Name   : Harish Aeyya

Email   : harishaeyya@gmail.com

Mobile : 8184891562

git reset:

It's used to undo the changes in your working directory and get back to a specific commit while discarding all the commits made after that one.

- Use this to return the entire working tree to the last committed state.

- This will discard commits.

- Changes which commit a branch HEAD is currently pointing at.

- It alters the existing commit histroy.

- Can be used to unstage a file.

git reset options:

1. **git reset --soft:**

   - remove data from local repo

   - git reset --soft<commit id>

2. **git reset --mixed:**

   - remove data from localrepo and staging area

   - the mixed option is the default if we don't provide git reset with an option

   - git reset --mixed<commit id>

3. **git reset --hard:**

   - remove date from local repo,staging area and working directory also

   - git reset --hard<commit id>

Practical:

open git bash and create a directory

cd is a command-line shell command used to change the current working directory

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop (master)
$ mkdir harish

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop (master)
$ cd harish

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git init
Initialized empty Git repository in C:/Users/hp/Desktop/haris
h/.git/
```

Git init:git init convert normal directory to git directory.

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ vi file1

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git add .
warning: in the working copy of 'file1', LF will be replaced
by CRLF the next time Git touches it

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git commit -m "1st commit"
[master (root-commit) adbd1fe] 1st commit
 1 file changed, 7 insertions(+)
 create mode 100644 file1

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ vi file1

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git add .
warning: in the working copy of 'file1', LF will be replaced
by CRLF the next time Git touches it

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git commit -m "2nd commit"
[master 49d3144] 2nd commit
 1 file changed, 4 insertions(+)
```

create a file1 and changes the commits

- 1st commit(7 insertions)

- 2nd commit(4 insertions)

```
iwertyjhgf
jhgfds
qwertyu
12345
098765
zxcvmnbvc
rtyuoiu
```

```
111111111
222222222
333333333
444444444
```

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ vi file1

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git add .
warning: in the working copy of 'file1', LF will be replaced
by CRLF the next time Git touches it

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git commit -m "3rd commit"
[master cdb508d] 3rd commit
 1 file changed, 5 insertions(+)

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ vi file1

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git add .
warning: in the working copy of 'file1', LF will be replaced
by CRLF the next time Git touches it

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git commit -m "4th commit"
[master 26c12ab] 4th commit
 1 file changed, 3 insertions(+), 1 deletion(-)
```

And also changes the 3rd and 4th commit in file1

- 3rd commit(5 insertions)
- 4th commit(3 insertions)

```
aaaaaaaaa
bbbbbbbbb
ccccccccc
ddddddddd
```

```
AAAAAAAAA
BBBBBBBBB
CCCCCCCCC
```

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ cat file1
iwertyjhgf
jhgfds
qwertyu
12345
098765
zxcvmnbvc
rtyuoiu
111111111
222222222
333333333
444444444
aaaaaaaaa
bbbbbbbbb
ccccccccc
ddddddddd
AAAAAAAAA
BBBBBBBBB
CCCCCCCCC
```

The git log command displays who committed.

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git log
commit 26c12abf8b5b25106b1d48a9a43186d0831ddfbb (HEAD -> master)
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 11:11:46 2024 +0530

    4th commit

commit cdb508db9e128ce3079d2a5f05dc4bd8dfcef670
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 11:10:22 2024 +0530

    3rd commit

commit 49d3144298f067828e96d12bcbf0027802a7c761
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 11:09:23 2024 +0530

    2nd commit

commit adbd1fe86a67fa4697fb41295a55e469e85cdcfd
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 11:07:39 2024 +0530

    1st commit
```

you don't want to commit 4th, copy the 3rd commit id

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git reset --soft cdb508db9e1

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:    file1


hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git reset -- mixed cdb508db9e1

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:    file1


hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git reset --hard cdb508db9e1
HEAD is now at cdb508d 3rd commit

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git status
On branch master
nothing to commit, working tree clean
```

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git log
commit cdb508db9e128ce3079d2a5f05dc4bd8dfcef670 (HEAD -> master)
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 11:10:22 2024 +0530

    3rd commit

commit 49d3144298f067828e96d12bcbf0027802a7c761
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 11:09:23 2024 +0530

    2nd commit

commit adbd1fe86a67fa4697fb41295a55e469e85cdcfd
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 11:07:39 2024 +0530

    1st commit
```

you can also observe the file1

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ cat file1
iwertyjhgf
jhgfds
qwertyu
12345
098765
zxcvmnbvc
rtyuoiu
111111111
222222222
333333333
444444444
aaaaaaaaa
bbbbbbbbb
ccccccccc
ddddddddd
```

you don't want to commit 3rd and 2nd commits,copy the 1st commit id

if you commit changes in a step use the command

git reset --hard<commit id>

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git reset --hard adbd1fe86a6
HEAD is now at adbd1fe 1st commit

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ cat file1
iwertyjhgf
jhgfds
qwertyu
12345
098765
zxcvmnbvc
rtyuoiu
```

git revert:

It's used for undoing changes to a repository's commit history

create a file2 and changes the commits 1st and 2nd

- 1st commit(3insertions)

```
11111111
22222222
33333333
```

- 2nd commit(3insertions)

```
aaaaaaaa
bbbbbbbb
cccccccc
```

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ vi file2

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git add .
warning: in the working copy of 'file2', LF will be replaced by CRLF the next time Git touches it

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git commit -m "my 1st commit"
[master c578b5b] my 1st commit
 1 file changed, 3 insertions(+)
 create mode 100644 file2

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ vi file2

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git add .
warning: in the working copy of 'file2', LF will be replaced by CRLF the next time Git touches it

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git commit -m "my 2nd commit"
[master 47e3173] my 2nd commit
 1 file changed, 3 insertions(+)
```

- 3rd commit(3insertions)

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ vi file2

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git add .
warning: in the working copy of 'file2', LF will be replaced by CRLF the next time Git touches it

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git commit -m "3rd commit"
[master 563adf3] 3rd commit
 1 file changed, 3 insertions(+)

hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ cat file2
11111111
22222222
33333333
aaaaaaaa
bbbbbbbb
cccccccc
AAAAAAAA
BBBBBBBB
CCCCCCCC
```

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git log
commit 563adf33bde8878ec6738e6d292f9fb54c752169 (HEAD -> master)
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 13:33:56 2024 +0530

    3rd commit

commit 47e317328623d09c8307ac113c2a5add135727a2
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 13:26:10 2024 +0530

    my 2nd commit

commit c578b5b0f6aaf85467dde10c0ebfeeb9c54d2659
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 13:25:20 2024 +0530

    my 1st commit

commit adbd1fe86a67fa4697fb41295a55e469e85cdcfd
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 11:07:39 2024 +0530

    1st commit
```

you don't want want commit 3rd commit copy the that commit id

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git revert HEAD 563adf33b
[master 9f00aa1] Revert "3rd commit"
 1 file changed, 3 deletions(-)
```

```
Revert "3rd commit"

This reverts commit 563adf33bde8878ec6738e6d292f9fb54c752169.

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Revert currently in progress.
#
# Changes to be committed:
#       modified:   file2
#
~
~
~
```

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ git log
commit 9f00aa1995986ae0aacc92cc1aa764d0d4c4840f (HEAD -> master)
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 13:47:08 2024 +0530

    Revert "3rd commit"

    This reverts commit 563adf33bde8878ec6738e6d292f9fb54c752169.

commit 563adf33bde8878ec6738e6d292f9fb54c752169
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 13:33:56 2024 +0530

    3rd commit

commit 47e317328623d09c8307ac113c2a5add135727a2
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 13:26:10 2024 +0530

    my 2nd commit

commit c578b5b0f6aaf85467dde10c0ebfeeb9c54d2659
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 13:25:20 2024 +0530

    my 1st commit

commit adbd1fe86a67fa4697fb41295a55e469e85cdcfd
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 12 11:07:39 2024 +0530

    1st commit
```

```
hp@DESKTOP-IOD7VPG MINGW64 ~/Desktop/harish (master)
$ cat file2
11111111
22222222
33333333
aaaaaaaa
bbbbbbbb
cccccccc

hp@DESKTOP-IOD7VPG MINGW64  /Desktop/harish (master)
```

git rebase:

Git rebase is a powerful feature in Git used for integrating changes from one branch onto another. It works by taking a set of commits, "replaying" them on top of another branch, effectively altering the commit history.

When you rebase a branch onto another, Git identifies the common ancestor commit between the two branches. Then, it "replays" the commits from the branch being rebased on top of the current branch. This results in a linear history without the merge commits that are typically created by git merge.

Basic workflow for using git rebase:

1. Checkout the branch you want to rebase onto: First, you switch to the branch you want to integrate changes into.

2. Start the rebase: You initiate the rebase process with the command git rebase <branch_to_rebase_from>. This command tells Git to take the commits from the specified branch and apply them on top of the current branch.

3. Resolve conflicts (if any): During the rebase process, Git might encounter conflicts if the same lines of code were modified in both branches. You'll need to resolve these conflicts manually.

4. Complete the rebase: After resolving any conflicts, you continue the rebase process by staging the resolved files with git add and then executing git rebase --continue. If you encounter any conflicts during this process, you'll need to resolve them and continue until the rebase is complete.

5. Optional: Push the rebased branch: If you rebased a branch that has already been pushed to a remote repository, you might need to force push the changes using git push --force since the commit history has been rewritten.

```
hp@DESKTOP-IOD7VPG MINGW64 ~/desktop
$ mkdir harish

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop
$ cd harish

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish
$ git init
Initialized empty Git repository in C:/Users/hp/Desktop/harish/.git/

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (master)
$ ls -la
total 8
drwxr-xr-x 1 hp 197121 0 Apr 19 13:55 ./
drwxr-xr-x 1 hp 197121 0 Apr 19 13:55 ../
drwxr-xr-x 1 hp 197121 0 Apr 19 13:55 .git/
```

```
hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (master)
$ vi about.txt

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (master)
$ vi home.txt

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (master)
$ git add .
warning: in the working copy of 'about.txt', LF will be replaced by CRLF the next time Git touches
it
warning: in the working copy of 'home.txt', LF will be replaced by CRLF the next time Git touches i
t

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (master)
$ git commit -m "intial commit"
[master (root-commit) f008aa9] intial commit
 2 files changed, 2 insertions(+)
 create mode 100644 about.txt
 create mode 100644 home.txt

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (master)
$ git log
commit f008aa98bb29a15022bd920266a238b4abe2a036 (HEAD -> master)
Author: harishaeyya <harishaeyya.gmail.com>
Date:   Fri Apr 19 13:56:47 2024 +0530

    intial commit
```

```
hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (master)
$ git checkout -b dev
Switched to a new branch 'dev'

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (dev)
$ ls
about.txt   home.txt

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (dev)
$ vi about.txt

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (dev)
$ git add .
warning: in the working copy of 'about.txt', LF will be replaced by CRLF the next time Git touches
it

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (dev)
$ git commit -m "add the line about"
[dev f2ea2ba] add the line about
 1 file changed, 1 insertion(+)

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (dev)
$ git log --oneline --all --graph
* f2ea2ba (HEAD -> dev) add the line about
* f008aa9 (master) intial commit
```

```
hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (dev)
$ git checkout master
Switched to branch 'master'

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (master)
$ vi home.txt

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (master)
$ git add .
warning: in the working copy of 'home.txt', LF will be replaced by CRLF the next time Git touches i
t

hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (master)
$ git commit -m "extra line in home"
[master d0c9a37] extra line in home
 1 file changed, 1 insertion(+)
```

```
hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (master)
$ git log --oneline --all --graph
* d0c9a37 (HEAD -> master) extra line in home
| * f2ea2ba (dev) add the line about
|/
* f008aa9 intial commit
```

```
hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (master)
$ git rebase dev
Successfully rebased and updated refs/heads/master.
```

git rebase command is merging the file,but it's don't create a extra commit and the commit history is straight line.

```
hp@DESKTOP-IOD7VPG MINGW64 ~/desktop/harish (master)
$ git log --oneline --all --graph
* 1b035bb (HEAD -> master) extra line in home
* f2ea2ba (dev) add the line about
* f008aa9 intial commit
```