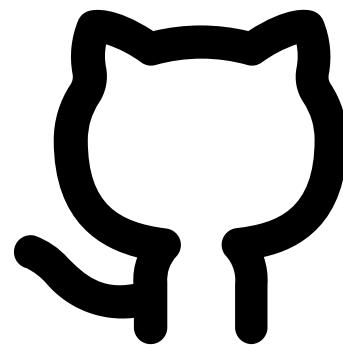
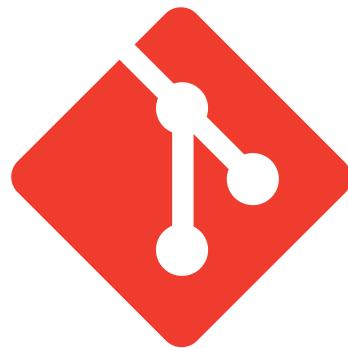


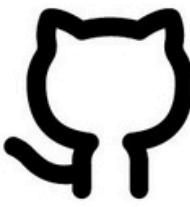
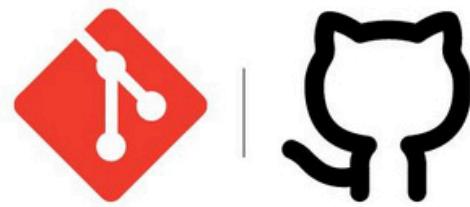
Advance Guide

Git
and
GitHub



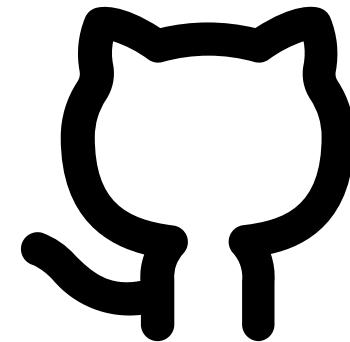
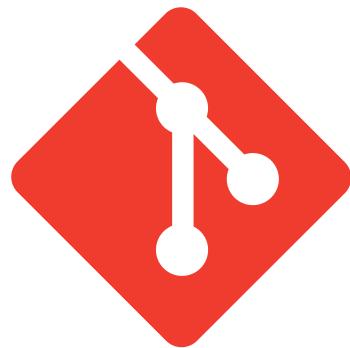


Before going through this document, please check out my Beginner's Guide



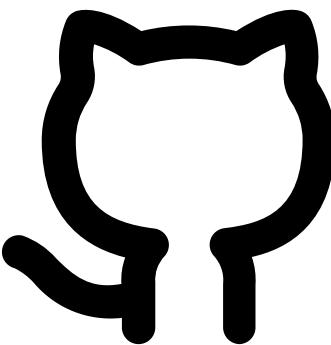
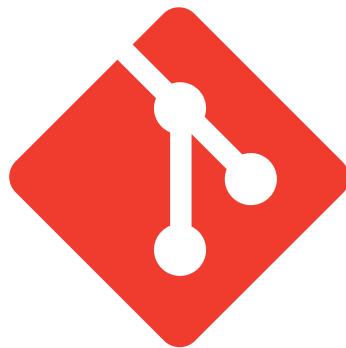
Beginner's Guide

Git and GitHub



What is git config

The git config command is a **convenience function** that is used to set Git configuration values on a global or local project level. These configuration levels correspond to **.gitconfig text files**. Executing git config will modify a configuration text file.



git config commands

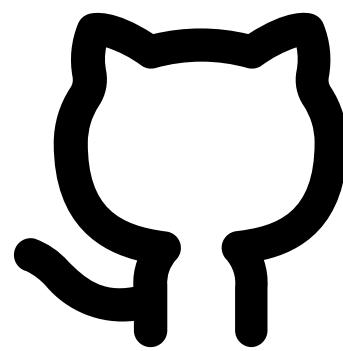
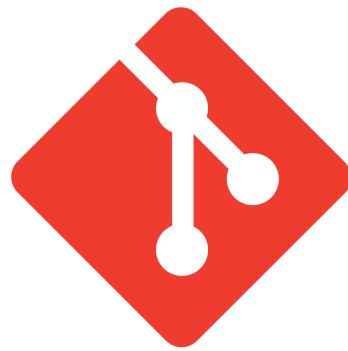


Bash

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

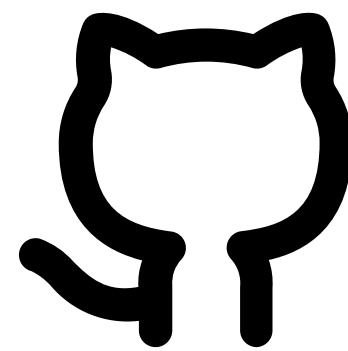
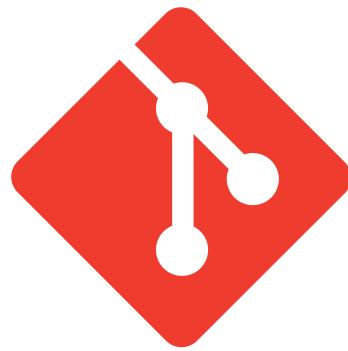
```
git config --global user.email  
"your.email@example.com"
```

```
git config --list
```



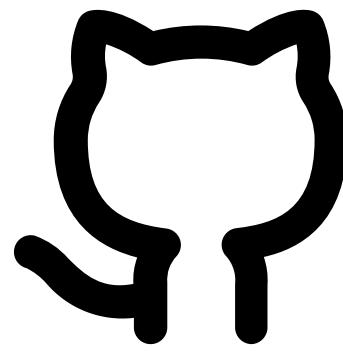
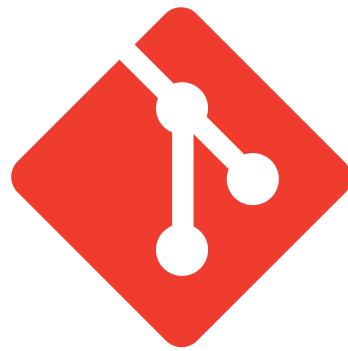
What is git status

The git status command displays the **state** of the working directory and the **staging area**. It lets you see which changes have been staged, which haven't, and which files aren't being tracked by Git



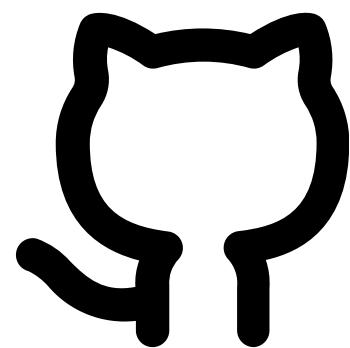
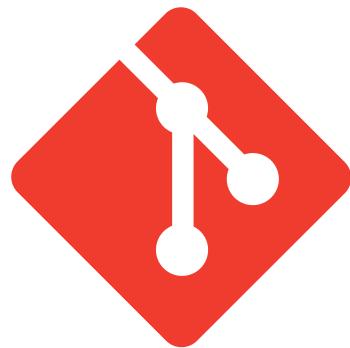
What is Staging Area

Git staging area is an **intermediate space** where you can stage changes before committing them, allowing you to selectively include modifications. Use ``git add <file>`` to stage changes and ``git commit -m "message"`` to commit them.

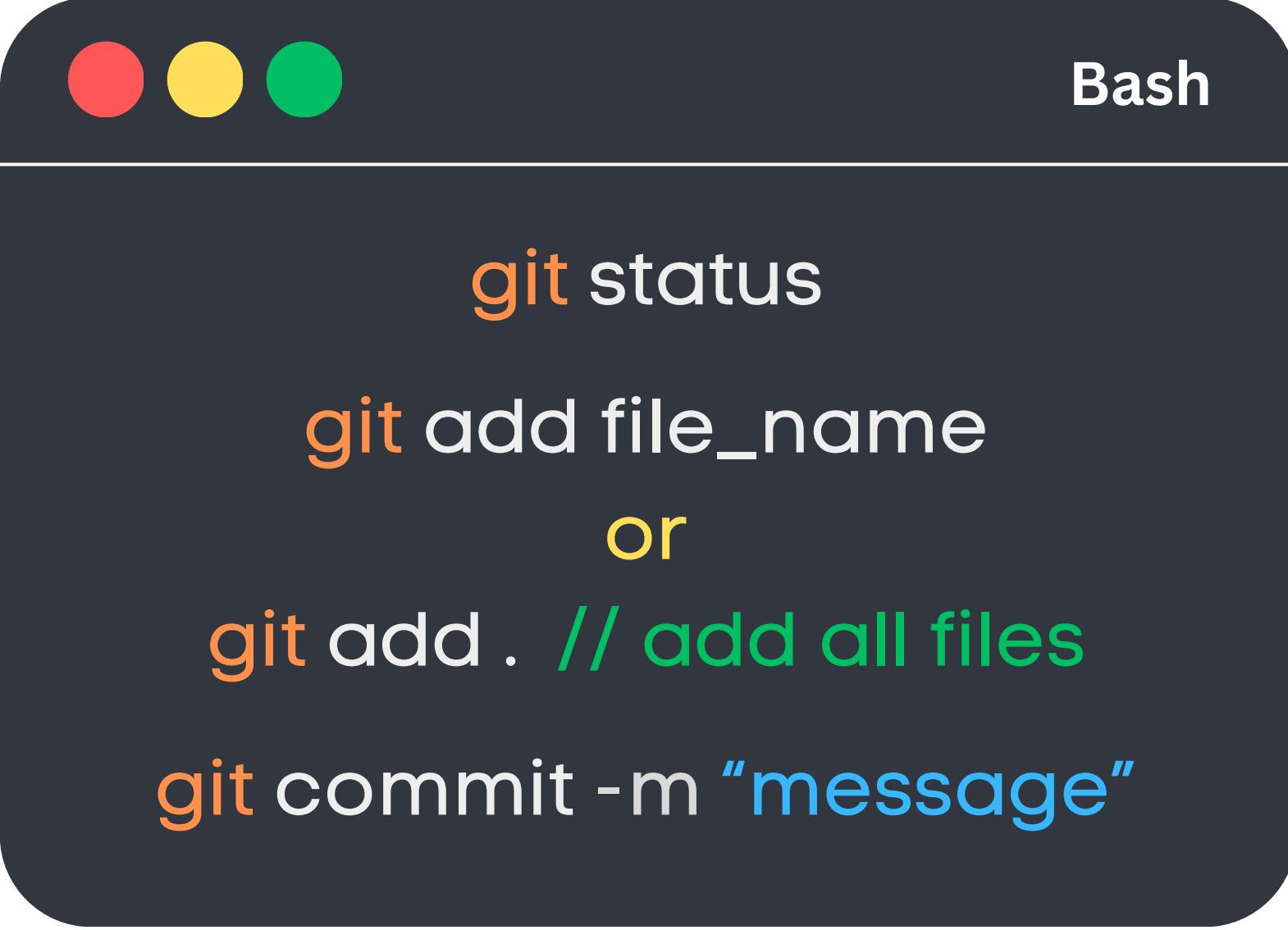


What is git commit

The git commit command captures a **snapshot** of the project's currently staged changes. Committed snapshots can be thought of as “safe” versions of a project. Git will **never change** them unless you explicitly **ask it to**.

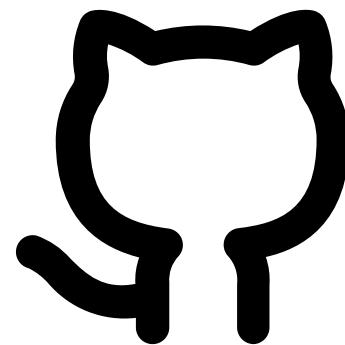
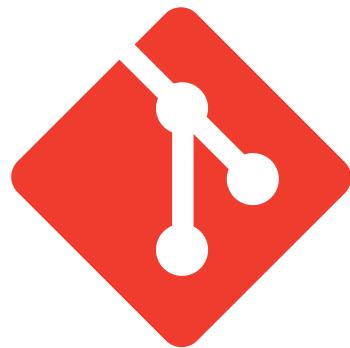


Staging commands



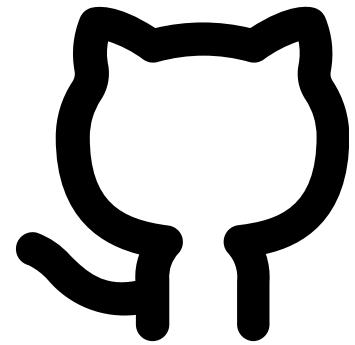
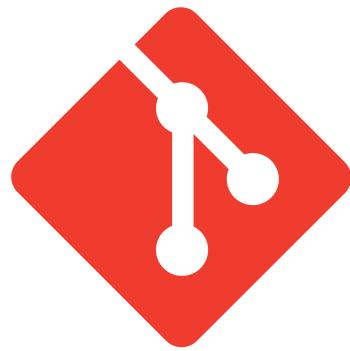
Bash

```
git status  
git add file_name  
or  
git add . // add all files  
git commit -m "message"
```

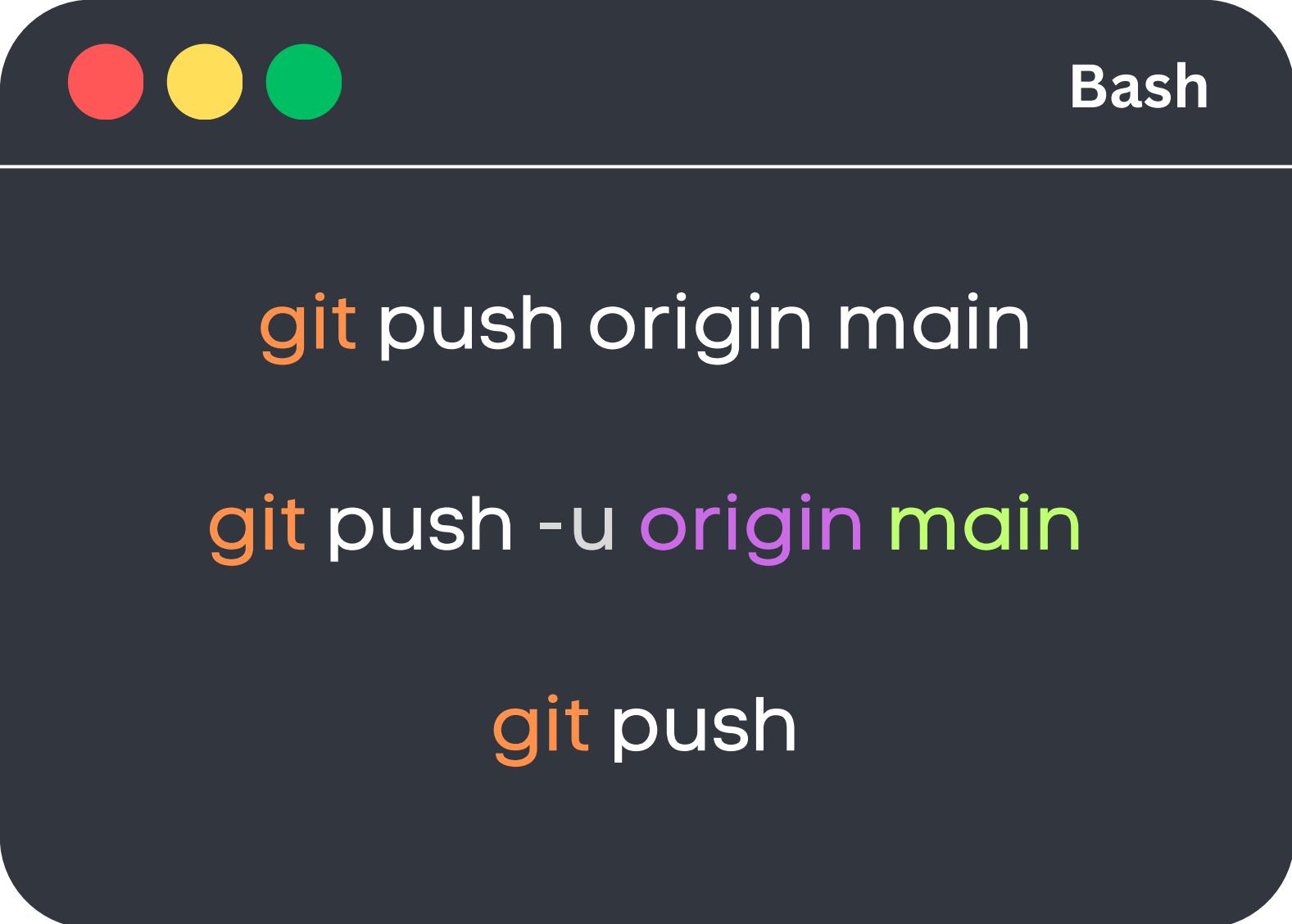


What is git push

The `git push` command is used to upload local repository content to a **remote repository**. Pushing is how you transfer commits from your local repository to a **remote repo in GitHub**. Pushing has the potential to overwrite changes, **caution** should be taken when pushing.



Push commands

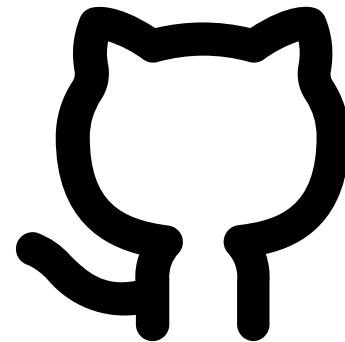
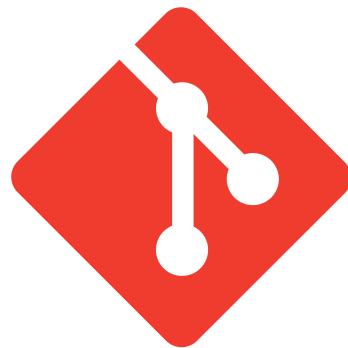


git push origin main

Bash

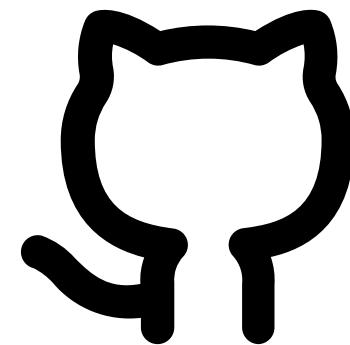
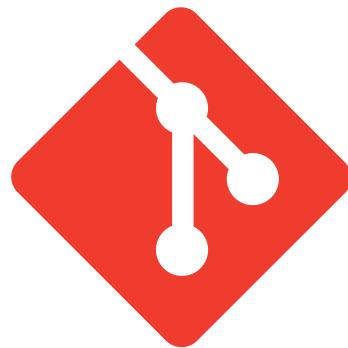
git push -u origin main

git push



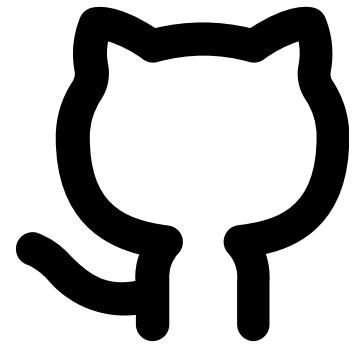
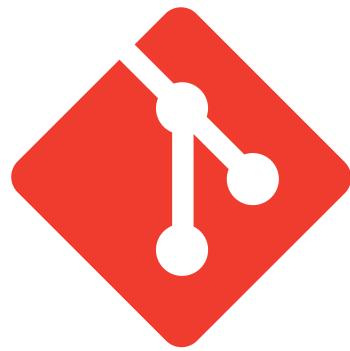
Commands Explain

1. Pushes your local main branch to the **remote repository origin**.
2. Pushes the main branch and sets origin/main as the default upstream branch **for future**.
3. Pushes changes to the remote repository for the current branch set with an upstream **tracking branch**.

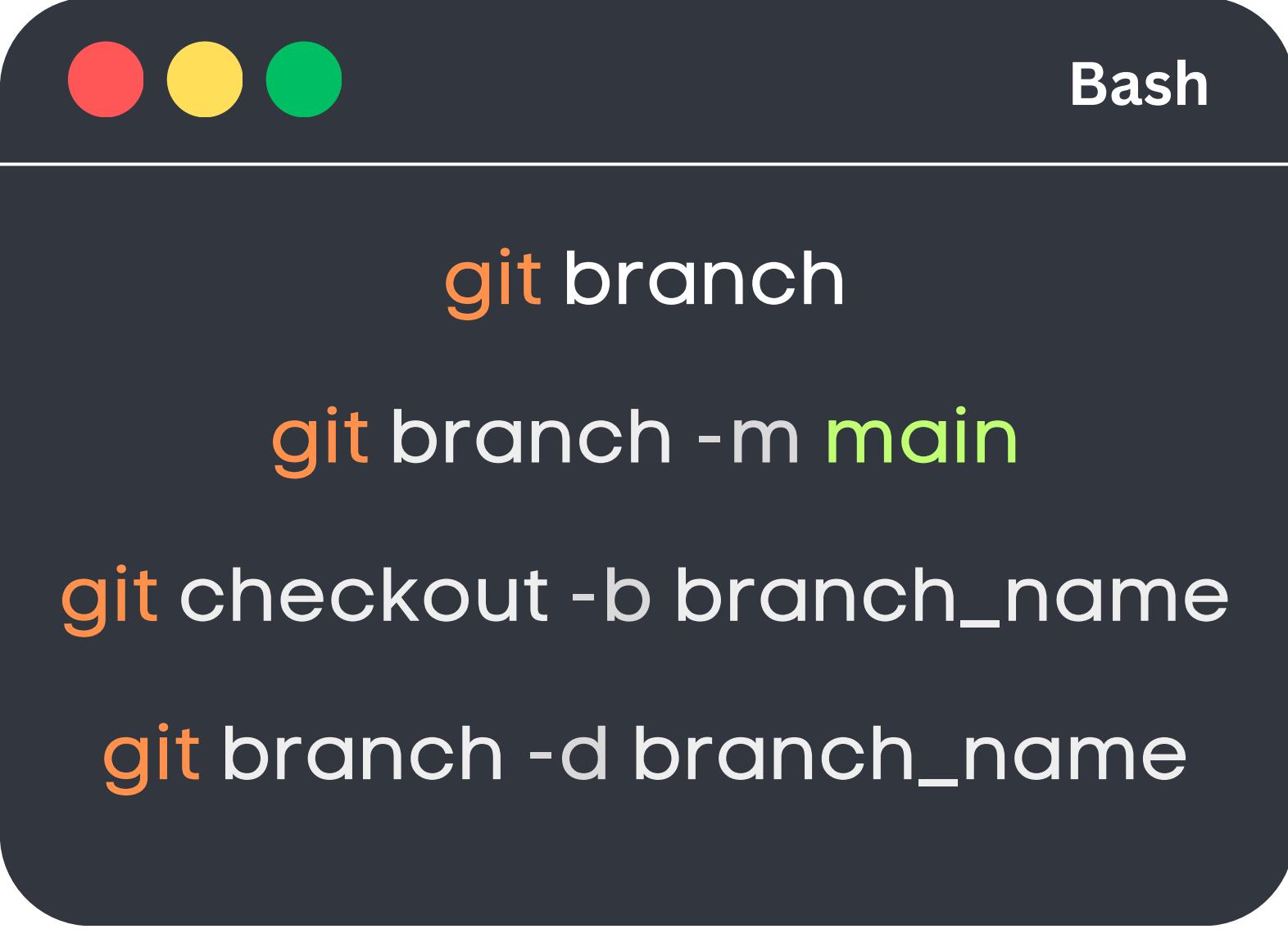


What is Branch

Git branches are **pointers** to a **snapshot** of your changes. When you want to add a new feature or fix a bug you **spawn** a new branch. This makes it harder for unstable code to get merged into the main code base, and it gives you the chance to clean up your future's history before **merging** it into the main branch.

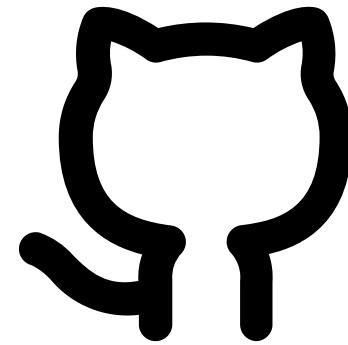
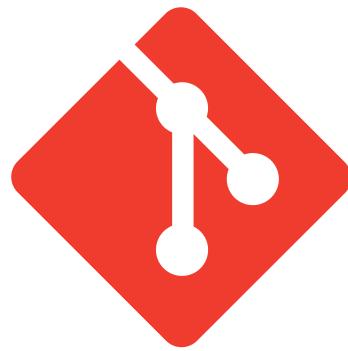


Branch commands



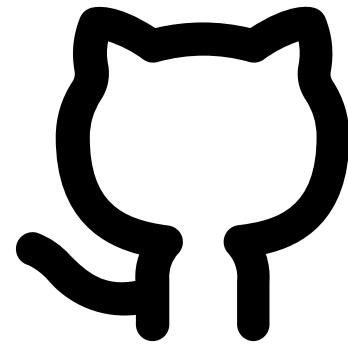
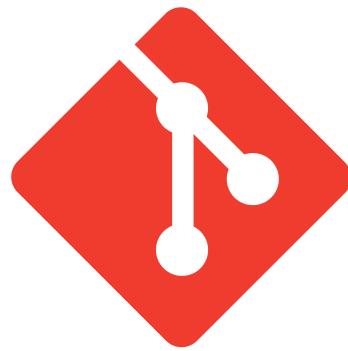
Bash

```
git branch  
git branch -m main  
git checkout -b branch_name  
git branch -d branch_name
```



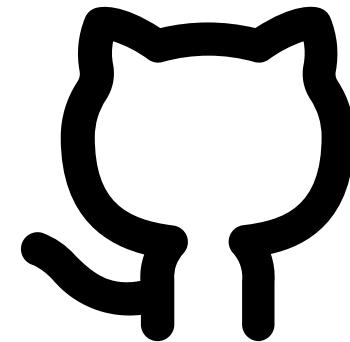
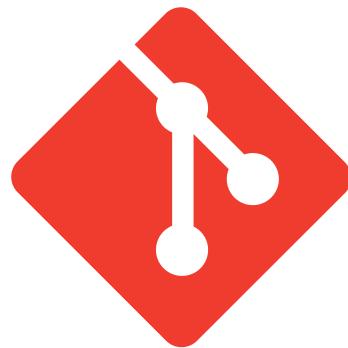
Commands Explain

1. **Lists** all local branches.
2. **Rename** the current branch to main.
3. Creates and **switches** to a new branch named "branch_name".
4. **Delete** the branch named "branch_name".



What is Merge

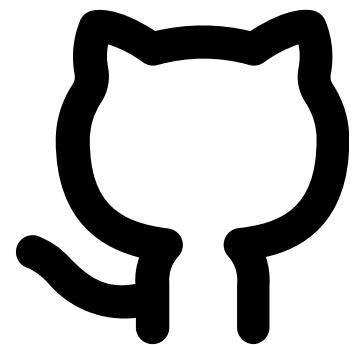
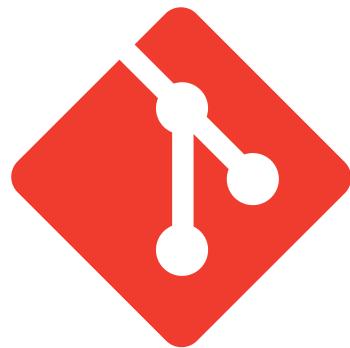
Merging is Git's way of putting a **forked** history back together again. The ``git merge`` command lets you take the independent lines of development created by ``git branch`` and integrate them into a single branch.



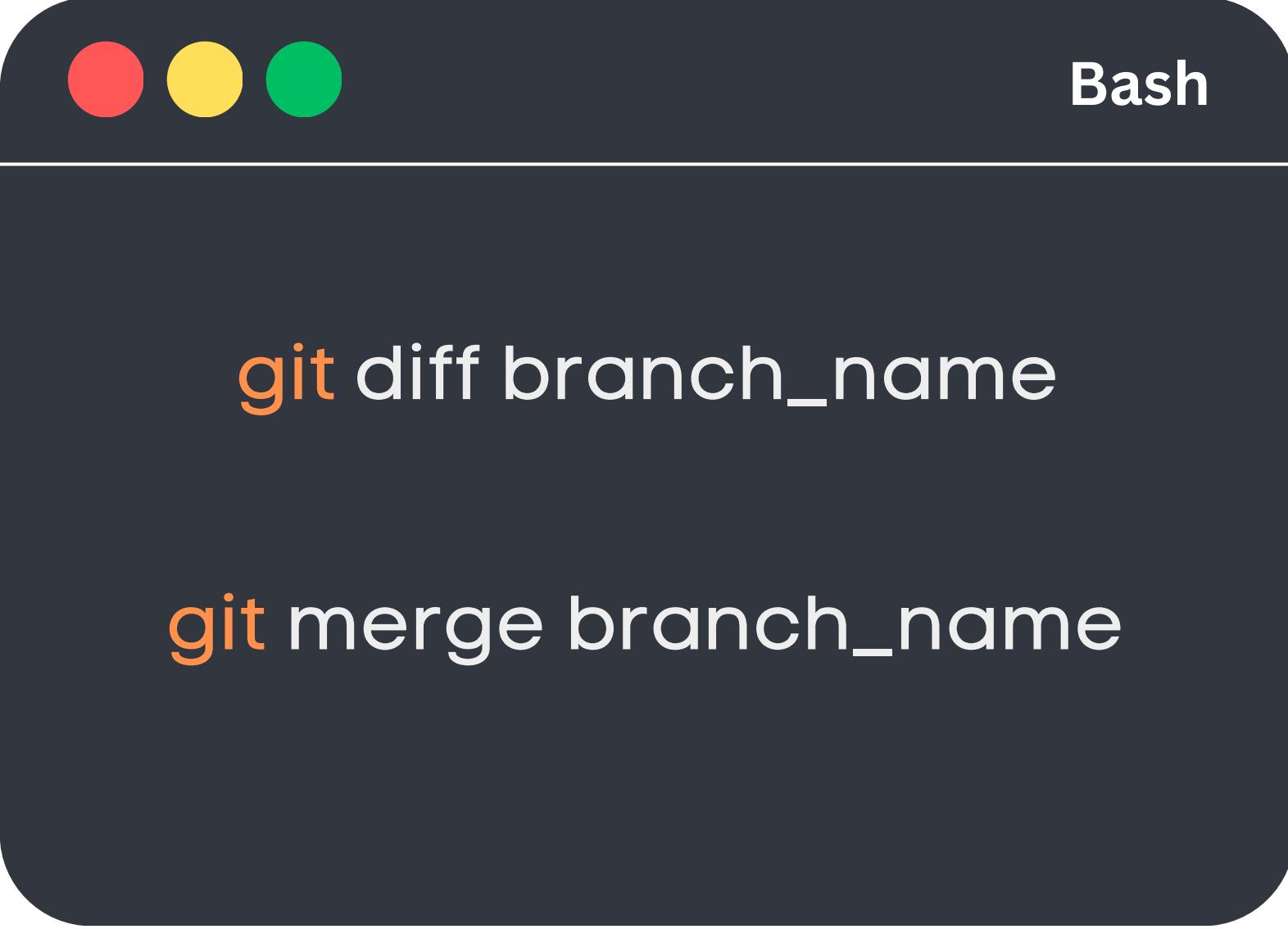
What is Fork

In **git** it is simply a **copy** of an existing repository in which the new owner disconnects the codebase from previous committers.

In **GitHub** it is a new repository that shares code and visibility settings with the original **upstream** repository.

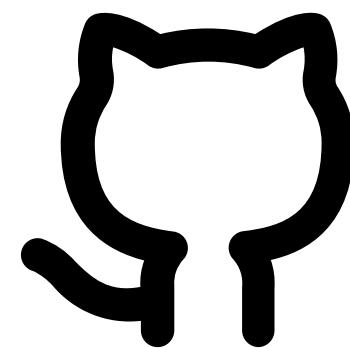
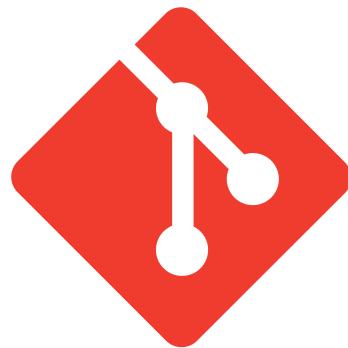


Merge commands



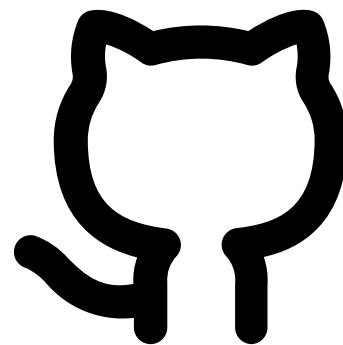
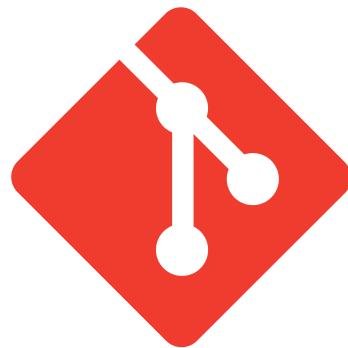
Bash

```
git diff branch_name  
git merge branch_name
```



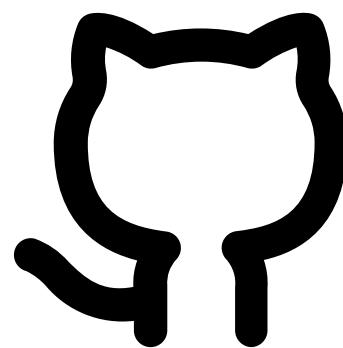
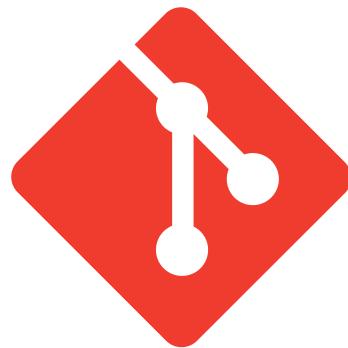
Commands Explain

1. Shows the **differences** between the current branch and "branch_name".
2. **Integrates changes** from "branch_name" into the current branch.



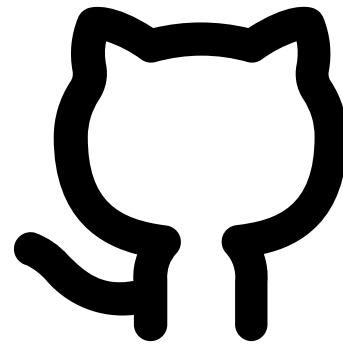
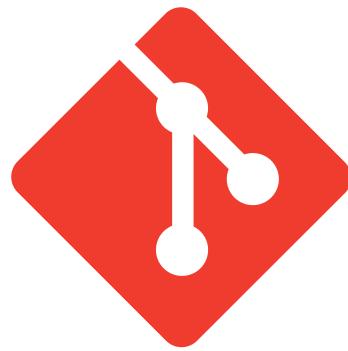
Merge Conflict

Merge conflicts happen when people make **different changes** to the **same line of the same file**, or when one person edits a file and another person deletes the same file. You must resolve all merge conflicts before you can merge a **pull request** on GitHub.



What is Pull Request

A pull request is a **proposal** to merge a set of changes from one branch into another. In a pull request, collaborators can **review and discuss** the proposed set of changes before they integrate the changes into the main **repository**.

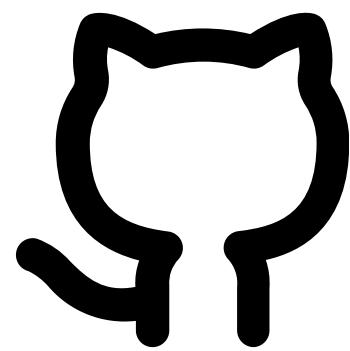
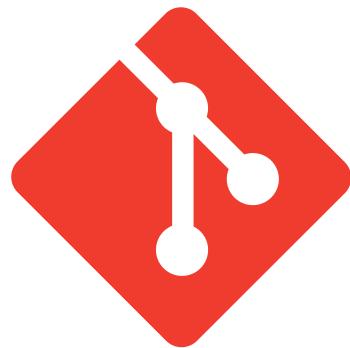


What is git reset

The git reset command is a complex and versatile tool for **undoing changes**. It has three primary forms

--soft --mixed --hard

The three arguments each correspond to three internal state management mechanism's, The Commit Tree (**HEAD**), The **Staging Index**, and The **Working Directory**.

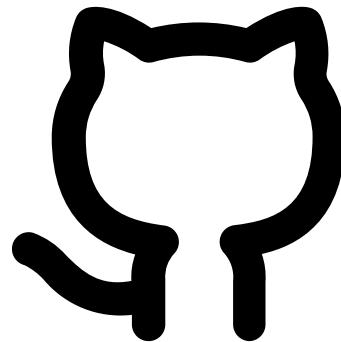
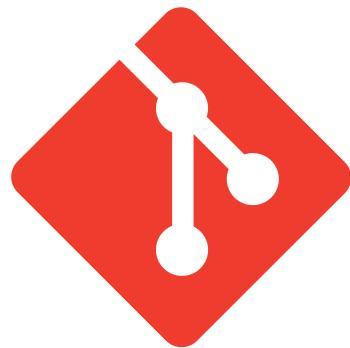


git reset commands



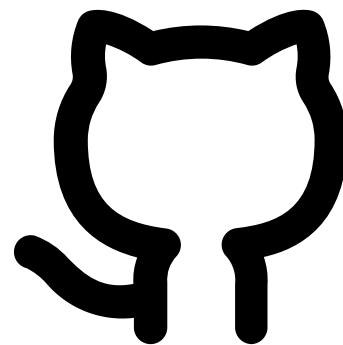
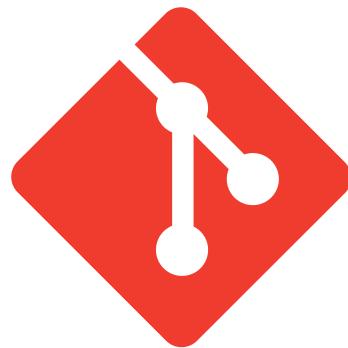
Bash

```
git reset  
git reset file_name  
git reset HEAD~n //n= 1,2,3...  
git reset commit_hash
```



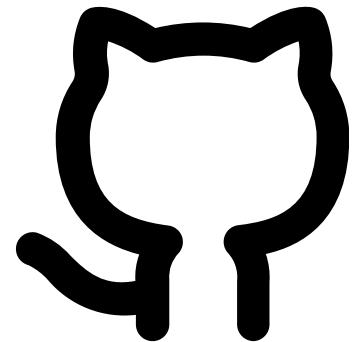
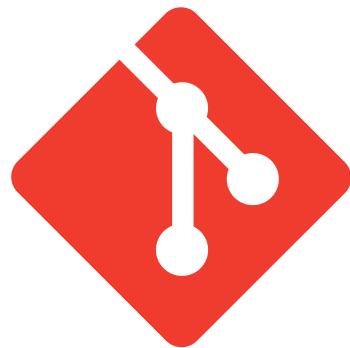
Commands Explain

1. Resets **all files**
2. Resets **specific file**
3. Moves the HEAD **n commits** backward, preserving changes.
4. Resets the HEAD to **commit_hash** and reset that specific commit. To find the commit hash use **`git log`** command.



What is git log

The `git log` command displays committed snapshots, that snapshot covers **commit hash** and various information. It lets you list the project history, filter it, and **search** for specific changes.

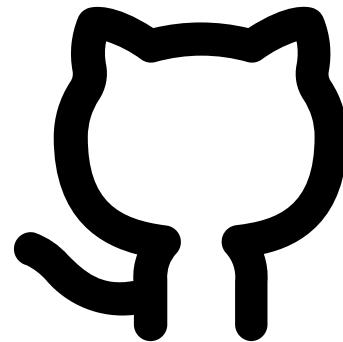
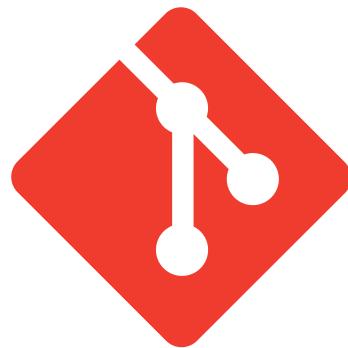


git reset commands



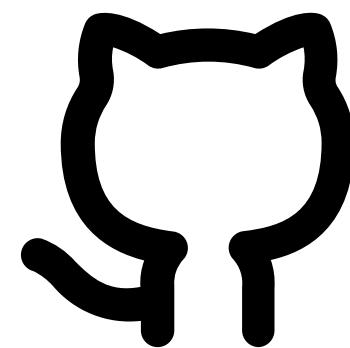
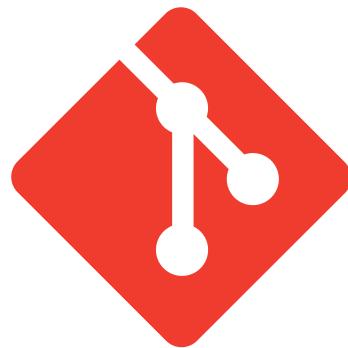
Bash

```
git reset --soft  
git reset --mixed  
git reset --hard
```



Commands Explain

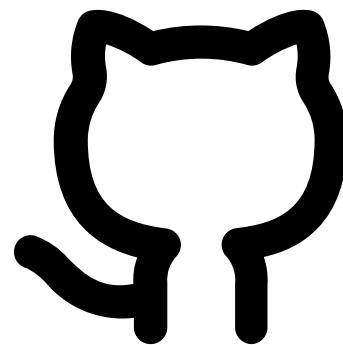
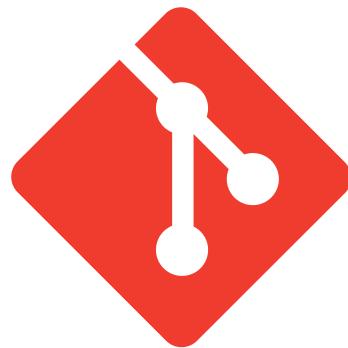
1. Moves HEAD to a prior commit,
keeping staged changes for
recommitting.
2. Resets HEAD to a prior commit,
unstaging changes for review
before recommitting.
3. Resets HEAD to a prior commit,
discarding all changes since
then irreversibly.



What is git pull

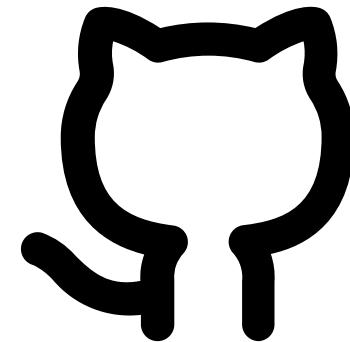
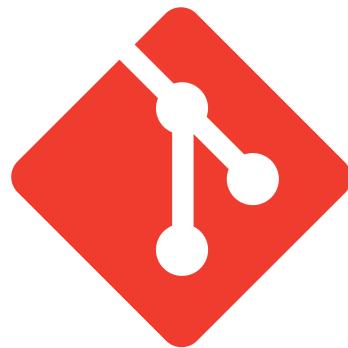
The `'git pull repo_name'` command is used to fetch and download content from a **remote repository** and immediately update the local repository to match that content.

Merging remote upstream changes into your local repository is a common task in Git-based collaboration work flows.



What is git clone

`git clone repo_name` is a Git command line utility which is used to target an existing repository and create a clone, or copy of the target repository.



“A guide can take your knowledge from beginner to advanced, but it's practice that makes you a master.”

THANK
YOU

A large, bold, black, hand-drawn style text that reads "THANK YOU". The letters are slightly irregular and have a textured appearance.

made by Parnab Bagchi