itadmin@DESKTOP-RRNPTP1 MINGW64 ~

$ git -- version

unknown option: --

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

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

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

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

<command> [<args>]

itadmin@DESKTOP-RRNPTP1 MINGW64 ~

$ git --version

git version 2.30.1.windows.1

itadmin@DESKTOP-RRNPTP1 MINGW64 ~

$ mkdir ProjectEureka

itadmin@DESKTOP-RRNPTP1 MINGW64 ~

$ cd ProjectEureka

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka

$ ls

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka

$ touch about.html

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka

$ touch testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka

$ ls

about.html testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka

$ touch {a..e}.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka

$ ls

a.txt about.html b.txt c.txt d.txt e.txt testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka

$ git init

Initialized empty Git repository in C:/Users/itadmin/ProjectEureka/.git/

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ ls

a.txt about.html b.txt c.txt d.txt e.txt testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ ls -l

total 0

-rw-r--r-- 1 itadmin 197121 0 Feb 10 11:15 a.txt

-rw-r--r-- 1 itadmin 197121 0 Feb 10 11:14 about.html

-rw-r--r-- 1 itadmin 197121 0 Feb 10 11:15 b.txt

-rw-r--r-- 1 itadmin 197121 0 Feb 10 11:15 c.txt

-rw-r--r-- 1 itadmin 197121 0 Feb 10 11:15 d.txt

-rw-r--r-- 1 itadmin 197121 0 Feb 10 11:15 e.txt

-rw-r--r-- 1 itadmin 197121 0 Feb 10 11:14 testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ ls -a

./ ../ .git/ a.txt about.html b.txt c.txt d.txt e.txt testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git status

On branch master

No commits yet

Untracked files:

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

a.txt

about.html

b.txt

c.txt

d.txt

e.txt

testing.py

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

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git add testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git add a.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git status

On branch master

No commits yet

Changes to be committed:

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

new file: a.txt

new file: testing.py

Untracked files:

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

about.html

b.txt

c.txt

d.txt

e.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git rm --cached a.txt

rm 'a.txt'

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git status

On branch master

No commits yet

Changes to be committed:

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

new file: testing.py

Untracked files:

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

a.txt

about.html

b.txt

c.txt

d.txt

e.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ vi testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ nano testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git add testing.py

warning: LF will be replaced by CRLF in testing.py.

The file will have its original line endings in your working directory

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git status

On branch master

No commits yet

Changes to be committed:

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

new file: testing.py

Untracked files:

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

a.txt

about.html

b.txt

c.txt

d.txt

e.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ nano a.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git add a.txt

warning: LF will be replaced by CRLF in a.txt.

The file will have its original line endings in your working directory

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git status

On branch master

No commits yet

Changes to be committed:

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

new file: a.txt

new file: testing.py

Untracked files:

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

about.html

b.txt

c.txt

d.txt

e.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git commit -m "Testing Git for first time!"

Author identity unknown

\*\*\* Please tell me who you are.

Run

git config --global user.email "you@example.com"

git config --global user.name "Your Name"

to set your account's default identity.

Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'itadmin@DESKTOP-RRNPTP1.(none)')

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

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

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git config --global user.email puneetvallurupalli1@gmail.com

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git config user.name

puneki

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git config user.email

puneetvallurupalli1@gmail.com

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git config --global user.email

puneetvallurupalli1@gmail.com

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git commit -m "First Commit!"

[master (root-commit) 5b014ff] First Commit!

2 files changed, 2 insertions(+)

create mode 100644 a.txt

create mode 100644 testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log

commit 5b014fff2e89428cb78ffbbfe4516fe239c8d63a (HEAD -> master)

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:42:52 2021 +0530

First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ nano testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (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: testing.py

Untracked files:

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

about.html

b.txt

c.txt

d.txt

e.txt

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

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git add testing.py

warning: LF will be replaced by CRLF in testing.py.

The file will have its original line endings in your working directory

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git status

On branch master

Changes to be committed:

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

modified: testing.py

Untracked files:

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

about.html

b.txt

c.txt

d.txt

e.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git commit -m "~~Second Commit~~"

[master 3bd1fe9] ~~Second Commit~~

1 file changed, 5 insertions(+), 1 deletion(-)

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log

commit 3bd1fe9fa8727e588bd3845552054557af1d5f9b (HEAD -> master)

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:51:43 2021 +0530

~~Second Commit~~

commit 5b014fff2e89428cb78ffbbfe4516fe239c8d63a

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:42:52 2021 +0530

First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git add .

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git status

On branch master

Changes to be committed:

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

new file: about.html

new file: b.txt

new file: c.txt

new file: d.txt

new file: e.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git commit -m "~~Third Commit~~"

[master ae84fb2] ~~Third Commit~~

5 files changed, 0 insertions(+), 0 deletions(-)

create mode 100644 about.html

create mode 100644 b.txt

create mode 100644 c.txt

create mode 100644 d.txt

create mode 100644 e.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log

commit ae84fb24972e4e465a32c41ebda6f4bc367b3a85 (HEAD -> master)

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:54:52 2021 +0530

~~Third Commit~~

commit 3bd1fe9fa8727e588bd3845552054557af1d5f9b

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:51:43 2021 +0530

~~Second Commit~~

commit 5b014fff2e89428cb78ffbbfe4516fe239c8d63a

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:42:52 2021 +0530

First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log --oneline

ae84fb2 (HEAD -> master) ~~Third Commit~~

3bd1fe9 ~~Second Commit~~

5b014ff First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ ^C

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git checkout 5b014ff

Note: switching to '5b014ff'.

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 5b014ff First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((5b014ff...))

$ ls

a.txt testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((5b014ff...))

$ cat testing.py

print("Hello")

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((5b014ff...))

$ git checkout 3bd1fe9

Previous HEAD position was 5b014ff First Commit!

HEAD is now at 3bd1fe9 ~~Second Commit~~

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((3bd1fe9...))

$ cat testing.py

print("Code to print even numbers from 1 to 100")

for i in range(100):

print(i)

print("\n")

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((3bd1fe9...))

$ nano a.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((3bd1fe9...))

$ git checkout 5b014ff

Previous HEAD position was 3bd1fe9 ~~Second Commit~~

HEAD is now at 5b014ff First Commit!

M a.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((5b014ff...))

$ cat a.txt

Testing Version Control

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((5b014ff...))

$ git checkout 5b014f^C

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((5b014ff...))

$ git checkout ae84fb2

Previous HEAD position was 5b014ff First Commit!

HEAD is now at ae84fb2 ~~Third Commit~~

M a.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((ae84fb2...))

$ cat a.txt

Testing Version Control

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((ae84fb2...))

$ git log --oneline

ae84fb2 (HEAD, master) ~~Third Commit~~

3bd1fe9 ~~Second Commit~~

5b014ff First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((ae84fb2...))

$ git checkout master

Switched to branch 'master'

M a.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ touch oddnum.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ nano oddnum.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (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: a.txt

Untracked files:

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

oddnum.py

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

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git add oddnum.py

warning: LF will be replaced by CRLF in oddnum.py.

The file will have its original line endings in your working directory

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git status

On branch master

Changes to be committed:

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

new file: oddnum.py

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: a.txt

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git add a.txt

warning: LF will be replaced by CRLF in a.txt.

The file will have its original line endings in your working directory

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git status

On branch master

Changes to be committed:

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

modified: a.txt

new file: oddnum.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git add .

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git commit -m "~~Fourth Commit~~"

[master 885ae5f] ~~Fourth Commit~~

2 files changed, 6 insertions(+), 1 deletion(-)

create mode 100644 oddnum.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log

commit 885ae5f2139d8ba601a5289fbf8a0689daf47c81 (HEAD -> master)

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 12:20:11 2021 +0530

~~Fourth Commit~~

commit ae84fb24972e4e465a32c41ebda6f4bc367b3a85

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:54:52 2021 +0530

~~Third Commit~~

commit 3bd1fe9fa8727e588bd3845552054557af1d5f9b

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:51:43 2021 +0530

~~Second Commit~~

commit 5b014fff2e89428cb78ffbbfe4516fe239c8d63a

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:42:52 2021 +0530

First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git status

On branch master

nothing to commit, working tree clean

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git commit --amend -m "~Fourth Commit~"

[master 8bc888e] ~Fourth Commit~

Date: Wed Feb 10 12:20:11 2021 +0530

2 files changed, 6 insertions(+), 1 deletion(-)

create mode 100644 oddnum.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log

commit 8bc888ef0612c2eee001a20a8e2c751447f9180f (HEAD -> master)

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 12:20:11 2021 +0530

~Fourth Commit~

commit ae84fb24972e4e465a32c41ebda6f4bc367b3a85

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:54:52 2021 +0530

~~Third Commit~~

commit 3bd1fe9fa8727e588bd3845552054557af1d5f9b

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:51:43 2021 +0530

~~Second Commit~~

commit 5b014fff2e89428cb78ffbbfe4516fe239c8d63a

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:42:52 2021 +0530

First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log --online

fatal: unrecognized argument: --online

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log --oneline

8bc888e (HEAD -> master) ~Fourth Commit~

ae84fb2 ~~Third Commit~~

3bd1fe9 ~~Second Commit~~

5b014ff First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git checkout 5b014ff

Note: switching to '5b014ff'.

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 5b014ff First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((5b014ff...))

$ git commit --amend -m "~First Commit~"

[detached HEAD 7db8b74] ~First Commit~

Date: Wed Feb 10 11:42:52 2021 +0530

2 files changed, 2 insertions(+)

create mode 100644 a.txt

create mode 100644 testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((7db8b74...))

$ git log --oneline

7db8b74 (HEAD) ~First Commit~

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((7db8b74...))

$ git checkout master

Warning: you are leaving 1 commit behind, not connected to

any of your branches:

7db8b74 ~First Commit~

If you want to keep it by creating a new branch, this may be a good time

to do so with:

git branch <new-branch-name> 7db8b74

Switched to branch 'master'

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git logi

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

The most similar command is

log

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log --oneline

8bc888e (HEAD -> master) ~Fourth Commit~

ae84fb2 ~~Third Commit~~

3bd1fe9 ~~Second Commit~~

5b014ff First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git checkout 5b014ff

Note: switching to '5b014ff'.

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 5b014ff First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((5b014ff...))

$ git log --oneline

5b014ff (HEAD) First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((5b014ff...))

$ git commit --amend -m "~First Commit~"

[detached HEAD 53e9d3f] ~First Commit~

Date: Wed Feb 10 11:42:52 2021 +0530

2 files changed, 2 insertions(+)

create mode 100644 a.txt

create mode 100644 testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((53e9d3f...))

$ git log --oneline

53e9d3f (HEAD) ~First Commit~

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((53e9d3f...))

$ git log

commit 53e9d3f442cfdb07f79055dca519145325c56d40 (HEAD)

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:42:52 2021 +0530

~First Commit~

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka ((53e9d3f...))

$ git checkout master

Warning: you are leaving 1 commit behind, not connected to

any of your branches:

53e9d3f ~First Commit~

If you want to keep it by creating a new branch, this may be a good time

to do so with:

git branch <new-branch-name> 53e9d3f

Switched to branch 'master'

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log

commit 8bc888ef0612c2eee001a20a8e2c751447f9180f (HEAD -> master)

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 12:20:11 2021 +0530

~Fourth Commit~

commit ae84fb24972e4e465a32c41ebda6f4bc367b3a85

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:54:52 2021 +0530

~~Third Commit~~

commit 3bd1fe9fa8727e588bd3845552054557af1d5f9b

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:51:43 2021 +0530

~~Second Commit~~

commit 5b014fff2e89428cb78ffbbfe4516fe239c8d63a

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:42:52 2021 +0530

First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log --oneline

8bc888e (HEAD -> master) ~Fourth Commit~

ae84fb2 ~~Third Commit~~

3bd1fe9 ~~Second Commit~~

5b014ff First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git reset --hard HEAD^

HEAD is now at ae84fb2 ~~Third Commit~~

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git liogi

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

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log

commit ae84fb24972e4e465a32c41ebda6f4bc367b3a85 (HEAD -> master)

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:54:52 2021 +0530

~~Third Commit~~

commit 3bd1fe9fa8727e588bd3845552054557af1d5f9b

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:51:43 2021 +0530

~~Second Commit~~

commit 5b014fff2e89428cb78ffbbfe4516fe239c8d63a

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:42:52 2021 +0530

First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log --oneline

ae84fb2 (HEAD -> master) ~~Third Commit~~

3bd1fe9 ~~Second Commit~~

5b014ff First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git reset --hard HEAD~2

HEAD is now at 5b014ff First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log

commit 5b014fff2e89428cb78ffbbfe4516fe239c8d63a (HEAD -> master)

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:42:52 2021 +0530

First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ nano test2.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git status

On branch master

Untracked files:

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

test2.py

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

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git add test2.py

warning: LF will be replaced by CRLF in test2.py.

The file will have its original line endings in your working directory

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git commit -m "~~Second Commit~~"

[master 03eb718] ~~Second Commit~~

1 file changed, 2 insertions(+)

create mode 100644 test2.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log --oneline

03eb718 (HEAD -> master) ~~Second Commit~~

5b014ff First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ nano test3.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git status

On branch master

Untracked files:

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

test3.py

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

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git add test3.py

warning: LF will be replaced by CRLF in test3.py.

The file will have its original line endings in your working directory

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git commit -m "~Third Commit~"

[master c90525d] ~Third Commit~

1 file changed, 1 insertion(+)

create mode 100644 test3.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log --oneline

c90525d (HEAD -> master) ~Third Commit~

03eb718 ~~Second Commit~~

5b014ff First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git rebase --onto branchname~commitnumbertodelete branchname~commitnumbertokeep master

fatal: invalid upstream 'branchname~commitnumbertokeep'

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git rebase --onto branchname~commitnumbertodelete branchname~commitnumbertokeep master

fatal: invalid upstream 'branchname~commitnumbertokeep'

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git rebase --onto master~1 master~3 master

fatal: invalid upstream 'master~3'

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git rebase --onto master~3 master~2 master

fatal: Does not point to a valid commit 'master~3'

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git rebase --onto master~2 master~0 master

Successfully rebased and updated refs/heads/master.

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log --oneline

5b014ff (HEAD -> master) First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$

itadmin@DESKTOP-RRNPTP1 MINGW64 ~

$ cd ProjectEureka

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git log

commit 5b014fff2e89428cb78ffbbfe4516fe239c8d63a (HEAD -> master)

Author: puneki <puneetvallurupalli1@gmail.com>

Date: Wed Feb 10 11:42:52 2021 +0530

First Commit!

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git config user.name

puneki

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git remote

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git remote add origin https://github.com/puneki/TestingRepositories.git

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git remote -v

origin https://github.com/puneki/TestingRepositories.git (fetch)

origin https://github.com/puneki/TestingRepositories.git (push)

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git remote

origin

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git push origin master

]

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git remote

origin

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ ]

bash: ]: command not found

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git push origin master

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% (4/4), 284 bytes | 284.00 KiB/s, done.

Total 4 (delta 0), reused 0 (delta 0), pack-reused 0

To https://github.com/puneki/TestingRepositories.git

\* [new branch] master -> master

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ ls

a.txt testing.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ git pull origin master

remote: Enumerating objects: 4, done.

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

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

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

Unpacking objects: 100% (3/3), 828 bytes | 63.00 KiB/s, done.

From https://github.com/puneki/TestingRepositories

\* branch master -> FETCH\_HEAD

5b014ff..67b6aac master -> origin/master

Updating 5b014ff..67b6aac

Fast-forward

primetest.py | 12 ++++++++++++

1 file changed, 12 insertions(+)

create mode 100644 primetest.py

itadmin@DESKTOP-RRNPTP1 MINGW64 ~/ProjectEureka (master)

$ ls

a.txt primetest.py testing.py