my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows

$ git config --global user.name "deepa"

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows

$ git init

Initialized empty Git repository in C:/Users/my pc/Desktop/git geekyshows/.git/

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

No commits yet

Untracked files:

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

file.txt

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: file.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: file.txt

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: file.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: file.txt

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: file.txt

Untracked files:

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

prod1.txt

prod2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: file.txt

new file: prod1.txt

new file: prod2.txt

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: file.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$

$ git reset prod1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: file.txt

new file: prod2.txt

Untracked files:

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

prod1.txt

$prod1 file ko unstage area main move kar diya git reset cmd se

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file.txt prod1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: file.txt

new file: prod1.txt

new file: prod2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git rm --cached file.txt

rm 'file.txt'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: prod1.txt

new file: prod2.txt

Untracked files:

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

file.txt

$ rm cached se file ko unstage mai move kar diay , sabhi changes track hongey isme ab

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git diff

**diff --git a/file.txt b/file.txt**

**index 3bd251f..740d28d 100644**

**--- a/file.txt**

**+++ b/file.txt**

@@ -5,7 +5,6 @@ hello1

hello2

-lenevo

+Deepanshu

-Dell

$ kya changes hua ye usko show karegaa , it show actual changes , change hua par stage nahi hua git diff show karega aapka

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git diff --staged

**diff --git a/file.txt b/file.txt**

**new file mode 100644**

**index 0000000..740d28d**

**--- /dev/null**

**+++ b/file.txt**

@@ -0,0 +1,10 @@

+

+mobile prodcts

+

+hello1

+

+hello2

+

+Deepanshu

+

+

**diff --git a/prod1.txt b/prod1.txtnew file mode 100644**

$ stage hua par commit nahi hua , us changes ko show karega , git diff –staged

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git diff prod2.txt

**diff --git a/prod2.txt b/prod2.txt**

**index e69de29..aae0ec4 100644**

**--- a/prod2.txt**

**+++ b/prod2.txt**

@@ -0,0 +1 @@

+hello to all

$ change hua par stages nahi hau aapka

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: file.txt

new file: prod1.txt

new file: prod2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git log

fatal: your current branch 'master' does not have any commits yet

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "all files add to master branch"

[master (root-commit) 9ab0b54] all files add to master branch

3 files changed, 11 insertions(+)

create mode 100644 file.txt

create mode 100644 prod1.txt

create mode 100644 prod2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

nothing to commit, working tree clean

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git log

commit 9ab0b54f1693544fedfa44f828fbe24c4d2e8e4b (**HEAD** -> **master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:32:24 2025 +0530

all files add to master branch

$ 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: prod2.txt

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add prod2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "commit all"

[master d306c73] commit all

1 file changed, 2 insertions(+)

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git staus

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

The most similar command is

status

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

nothing to commit, working tree clean

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ 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: prod2.txt

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -a -m "prod2 modify"

[master e2d54e7] prod2 modify

1 file changed, 2 insertions(+)

$ modify or added sath main sab ek sath hua

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git log -p

commit e2d54e71afcd1377ef0ed884e8f9c2aab61fec69 (**HEAD** -> **master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:36:15 2025 +0530

prod2 modify

**diff --git a/prod2.txt b/prod2.txt**

**index 421b954..d9e7a7d 100644**

**--- a/prod2.txt**

**+++ b/prod2.txt**

@@ -1,3 +1,5 @@

hello to all

keyboard products

+

+mouse products

commit d306c73e0dac09d5942ed29796020c2b4b206162

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:34:11 2025 +0530

commit all

**diff --git a/prod2.txt b/prod2.txt**

**index aae0ec4..421b954 100644**

**--- a/prod2.txt**

**+++ b/prod2.txt**

@@ -1 +1,3 @@

hello to all

+

+keyboard products

commit 9ab0b54f1693544fedfa44f828fbe24c4d2e8e4b

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:32:24 2025 +0530

all files add to master branch

**diff --git a/file.txt b/file.txt**

**new file mode 100644**

**index 0000000..740d28d**

**--- /dev/null**

**+++ b/file.txt**

@@ -0,0 +1,10 @@

+

+mobile prodcts

+

+hello1

:

$ show the actual changes in our code

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git reflog

e2d54e7 (**HEAD** -> **master**) HEAD@{0}: commit: prod2 modify

d306c73 HEAD@{1}: commit: commit all

9ab0b54 HEAD@{2}: commit (initial): all files add to master branch

$ unstage se untrack hote hai

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

nothing to commit, working tree clean

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git rm prod2.txt

rm 'prod2.txt'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

Changes to be committed:

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

deleted: prod2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git mv prod1.txt p1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

Changes to be committed:

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

renamed: prod1.txt -> p1.txt

deleted: prod2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

Changes to be committed:

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

renamed: prod1.txt -> p1.txt

deleted: prod2.txt

Untracked files:

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

file2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git reset HEAD file2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

Changes to be committed:

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

renamed: prod1.txt -> p1.txt

deleted: prod2.txt

Untracked files:

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

file2.txt

# unsate using git reset

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git reset HEAD -p

**diff --git a/file2.txt b/file2.txt**

**new file mode 100644**

**index 0000000..e69de29**

**(1/1) Unstage addition [y,n,q,a,d,?]?** y

**diff --git a/p1.txt b/p1.txt**

**new file mode 100644**

**index 0000000..e69de29**

**(1/1) Unstage addition [y,n,q,a,d,?]?** y

**diff --git a/prod1.txt b/prod1.txt**

**deleted file mode 100644**

**index e69de29..0000000**

**(1/1) Unstage deletion [y,n,q,a,d,p,?]?** y

**diff --git a/prod2.txt b/prod2.txt**

**deleted file mode 100644**

**index d9e7a7d..0000000**

**--- a/prod2.txt**

**+++ /dev/null**

@@ -1,5 +0,0 @@

-hello to all

-

-keyboard products

-

-mouse products

**(1/1) Unstage deletion [y,n,q,a,d,p,?]?** n

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ ^C

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

Changes to be committed:

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

deleted: prod2.txt

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: prod1.txt

Untracked files:

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

file2.txt

p1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git log

commit e2d54e71afcd1377ef0ed884e8f9c2aab61fec69 (**HEAD** -> **master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:36:15 2025 +0530

prod2 modify

commit d306c73e0dac09d5942ed29796020c2b4b206162

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:34:11 2025 +0530

commit all

commit 9ab0b54f1693544fedfa44f828fbe24c4d2e8e4b

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:32:24 2025 +0530

all files add to master branch

**$ ROLLBACK KARNA**

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows

$ git init

Initialized empty Git repository in C:/Users/my pc/Desktop/git geekyshows/.git/

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

No commits yet

Untracked files:

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

file1.txt

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ ^C

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file1 commit"

[master (root-commit) 14850ba] file1 commit

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

create mode 100644 file1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file2 added"

[master 727a487] file2 added

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

create mode 100644 file2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file3.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file3 added"

[master b10c54e] file3 added

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

create mode 100644 file3.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file4.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file4 added"

[master b80de34] file4 added

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

create mode 100644 file4.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git log

commit b80de346e6ef5938dbdb7f079920f4b77ec9713c (**HEAD** -> **master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:52:12 2025 +0530

file4 added

commit b10c54ee173d522199089d32317f570ac0424f56

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:51:51 2025 +0530

file3 added

commit 727a487395cd1b3ff7cde3f0864a8157a0d07fba

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:51:35 2025 +0530

file2 added

commit 14850ba0587aa77c9601b40ea5332244ebfc89ec

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:50:31 2025 +0530

file1 commit

git revert HEAD – FILE4 commit delted

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file3.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

Revert "file4 added"

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git log

commit 4578c9167e6a54ae7779002e752b8ece1d7b1b69 (**HEAD** -> **master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:53:22 2025 +0530

Revert "file4 added"

This reverts commit b80de346e6ef5938dbdb7f079920f4b77ec9713c.

commit b80de346e6ef5938dbdb7f079920f4b77ec9713c

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:52:12 2025 +0530

file4 added

commit b10c54ee173d522199089d32317f570ac0424f56

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:51:51 2025 +0530

file3 added

commit 727a487395cd1b3ff7cde3f0864a8157a0d07fba

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:51:35 2025 +0530

file2 added

commit 14850ba0587aa77c9601b40ea5332244ebfc89ec

Author: Deepabshu <dk0962422@gmail.com>

Date: Fri Feb 7 23:50:31 2025 +0530

file1 commit

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git reflog

4578c91 (**HEAD** -> **master**) HEAD@{0}: revert: Revert "file4 added"

b80de34 HEAD@{1}: commit: file4 added

b10c54e HEAD@{2}: commit: file3 added

727a487 HEAD@{3}: commit: file2 added

14850ba HEAD@{4}: commit (initial): file1 commit

**$ show commit history**

git revert file2

# toh f2 delete , file2 delete , rollback kar diya hamen

# TAG IN GITHUB

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows

$ git init

Initialized empty Git repository in C:/Users/my pc/Desktop/git geekyshows/.git/

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file1 added"

[master (root-commit) 2aa79a0] file1 added

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

create mode 100644 file1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

Untracked files:

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

file2.txt

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

Changes to be committed:

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

new file: file2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file2 added"

[master f85e9fb] file2 added

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

create mode 100644 file2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file3.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file3 added"

[master c0c1a39] file3 added

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

create mode 100644 file3.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

Untracked files:

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

file4.txt

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file4.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file4 added"

[master 962f486] file4 added

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

create mode 100644 file4.txt

$ tag bookmark lagana hai hame toh use karo

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git tag v1

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git log

commit 962f4860cd5ee17ec3c88305b1c1ad6e6165f27b (**HEAD** -> **master**, **tag: v1**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:02:39 2025 +0530

file4 added

commit c0c1a392700908de8a24268bb09234ac356d2f56

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:01:57 2025 +0530

file3 added

commit f85e9fbe12ffa4e289ca332c3f505530ab744203

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:01:14 2025 +0530

file2 added

commit 2aa79a00eb8826b9433966573d346e5dc13afc43

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:00:29 2025 +0530

file1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$

$ tag point carete karta hai , previous bookmark ka liye

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add file5.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

Changes to be committed:

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

new file: file5.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file5 added"

[master bc928b5] file5 added

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

create mode 100644 file5.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git reflog

bc928b5 (**HEAD** -> **master**) HEAD@{0}: commit: file5 added

962f486 (**tag: v1**) HEAD@{1}: commit: file4 added

c0c1a39 HEAD@{2}: commit: file3 added

f85e9fb HEAD@{3}: commit: file2 added

2aa79a0 HEAD@{4}: commit (initial): file1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git checkout v1

Note: switching to 'v1'.

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 962f486 file4 added

$ stable version ka tag bana sakte ho

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows ((f85e9fbe12ffa4e289ca332c3f505530ab744203))

$ git log

commit 962f4860cd5ee17ec3c88305b1c1ad6e6165f27b (**HEAD**, **tag: v1**, **tag: f85e9fbe12ffa4e28**

**9ca332c3f505530ab744203**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:02:39 2025 +0530

file4 added

commit c0c1a392700908de8a24268bb09234ac356d2f56

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:01:57 2025 +0530

file3 added

commit f85e9fbe12ffa4e289ca332c3f505530ab744203

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:01:14 2025 +0530

file2 added

commit 2aa79a00eb8826b9433966573d346e5dc13afc43

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:00:29 2025 +0530

file1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows ((f85e9fbe12ffa4e289ca332c3f505530ab744203))

$ git tag v0.1 f85e9fbe12ffa4e289ca332c3f505530ab744203

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows ((f85e9fbe12ffa4e289ca332c3f505530ab744203))

$ git checkout v0.1

Previous HEAD position was 962f486 file4 added

HEAD is now at f85e9fb file2 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows ((v0.1))

$ git log

commit f85e9fbe12ffa4e289ca332c3f505530ab744203 (**HEAD**, **tag: v0.1**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:01:14 2025 +0530

file2 added

commit 2aa79a00eb8826b9433966573d346e5dc13afc43

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:00:29 2025 +0530

file1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows ((v0.1))

$ git tag -d v0.1

Deleted tag 'v0.1' (was f85e9fb)

$ BRACNH

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows

$ git init

Initialized empty Git repository in C:/Users/my pc/Desktop/git geekyshows/.git/

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

No commits yet

Untracked files:

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

file1.txt

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file1 added"

[master (root-commit) a6b153d] file1 added

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

create mode 100644 file1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch

\* master

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

Untracked files:

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

file2.txt

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file2 added"

[master 499135f] file2 added

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

create mode 100644 file2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch

\* master

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ 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: file1.txt

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

Changes to be committed:

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

modified: file1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file1 added"

[master 1bcf026] file1 added

1 file changed, 1 insertion(+)

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch feature1

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git checkout feature1

Switched to branch 'feature1'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch feature1

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git checkout feature1

Switched to branch 'feature1'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ ^C

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git status

On branch feature1

Untracked files:

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

file3.txt

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git add file3.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git commit -m "file3 added"

[feature1 00343bd] file3 added

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

create mode 100644 file3.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git log

commit 00343bd11b142883fb45311b9aa28d1674b72949 (**HEAD** -> **feature1**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:20:18 2025 +0530

file3 added

commit 1bcf026db2f708c13ab689c683dfde044a332d7c (**master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:18:23 2025 +0530

file1 added

commit 499135f71c4357f7b431b1828047a4d770b6d290

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:16:47 2025 +0530

file2 added

commit a6b153d0e77d811ef1a77b5de61d97442df2b7fa

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:16:01 2025 +0530

file1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git checkout master

Switched to branch 'master'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git checkout feature1

Switched to branch 'feature1'

g

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git checkout -b feature2

Switched to a new branch 'feature2'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git checkout -b feature2

Switched to a new branch 'feature2'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ ^C

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ git status

On branch feature2

Untracked files:

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

feature2.txt

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ git add feature2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ git commit -m "feature2 file

> ^C

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ git commit -m "Feature2 file"

[feature2 180279b] Feature2 file

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

create mode 100644 feature2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ git log

commit 180279bcbb73356ef5158c288cd5d11b6829bb9b (**HEAD** -> **feature2**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:23:02 2025 +0530

Feature2 file

commit 00343bd11b142883fb45311b9aa28d1674b72949 (**feature1**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:20:18 2025 +0530

file3 added

commit 1bcf026db2f708c13ab689c683dfde044a332d7c (**master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:18:23 2025 +0530

file1 added

commit 499135f71c4357f7b431b1828047a4d770b6d290

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:16:47 2025 +0530

file2 added

commit a6b153d0e77d811ef1a77b5de61d97442df2b7fa

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:16:01 2025 +0530

file1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ git checkout master

Switched to branch 'master'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git checkout feature1

Switched to branch 'feature1'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git checkout feature2

Switched to branch 'feature2'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ git branch -d feature2

error: cannot delete branch 'feature2' used by worktree at 'C:/Users/my pc/Desktop/git geekys

hows'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ git checkout master

Switched to branch 'master'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch -d feature2

error: the branch 'feature2' is not fully merged

hint: If you are sure you want to delete it, run 'git branch -D feature2'

hint: Disable this message with "git config advice.forceDeleteBranch false"

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch

feature1

feature2

\* master

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch -D feature2

Deleted branch feature2 (was 180279b).

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git checkout feature1

Switched to branch 'feature1

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git checkout feature1

Already on 'feature1'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git checkout master

Switched to branch 'master'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch

feature1

\* master

g

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git merge feature1

Updating 1bcf026..00343bd

Fast-forward

file3.txt | 0

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

create mode 100644 file3.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch -d feature1

Deleted branch feature1 (was 00343bd).

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git log

commit 00343bd11b142883fb45311b9aa28d1674b72949 (**HEAD** -> **master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:20:18 2025 +0530

file3 added

commit 1bcf026db2f708c13ab689c683dfde044a332d7c

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:18:23 2025 +0530

file1 added

commit 499135f71c4357f7b431b1828047a4d770b6d290

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:16:47 2025 +0530

file2 added

commit a6b153d0e77d811ef1a77b5de61d97442df2b7fa

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:16:01 2025 +0530

file1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch feature1

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch feature2

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch

feature1

feature2

\* master

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git checkout feature1

Switched to branch 'feature1'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git commit -m "fiel1 added to feature1"

[feature1 f9c465c] fiel1 added to feature1

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

create mode 100644 feature1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git checkout feature2

Switched to branch 'feature2'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git checkout feature2

Switched to branch 'feature2'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ ^C

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ git commit -m "file2 added to feature branch"

[feature2 259f7c8] file2 added to feature branch

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

create mode 100644 feature2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ git log

commit 259f7c8d77b16c4af6213c5f3ca2aa885ffef916 (**HEAD** -> **feature2**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:32:39 2025 +0530

file2 added to feature branch

commit 00343bd11b142883fb45311b9aa28d1674b72949 (**master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:20:18 2025 +0530

file3 added

commit 1bcf026db2f708c13ab689c683dfde044a332d7c

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:18:23 2025 +0530

file1 added

commit 499135f71c4357f7b431b1828047a4d770b6d290

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:16:47 2025 +0530

file2 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ git rebase feature1

Successfully rebased and updated refs/heads/feature2.

$ feature1 ka sare data feature2 barnch main aagay a

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ git checkout feature1

Switched to branch 'feature1'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ ^C

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git checkout feature2

Switched to branch 'feature2'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature2)

$ ^C

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows

$ git init

Initialized empty Git repository in C:/Users/my pc/Desktop/git geekyshows/.git/

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file1 added:

> ^C

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file1 added"

[master (root-commit) d864e28] file1 added

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

create mode 100644 file1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch

\* master

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch feature1

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git checkout feature1

Switched to branch 'feature1'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git commit -m "Feature1 added"

[feature1 ca495ff] Feature1 added

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

create mode 100644 feature1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git log

commit ca495ff2b7ce16f89a7d6547ab8ebca2d5ec353f (**HEAD** -> **feature1**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:36:55 2025 +0530

Feature1 added

commit d864e28a5d92278fb692fb6d15ad6b36983b011b (**master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:35:50 2025 +0530

file1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git status

On branch feature1

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: feature1.txt

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git add feature1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git status

On branch feature1

Changes to be committed:

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

modified: feature1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git checkout master

error: Your local changes to the following files would be overwritten by checkou

t:

feature1.txt

Please commit your changes or stash them before you switch branches.

Aborting

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git stash

Saved working directory and index state WIP on feature1: ca495ff Feature1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git checkout master

Switched to branch 'master'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git checkout feature1

Switched to branch 'feature1'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git stash pop

On branch feature1

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: feature1.txt

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

Dropped refs/stash@{0} (f76df7c783c08900d11003ac5c9f74d16fcced0c)

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$

**$ git stash save karta hai stash changes ko , aur git stash pop ko changes wapas se dekh sakte hoo**

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git stash list

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git stash

Saved working directory and index state WIP on feature1: ca495ff Feature1 added

**$ REMOTE REPOSITORY**

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows

$ git init

Initialized empty Git repository in C:/Users/my pc/Desktop/git geekyshows/.git/

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

No commits yet

Untracked files:

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

file1.txt

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file1 added"

[master (root-commit) adfb722] file1 added

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

create mode 100644 file1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git log

commit adfb722d5ff2e8cbbcc91403de2e9e1fe8dac5ec (**HEAD** -> **master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:53:30 2025 +0530

file1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git branch

\* master

g

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git remote add origin https://github.com/deepanshu1334/proj1.git

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ 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

To have this happen automatically for branches without a tracking

upstream, see 'push.autoSetupRemote' in 'git help config'.

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git remote -v

origin https://github.com/deepanshu1334/proj1.git (fetch)

origin https://github.com/deepanshu1334/proj1.git (push)

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git remote show origin

\* remote origin

Fetch URL: https://github.com/deepanshu1334/proj1.git

Push URL: https://github.com/deepanshu1334/proj1.git

HEAD branch: (unknown)

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git push -u origin master

Enumerating objects: 3, done.

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

Writing objects: 100% (3/3), 211 bytes | 19.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)

To https://github.com/deepanshu1334/proj1.git

\* [new branch] master -> master

branch 'master' set up to track 'origin/master'.

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git status

On branch master

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

Untracked files:

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

file2.txt

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file2 added"

[master 4550e6e] file2 added

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

create mode 100644 file2.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git log

commit 4550e6e2e0ae921df48305ea724b0fd93eedc3dd (**HEAD** -> **master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:57:12 2025 +0530

file2 added

commit adfb722d5ff2e8cbbcc91403de2e9e1fe8dac5ec (**origin/master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:53:30 2025 +0530

file1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git push

Enumerating objects: 3, done.

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

Delta compression using up to 4 threads

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

Writing objects: 100% (2/2), 237 bytes | 237.00 KiB/s, done.

Total 2 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)

To https://github.com/deepanshu1334/proj1.git

adfb722..4550e6e master -> master

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git checkout feature1

Switched to branch 'feature1'

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git commit -m "feature1 added:

> ^C

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git commit -m "feature1 added"

[feature1 a008e86] feature1 added

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

create mode 100644 feature1.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git status

On branch feature1

nothing to commit, working tree clean

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git log

commit a008e860d9af44c57e5a7b49bfc375daf3c886dc (**HEAD** -> **feature1**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:59:18 2025 +0530

feature1 added

commit 4550e6e2e0ae921df48305ea724b0fd93eedc3dd (**origin/master**, **master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:57:12 2025 +0530

file2 added

commit adfb722d5ff2e8cbbcc91403de2e9e1fe8dac5ec

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:53:30 2025 +0530

file1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git branch

\* feature1

master

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git remote -v

origin https://github.com/deepanshu1334/proj1.git (fetch)

origin https://github.com/deepanshu1334/proj1.git (push)

i

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git push origin feature1

Enumerating objects: 3, done.

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

Delta compression using up to 4 threads

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

Writing objects: 100% (2/2), 246 bytes | 123.00 KiB/s, done.

Total 2 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)

remote:

remote: Create a pull request for 'feature1' on GitHub by visiting:

remote: https://github.com/deepanshu1334/proj1/pull/new/feature1

remote:

To https://github.com/deepanshu1334/proj1.git

\* [new branch] feature1 -> feature1

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git branch

\* feature1

master

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (feature1)

$ git checkout master

Switched to branch 'master'

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

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ ^C

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git tag --list

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git add .

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git commit -m "file3 added"

[master b81b02a] file3 added

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

create mode 100644 file3.txt

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git log

commit b81b02aee9f4114fc00570441e1dd2a5dfa087ec (**HEAD** -> **master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 01:02:15 2025 +0530

file3 added

commit 4550e6e2e0ae921df48305ea724b0fd93eedc3dd (**origin/master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:57:12 2025 +0530

file2 added

commit adfb722d5ff2e8cbbcc91403de2e9e1fe8dac5ec

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:53:30 2025 +0530

file1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git tag b81b02aee9f4114fc00570441e1dd2a5dfa087ec

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git log

commit b81b02aee9f4114fc00570441e1dd2a5dfa087ec (**HEAD** -> **master**, **tag: b81b02aee9f4114fc**

**00570441e1dd2a5dfa087ec**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 01:02:15 2025 +0530

file3 added

commit 4550e6e2e0ae921df48305ea724b0fd93eedc3dd (**origin/master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:57:12 2025 +0530

file2 added

commit adfb722d5ff2e8cbbcc91403de2e9e1fe8dac5ec

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:53:30 2025 +0530

file1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git tag -d b81b02aee9f4114fc

00570441e1dd2a5dfa087ec

error: tag 'b81b02aee9f4114fc' not found.

bash: 00570441e1dd2a5dfa087ec: command not found

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git tag -d b81b02aee9f4114fc00570441e1dd2a5dfa087ec

Deleted tag 'b81b02aee9f4114fc00570441e1dd2a5dfa087ec' (was b81b02a)

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git tag ^C

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git tag v1 adfb722d5ff2e8cbbcc91403de2e9e1fe8dac5ec

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git log

commit b81b02aee9f4114fc00570441e1dd2a5dfa087ec (**HEAD** -> **master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 01:02:15 2025 +0530

file3 added

commit 4550e6e2e0ae921df48305ea724b0fd93eedc3dd (**origin/master**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:57:12 2025 +0530

file2 added

commit adfb722d5ff2e8cbbcc91403de2e9e1fe8dac5ec (**tag: v1**)

Author: Deepabshu <dk0962422@gmail.com>

Date: Sat Feb 8 00:53:30 2025 +0530

file1 added

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows (master)

$ git push origin tag v1

Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)

To https://github.com/deepanshu1334/proj1.git

\* [new tag] v1 -> v1

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows

$ git fetch

fatal: not a git repository (or any of the parent directories): .git

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows

$ git push

fatal: not a git repository (or any of the parent directories): .git

my pc@DESKTOP-BA6T73K MINGW64 ~/Desktop/git geekyshows

$

git clone repo-url

.gitignore – nodemoduel ,env , tempfiles , don’t want to push to giyshub

.gitignore - .env , kuch bhi likho isme koi stage or na hi change hoga aapka

Git add .gitignore

Git commit –m “cmmit”

Git statjs

GIT PUSH

Ye ignore kar dega gitignore kar dega aap iskeo

BRANCHING STARGAGEY GEEKYSHOWS

**MSTER SE kabhi relase nahi karenge , phir master phir relase barnach banao usko relase to customer**

**HOT FIX URGETN BASIS PA MERGER KARENGE HAM APNE CHANGES KOO**

Initialized empty Git repository in C:/Users/my pc/Desktop/git geekyshows/branching/.git/

PS C:\Users\my pc\Desktop\git geekyshows\branching> git add .

PS C:\Users\my pc\Desktop\git geekyshows\branching> git commit -m "commit-all"

[master (root-commit) 208f17a] commit-all

1 file changed, 12 insertions(+)

create mode 100644 index.html

PS C:\Users\my pc\Desktop\git geekyshows\branching> git remote add origin https://github.com/deepanshu1334/geeks.git

origin https://github.com/deepanshu1334/geeks.git (push)

PS C:\Users\my pc\Desktop\git geekyshows\branching> 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

To have this happen automatically for branches without a tracking

upstream, see 'push.autoSetupRemote' in 'git help config'.

PS C:\Users\my pc\Desktop\git geekyshows\branching> git push --set-upstream origin master

Enumerating objects: 3, done.

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

Delta compression using up to 4 threads

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

Writing objects: 100% (3/3), 388 bytes | 97.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)

To https://github.com/deepanshu1334/geeks.git

\* [new branch] master -> master

branch 'master' set up to track 'origin/master'.

PS C:\Users\my pc\Desktop\git geekyshows\branching>

#change tile

PS C:\Users\my pc\Desktop\git geekyshows\branching> git add .

PS C:\Users\my pc\Desktop\git geekyshows\branching> git commit -m "add title"

[master 91713e2] add title

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

PS C:\Users\my pc\Desktop\git geekyshows\branching> git push

PS C:\Users\my pc\Desktop\git geekyshows\branching> git checkout master

Switched to branch 'master'

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

PS C:\Users\my pc\Desktop\git geekyshows\branching> git checkout relase

Switched to branch 'relase'

PS C:\Users\my pc\Desktop\git geekyshows\branching> git branch

master

\* relase

PS C:\Users\my pc\Desktop\git geekyshows\branching> git push origin relase

Counting objects: 100% (4/4), done.

Delta compression using up to 4 threads

Writing objects: 100% (3/3), 279 bytes | 69.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)

remote:

remote: Create a pull request for 'relase' on GitHub by visiting:

remote: https://github.com/deepanshu1334/geeks/pull/new/relase

remote:

To https://github.com/deepanshu1334/geeks.git

\* [new branch] relase -> relase

PS C:\Users\my pc\Desktop\git geekyshows\branching> git checkout master

PS C:\Users\my pc\Desktop\git geekyshows\branching> 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: index.html

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

PS C:\Users\my pc\Desktop\git geekyshows\branching> git add .

PS C:\Users\my pc\Desktop\git geekyshows\branching> git commit -m "change after relase1"

[master 652a3a0] change after relase1

1 file changed, 3 insertions(+)

PS C:\Users\my pc\Desktop\git geekyshows\branching> 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), 337 bytes | 4.00 KiB/s, done.

Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)

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

To https://github.com/deepanshu1334/geeks.git

91713e2..652a3a0 master -> master

PS C:\Users\my pc\Desktop\git geekyshows\branching>

PS C:\Users\my pc\Desktop\git geekyshows\branching> git branch feature1

PS C:\Users\my pc\Desktop\git geekyshows\branching> git branch

feature1

\* master

relase

PS C:\Users\my pc\Desktop\git geekyshows\branching> git checkout feature1

Switched to branch 'feature1'

PS C:\Users\my pc\Desktop\git geekyshows\branching> git branch

\* feature1

master

relase

PS C:\Users\my pc\Desktop\git geekyshows\branching> git add .

PS C:\Users\my pc\Desktop\git geekyshows\branching> git commit -m "feature1 add"

[feature1 3ac2431] feature1 add

1 file changed, 3 insertions(+)

create mode 100644 style.css

PS C:\Users\my pc\Desktop\git geekyshows\branching> git checkout master

Switched to branch 'master'

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

PS C:\Users\my pc\Desktop\git geekyshows\branching> git checkout relase

Switched to branch 'relase'

PS C:\Users\my pc\Desktop\git geekyshows\branching> git checkout feature1

Switched to branch 'feature1'

PS C:\Users\my pc\Desktop\git geekyshows\branching>

PS C:\Users\my pc\Desktop\git geekyshows\branching> git branch

\* feature1

master

relase

PS C:\Users\my pc\Desktop\git geekyshows\branching> git add .

PS C:\Users\my pc\Desktop\git geekyshows\branching> git commit -m "color add in feature1"

[feature1 d7a0360] color add in feature1

1 file changed, 1 insertion(+)

PS C:\Users\my pc\Desktop\git geekyshows\branching> git push origin feature1

PS C:\Users\my pc\Desktop\git geekyshows\branching> git checkout master

Switched to branch 'master'

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

PS C:\Users\my pc\Desktop\git geekyshows\branching> git merge feature1

Updating 652a3a0..d7a0360

Fast-forward

style.css | 4 ++++

1 file changed, 4 insertions(+)

create mode 100644 style.css

PS C:\Users\my pc\Desktop\git geekyshows\branching> git branch

feature1

\* master

relase

PS C:\Users\my pc\Desktop\git geekyshows\branching> git branch -d feature1

Deleted branch feature1 (was d7a0360).

PS C:\Users\my pc\Desktop\git geekyshows\branching> git add .

PS C:\Users\my pc\Desktop\git geekyshows\branching> git commit -m "update latest1"

[master 5225108] update latest1

1 file changed, 1 insertion(+)

PS C:\Users\my pc\Desktop\git geekyshows\branching> git push origin master

Enumerating objects: 5, done.

PS C:\Users\my pc\Desktop\git geekyshows\branching> git branch relase2

PS C:\Users\my pc\Desktop\git geekyshowsclc\branching> git checkout relase2

Switched to branch 'relase2'

PS C:\Users\my pc\Desktop\git geekyshows\branching> git add .

PS C:\Users\my pc\Desktop\git geekyshows\branching> git commit -m "realse2 test"

[relase2 f716d71] realse2 test

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

create mode 100644 relase2test

PS C:\Users\my pc\Desktop\git geekyshows\branching> git push origin relase2

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), 309 bytes | 154.00 KiB/s, done.

Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)

remote:

remote: Create a pull request for 'relase2' on GitHub by visiting:

remote: https://github.com/deepanshu1334/geeks/pull/new/relase2

remote:

To https://github.com/deepanshu1334/geeks.git

\* [new branch] relase2 -> relase2

PS C:\Users\my pc\Desktop\git geekyshows\branching> git branch

\* master

relase

relase2

PS C:\Users\my pc\Desktop\git geekyshows\branching> git branch relase3

PS C:\Users\my pc\Desktop\git geekyshows\branching> git add .

PS C:\Users\my pc\Desktop\git geekyshows\branching> git commit -m "relase3 push"

[master f228d10] relase3 push

1 file changed, 1 insertion(+)

PS C:\Users\my pc\Desktop\git geekyshows\branching> git push origin relase3

Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)

remote:

remote: Create a pull request for 'relase3' on GitHub by visiting:

**🌟 Git Branching Workflow with Commands**

**Step 1: Initialize the Repository (Skip if already initialized)**

bash

CopyEdit

git init

git remote add origin <your-repo-url> # Link to GitHub/GitLab/Bitbucket

git pull origin master # Ensure you have the latest master branch

**Step 2: Create and Work on the develop Branch**

bash

CopyEdit

git checkout -b develop # Create and switch to 'develop' branch

git push origin develop # Push 'develop' to remote

💡 develop will contain all tested and stable features before moving to production.

**Step 3: Create a Feature Branch and Work on a Feature**

bash

CopyEdit

git checkout -b feature/new-feature # Create feature branch

# Make changes and add files

git add .

git commit -m "Added new feature"

git push origin feature/new-feature # Push to remote

**Step 4: Merge Feature Branch into develop**

bash

CopyEdit

git checkout develop # Switch to develop branch

git merge feature/new-feature # Merge feature into develop

git push origin develop # Push updates to remote

git branch -d feature/new-feature # Delete local feature branch

git push origin --delete feature/new-feature # Delete remote feature branch

**Step 5: Create a Release Branch for Deployment**

bash

CopyEdit

git checkout -b release/v1.0 # Create a release branch from develop

git push origin release/v1.0 # Push to remote

💡 release branches are used for final testing before production.

**Step 6: Merge release into master (Production)**

bash

CopyEdit

git checkout master # Switch to master

git merge release/v1.0 # Merge release into master

git push origin master # Push production-ready code

**Step 7: Keep develop Up-to-Date**

bash

CopyEdit

git checkout develop

git merge master # Merge production changes back to develop

git push origin develop

💡 This ensures that develop has the latest production code.

**Step 8: Delete the Release Branch**

bash

CopyEdit

git branch -d release/v1.0 # Delete local release branch

git push origin --delete release/v1.0 # Delete remote release branch

**🎯 Final Git Branching Flow**

sql

CopyEdit

master <-- (Production-Ready Code)

|

├── develop <-- (Stable Features Before Release)

| ├── feature/new-feature <-- (Development in Progress)

|

├── release/v1.0 <-- (Production Testing)

🔥 **With this strategy:** ✅ master always contains stable production code.  
✅ develop is used for integrating and testing new features.  
✅ feature/\* branches are used for development.  
✅ release/\* branches are used for final testing before production.

🚀 **Now you have a professional Git branching strategy!**