthir1017/disha.git (fetch)
upstream      git@github.com:shruthir1017/disha.git (push)

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote
origin
upstream
```

Git remote rename origin <new name>

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote rename origin abc

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote
abc
upstream
```

Git remote set-url origin <new-repo-url>

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote add origin git@github.com:shruthir1017/OPQ.git

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote add upstream git@github.com:shruthir1017/disha.git

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote -v
origin  git@github.com:shruthir1017/OPQ.git (fetch)
origin  git@github.com:shruthir1017/OPQ.git (push)
upstream      git@github.com:shruthir1017/disha.git (fetch)
upstream      git@github.com:shruthir1017/disha.git (push)

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ ^C

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote set-url origin git@github.com:shruthir1017/disha.git

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git remote -v
origin  git@github.com:shruthir1017/disha.git (fetch)
origin  git@github.com:shruthir1017/disha.git (push)
upstream      git@github.com:shruthir1017/disha.git (fetch)
upstream      git@github.com:shruthir1017/disha.git (push)
```

27) Git pull request

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ git clone git@github.com:shruthir1017/disha.git
Cloning into 'disha'...
remote: Enumerating objects: 19, done.
remote: Counting objects: 100% (19/19), done.
remote: Compressing objects: 100% (11/11), done.
remote: Total 19 (delta 2), reused 12 (delta 1), pack-reused 0
Receiving objects: 100% (19/19), done.
Resolving deltas: 100% (2/2), done.

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ ls
disha/

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3) (master)
$ cd disha/

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3)/disha (main)
$ ls
README.md  hello


Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3)/disha (main)
$ git checkout -b branch1
Switched to a new branch 'branch1'

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3)/disha (branch1)
$ vi file1
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3)/disha (branch1)
$ git add file1
warning: in the working copy of 'file1', LF will be replaced by CRLF the next time Git touches it

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3)/disha (branch1)
$ git commit -m "file1 added"
[branch1 549c32f] file1 added
1 file changed, 1 insertion(+)
create mode 100644 file1

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3)/disha (branch1)
$ git push origin branch1
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 300 bytes | 100.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'branch1' on GitHub by visiting:
remote:   https://github.com/shruthir1017/disha/pull/new/branch1
remote:
To github.com:shruthir1017/disha.git
 * [new branch]      branch1 -> branch1
```


 disha

Public

 Pin

 Unwatch 1

 **branch1** had recent pushes 2 minutes ago

Compare & pull request

 branch1

5 Branches

0 Tags


Go to file

Add file

<> Code

This branch is 1 commit ahead of main.











 Contribute

 Add a title



file1 added

Add a description

WritePreview

H B I        @   

code merged

 Markdown is supported  Paste, drop, or click to add files

Create pull request

Review

No review

Assignee

No one

Labels

None yet

Projects

None yet

Milestones

No milestones

Development

Use Cloak to automate

Helpful

Conversation 0 Commits 1 Checks 0 Files changed 1



shruthir1017 commented now

Owner ...

code merged



file1 added

549c32f



Require approval from specific reviewers before merging
[Rulesets](#) ensure specific people approve pull requests before they're merged.

Add rule X

Continuous integration has not been set up
[GitHub Actions](#) and [several other apps](#) can be used to automatically catch bugs and enforce style.

This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request

You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Conversation 0 Commits 1 Checks 0 Files changed 1



shruthir1017 commented 1 minute ago

Owner ...

code merged



file1 added

549c32f

shruthir1017 merged commit 5223459 into main now

Revert



Pull request successfully merged and closed
You're all set—the `branch1` branch can be safely deleted.

Delete branch

28) Git fetch

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3)/disha (branch1)
$ git clone git@github.com:shruthir1017/project1.git
Cloning into 'project1'...
remote: Enumerating objects: 14, done.
remote: Counting objects: 100% (14/14), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 14 (delta 2), reused 2 (delta 0), pack-reused 0
Receiving objects: 100% (14/14), done.
Resolving deltas: 100% (2/2), done.

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3)/disha (branch1)
$ ls
README.md  file1  hello  project1/








Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3)/disha (branch1)
$ cd project1/

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3)/disha/project1 (main)
$ ls
README.md  file1.txt  file2  file3

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3)/disha/project1 (main)
$ git fetch origin
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 908 bytes | 60.00 KiB/s, done.
From github.com:shruthir1017/project1
    d923695..2419472  main       -> origin/main

Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3)/disha/project1 (main)
$ git branch -r
  origin/HEAD -> origin/main
  origin/branch1
  origin/branch2
  origin/main
```

```
Shruthi@DESKTOP-3FATL5V MINGW64 /f/devops class/New folder (3)/disha/project1 (origin/branch1)
$ git push --set-upstream origin origin/branch1
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (9/9), 1.97 KiB | 1.97 MiB/s, done.
Total 9 (delta 2), reused 9 (delta 2), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
remote:
remote: Create a pull request for 'origin/branch1' on GitHub by visiting:
remote:   https://github.com/shruthir1017/project1/pull/new/origin/branch1
remote:
To github.com:shruthir1017/project1.git
 * [new branch]      origin/branch1 -> origin/branch1
branch 'origin/branch1' set up to track 'origin/origin/branch1'.
```

|  shruthir1017 Create file 4 | | 2419472 · 2 minutes ago |  History |
|--|---------------------|-------------------------|---|
| Name | Last commit message | Last commit date | |
|  README.md | Initial commit | 2 weeks ago | |
|  file 4 | Create file 4 | 2 minutes ago | |
|  file1.txt | Create file1.txt | 2 weeks ago | |
|  file2 | file2 added | 1 hour ago | |
|  file3 | Create file3 | 1 hour ago | |

branch1

3 Branches

0 Tags

Go to file

Add file

<> Code

This branch is 5 commits behind main.

shruthir1017

Create file2.txt

cf81415 · 2 weeks ago

2 Commits

| | | |
|-----------|------------------|-------------|
| README.md | Initial commit | 2 weeks ago |
| file2.txt | Create file2.txt | 2 weeks ago |

README

Create file2.txt #2

Open shruthir1017 wants to merge 1 commit into origin/branch1 from branch1

Conversation 0

Commits 1

Checks 0

Files changed 1

shruthir1017 commented now

No description provided.

Create file2.txt

shruthir1017 merged commit 5e164e0 into origin/branch1 now

Pull request successfully merged and closed

You're all set—the branch1 branch can be safely deleted.

Delete branch