KDnuggets | Visit KDnuggets.com for more cheatsheets and additional learning resources.

Git CheatSheet

Git is open-source software for distributed version control. It is used for tracking files, team collaboration, software development, and disaster management.

Basic

Clone the repo is located in the local directory <repo>. You can also clone a remote repository located in a remote server using a URL via HTTP or SSH.

$ git clone <repo>

Create an empty repository in a specific directory <dir>.

$ git init <dir>

Define the author name for the current repository.

$ git config user.name <name>

Check out the staged, unstaged, and untracked files.

$ git status

Display all of the commit histories.

$ git log

Show the unstaged changes between the index and the working directory.

$ git diff

Local Changes

Add the directory to the staging area. You can add changes from single or multiple files or even use "." to add all the files to the staging area.

$ git add <directory>

Commit the changes in the staging area with a message "<message>" describing the changes.

$ git commit -m "<message>"

Branching

Lists all the branches in the current repository.

$ git branch

Create a new branch <new-branch>.

$ git branch <new-branch>

Create and checkout the new branch <new-branch>.

$ git checkout -b <new-branch>

Merge an <branch> into a current branch.

$ git merge <branch>

Delete the local branch <branch>.

$ git branch -d <branch>

Update and Publish

Show the list of all locally configure remotes

$ git remote -v

Add new remote with name <name> and repository address <URL>

$ git remote add <name> <URL>

Download the changes from a remote to a specific branch <remote> <branch> without integrating it with HEAD.

$ git fetch <remote> <branch>

Download the changes from the remote-specific branch and merge them into HEAD.

$ git pull <remote> <branch>

Push the local changes to the remote branch.

$ git push <remote> <branch>

Delete the branch on the remote repository.

$ git branch -dr <branch>

Undo

Create a new commit that undoes all of the changes made in <commit> and applies it to the current branch.

$ git revert <commit>

Remove a file or multiple files from the staging area.

$ git reset <file>

Rebase the current branch onto the <base>.

$ git rebase <base>

Shows which files will be removed from the working directory.

$ git clean -n

Review your work

Display the changes between the working directory and the staging area

$ git diff <file>

Display the changes between the staging area and the repository.

$ git diff --staged <file>

An overview with reference labels and a history graph. One commit per line.

$ git log --oneline --graph --decorate

List commits that are present on the current branch and not merged into a branch name or a tag name.

$ git log ref..

List operations (checkouts, commits,...) made on a local repository.

$ git reflog

Tagging the Commits

Display the list of all tags.

$ git tag

Create a tag reference named name for the current commit.

$ git tag <name> <commit sha>

Abid Ali Awan | 2022