

GIT CHEAT SHEET

\$ git status

\$ git branch

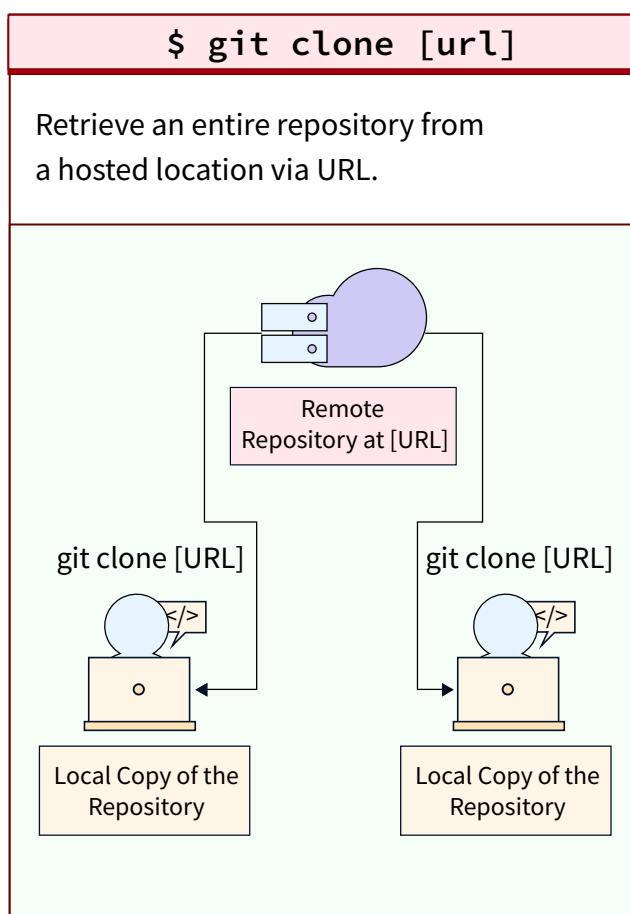
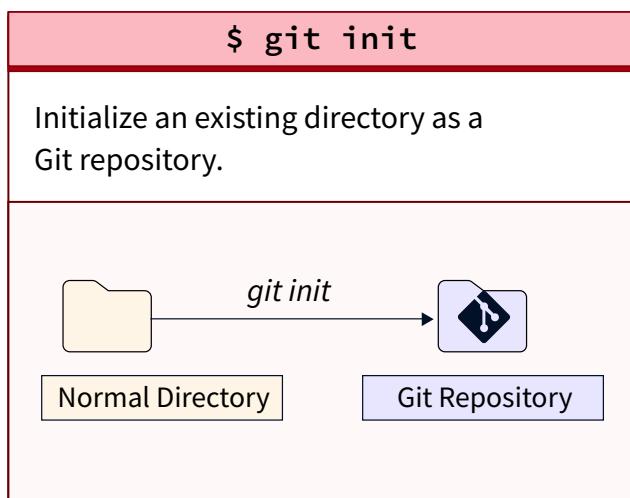
\$ git log

\$ git fetch

SCALER
Topics

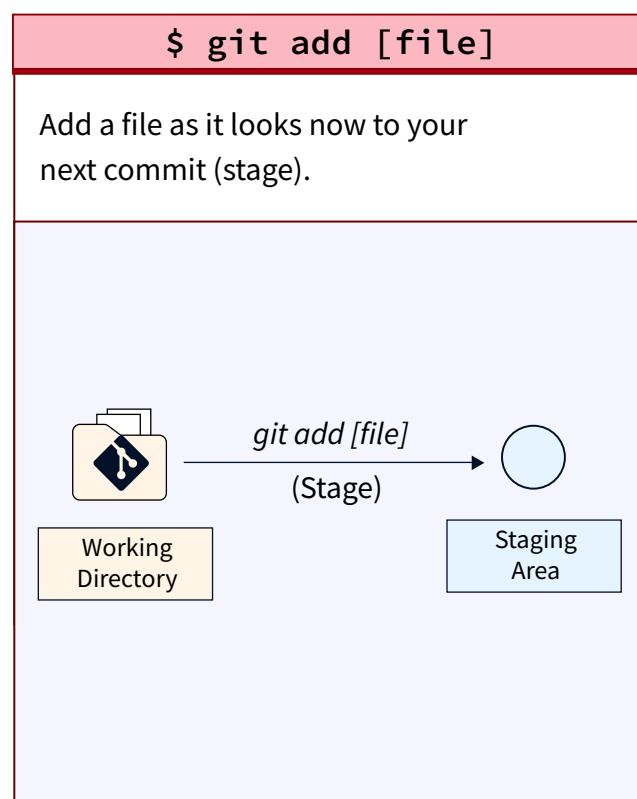
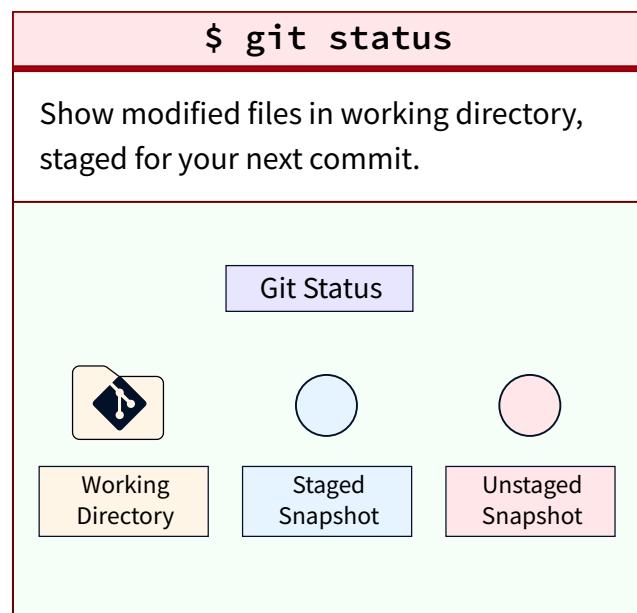
SETUP & INIT

Configuring user information, initializing and cloning repositories.



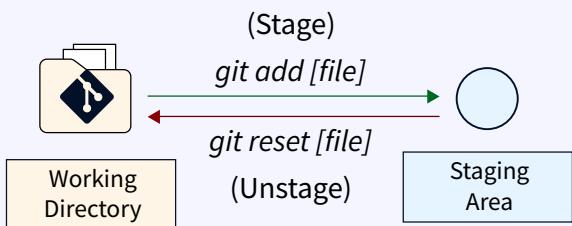
STAGE & SNAPSHOT

Working with snapshots and the Git staging area.



\$ git reset [file]

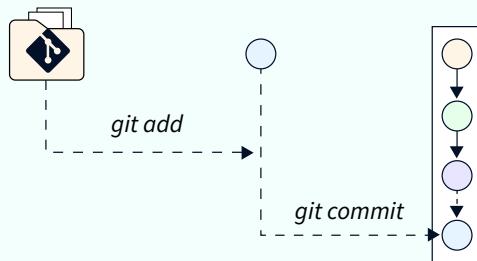
Unstage a file while retaining the changes in working directory.



\$ git commit -m “[descriptive message]”

Commit your staged content as a new commit snapshot.

Working Directory Staging Area Git Repository (Commit History)



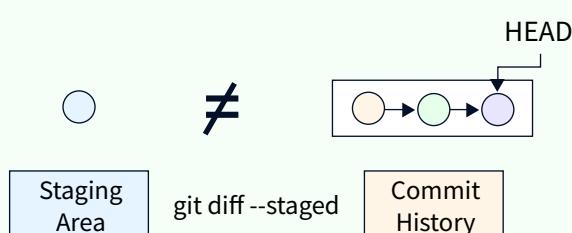
\$ git diff

Diff of what is changed but not staged



git diff --staged

Diff of what is staged but not yet committed.

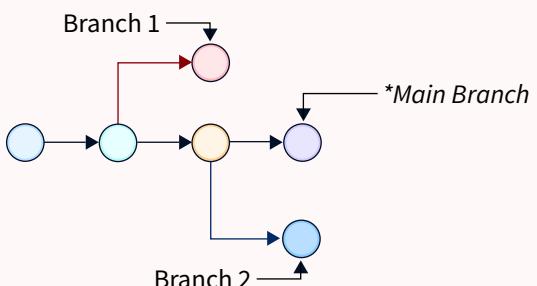


BRANCH & MERGE

Isolating work in branches, changing context, and integrating changes.

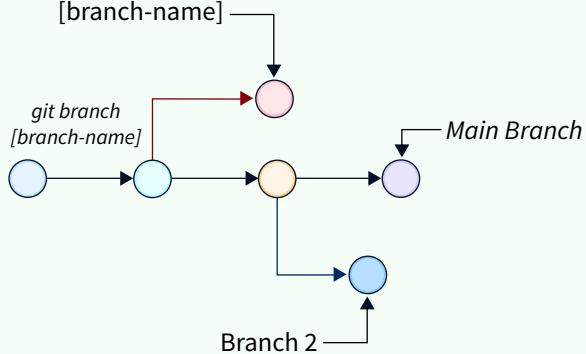
\$ git branch

List your branches. A * will appear next to the currently active branch.



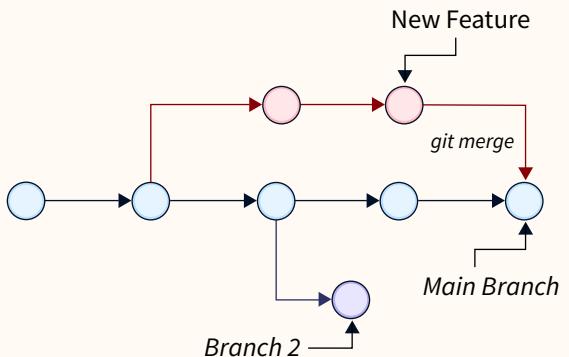
\$ git branch [branch-name]

Create a new branch at the current commit.



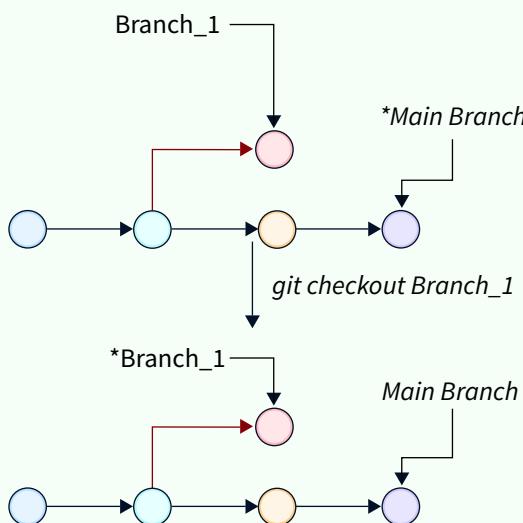
\$ git merge [branch]

Merge the specified branch's history into the current one.



\$ git checkout

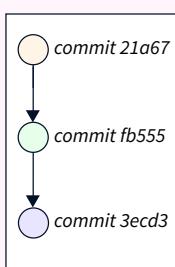
Switch to another branch and check it out into your working directory.



\$ git log

Show all commits in the current branch's history.

Git Repository



Git Logs

commit 21a67	Author : xyz	Date: Mon May 16 16:03:16 2022	Commit Message
commit fb555	Author : ABC	Date: Tue May 17 09:05:45 2022	Commit Message
commit 3ecd3	Author : XYZ	Date: Sun May 22 19:45:34 2022	Commit Message

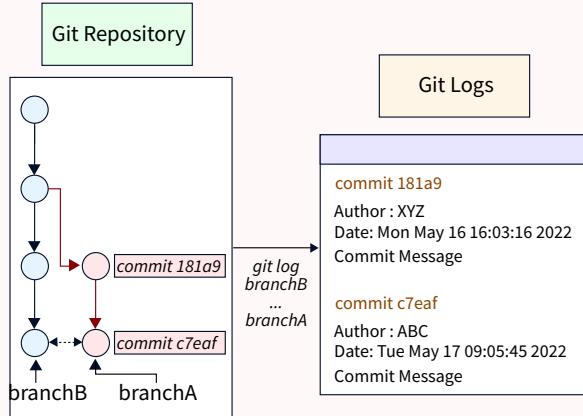


INSPECT & COMPARE

Configuring user information,
initializing and cloning repositories.

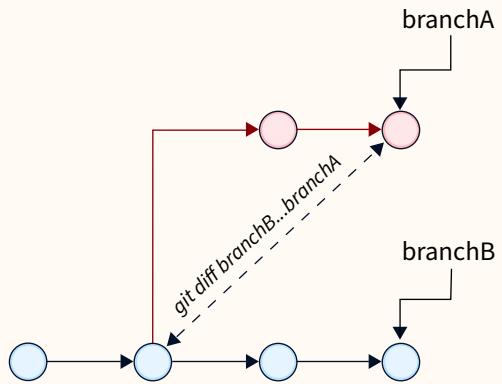
\$ git log branchB..branchA

Show the commits on branchA that
are not on branchB.



\$ git diff branchB...branchA

Show the diff of what is in branchA
that is not in branchB.

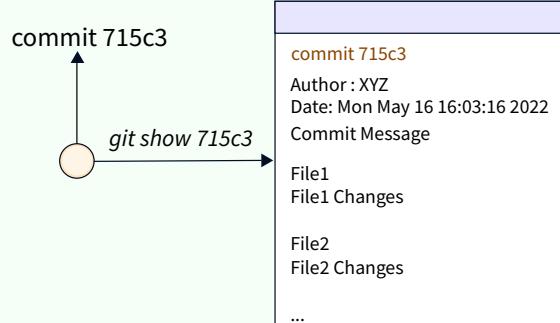


\$ git log --follow [file]

Show the commits that changed file,
even across renames.

\$ git show [SHA]

Show any object in Git in
human-readable format.

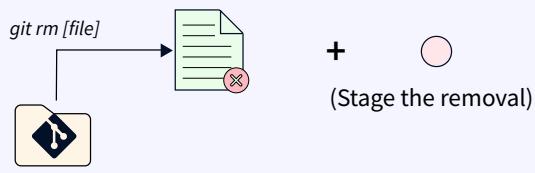


TRACKING PATH CHANGES

Versioning file removes and path changes.

```
$ git rm [file]
```

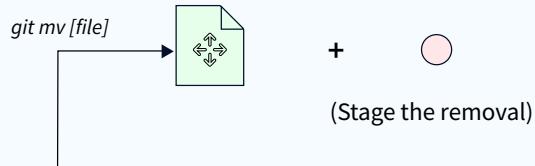
Delete the file from the project and stage the removal for commit.



Git Repository

```
$ git mv [existing-path] [new-path]
```

Change an existing file path and stage the move.



Git Repository

```
$ git log --stat -M
```

Show all commit logs with indication of any paths that moved.

IGNORING PATTERNS

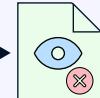
Preventing unintentional staging or committing of files.

logs/
*.notes
pattern*/

Save a file with desired patterns as .gitignore with either direct string matches or wildcard globs.

.gitignore

logs/
*.notes
pattern*/



```
$ git config --global core.excludesfile [file]
```

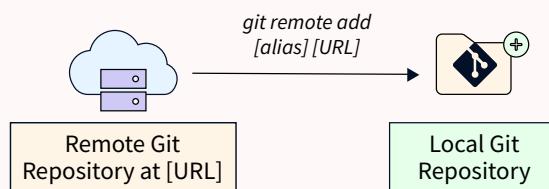
System wide ignore pattern for all local repositories

SHARE & UPDATE

Retrieving updates from another repository and updating local repos.

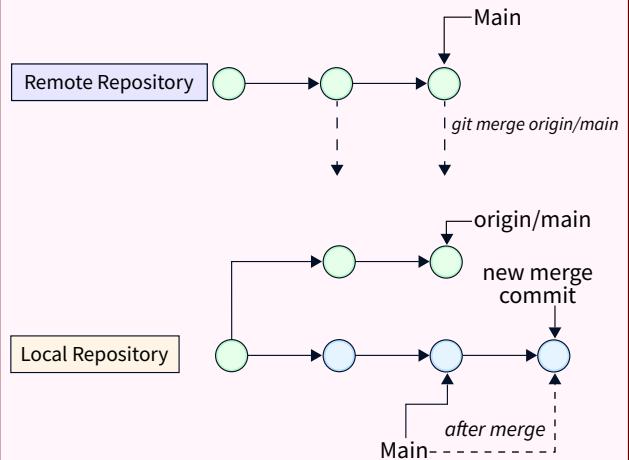
\$ git remote add [alias] [url]

Add a git URL as an alias.



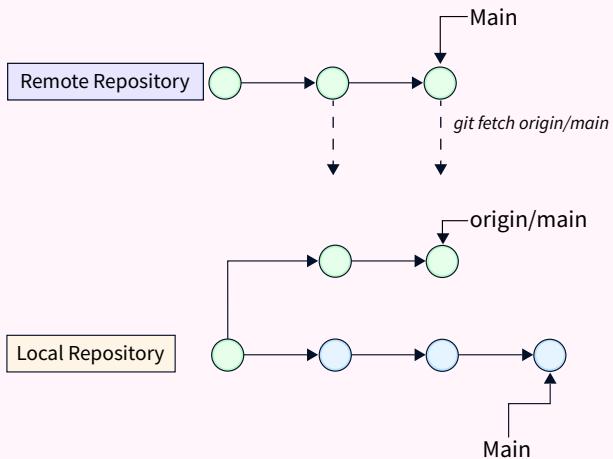
\$ git merge [alias]/[branch]

Merge a remote branch into your current branch to bring it up to date.



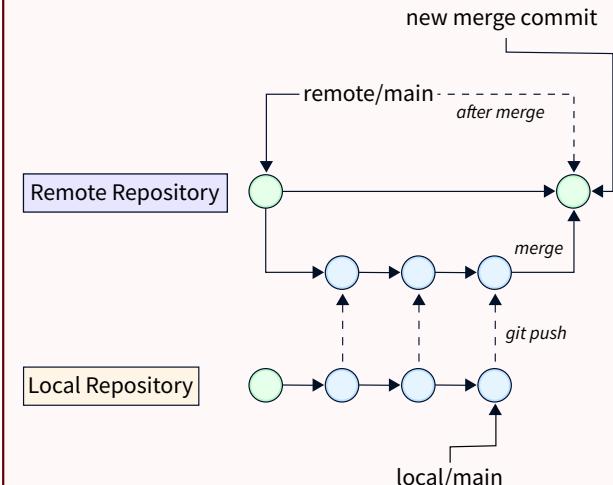
\$ git fetch [alias]

Fetch down all the branches from that Git remote.



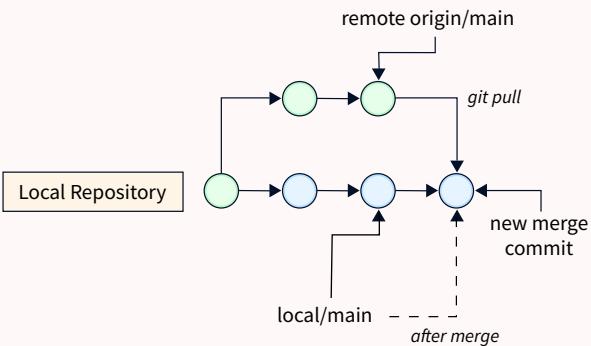
\$ git push [alias] [branch]

Transmit local branch commits to the remote repository branch.



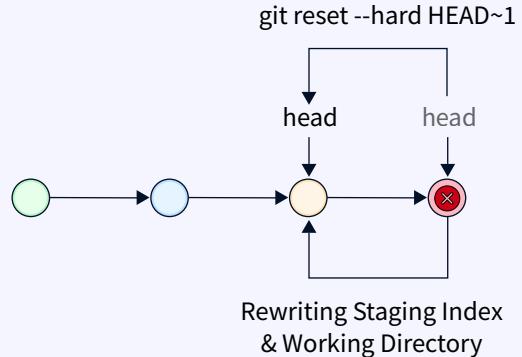
\$ git pull

Fetch and merge any commits from the tracking remote branch.



\$ git reset --hard [commit]

Clear staging area, rewrite working tree from specified commit.

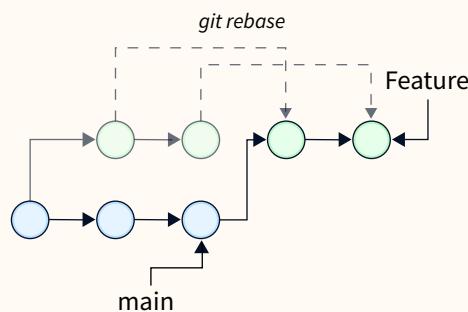


REWRITE HISTORY

Rewriting branches, updating commits and clearing history.

\$ git rebase [branch]

Apply any commits of the current branch ahead of specified one.

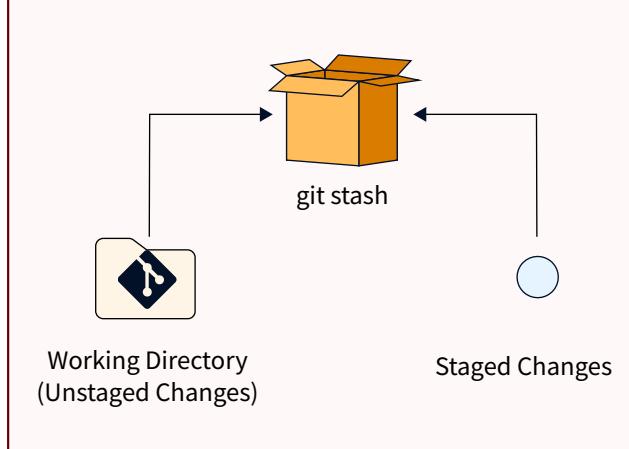


TEMPORARY COMMITS

Temporarily store modified, tracked files in order to change branches.

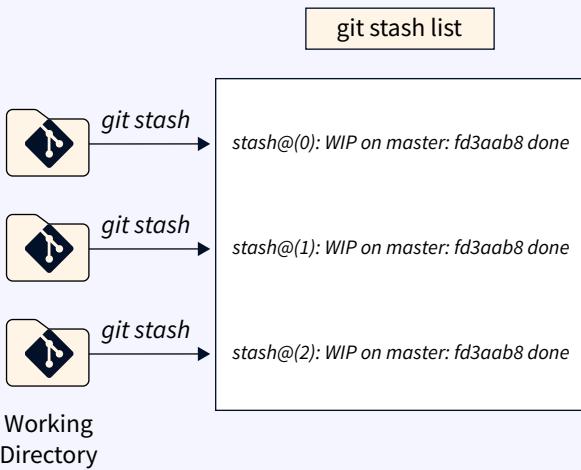
\$ git stash

Save modified and staged changes.



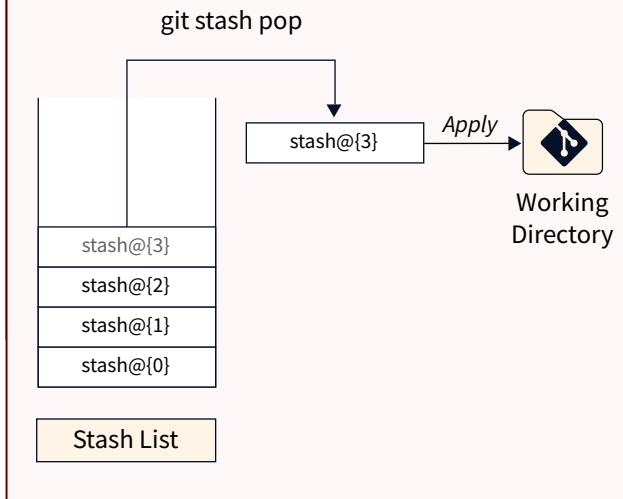
\$ git stash list

List stack-order of stashed file changes.



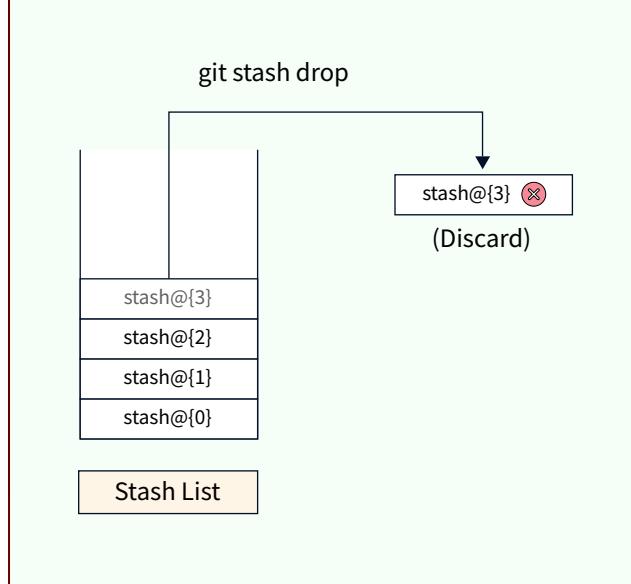
\$ git stash pop

Write working from the top of the stash stack.



\$ git stash drop

Discard the changes from the top of the stash stack.



SCALER TOPICS

Unlock your potential in software development with
FREE COURSES from SCALER TOPICS!

Register now and take the first step towards your future Success!



PRATEEK NARANG

C++ for Beginners

5.9k enrolled

₹ Free



TARUN LUTHRA

Java for Beginners

₹ Free

6.8k enrolled

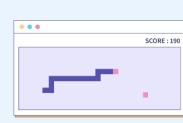
That's not it. Explore 20+ Courses by clicking below

Explore Other Courses

Practice **CHALLENGES**
and become 1% better everyday



CIFAR-10 Image
Classification Using PyTorch
Article
No. Of Questions : 3
[Go to Challenge >](#)



How to Build a Snake Game
in JavaScript?
Article
No. Of Questions : 3
[Go to Challenge >](#)

Explore Other Challenges