Git

Atul Phad

Introduction

Git is a version control system.

Git helps you keep track of code changes.

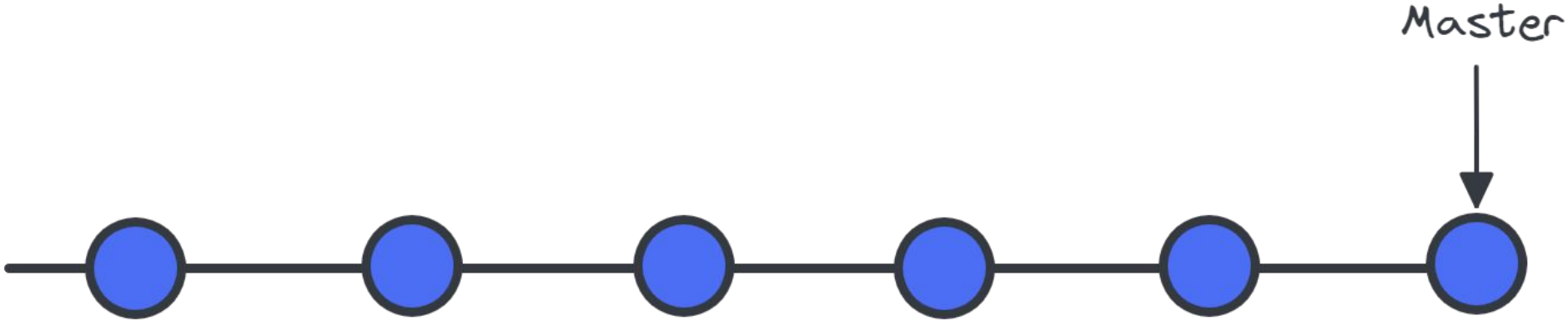
Git is used to collaborate on code

Introduction

Git is a distributed, open-source version control system. It enables developers and data scientists to track code, merge changes and revert to older versions - AWS. It allows you to sync changes with a remote server. Due to its flexibility and popularity, Git has become an industry standard as it supports almost all development environments, command-line tools, and operating systems.

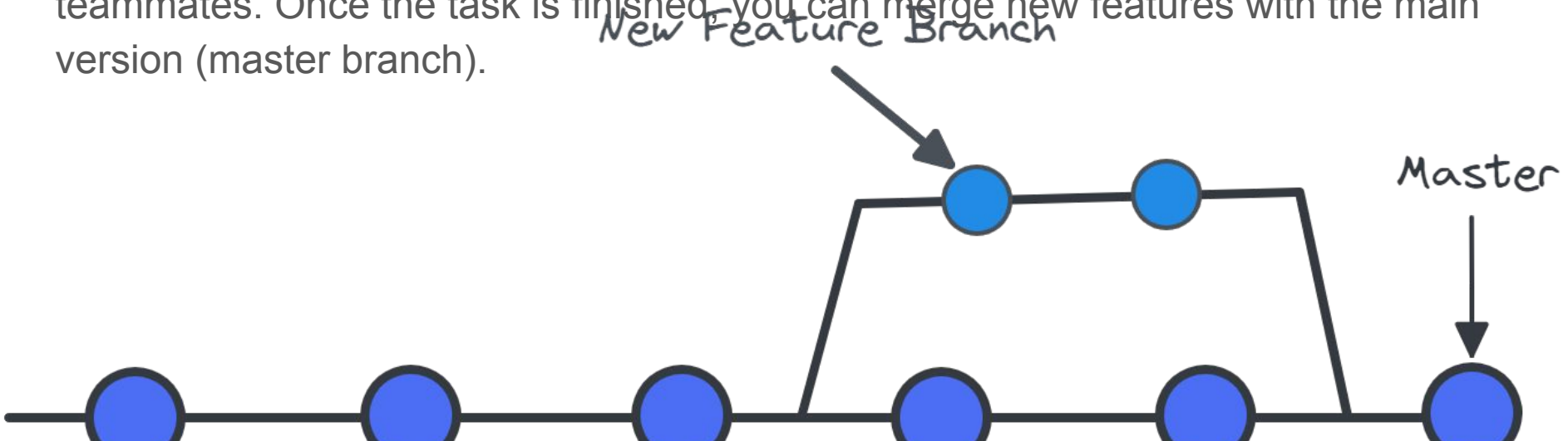
How it works?

Git stores your files and their development history in a local repository. Whenever you save changes you have made, Git creates a commit. A commit is a snapshot of current files. These commits are linked with each other, forming a development history graph, as shown below. It allows us to revert back to the previous commit, compare changes, and view the progress of the development project .



Branch

The branches are copies of the source code that works parallel to the main version. To save the changes made, merge the branch into the main version. This feature promotes conflict-free teamwork. Each developer has his/her task, and by using branches, they can work on the new feature without the interference of other teammates. Once the task is finished, you can merge new features with the main version (master branch).



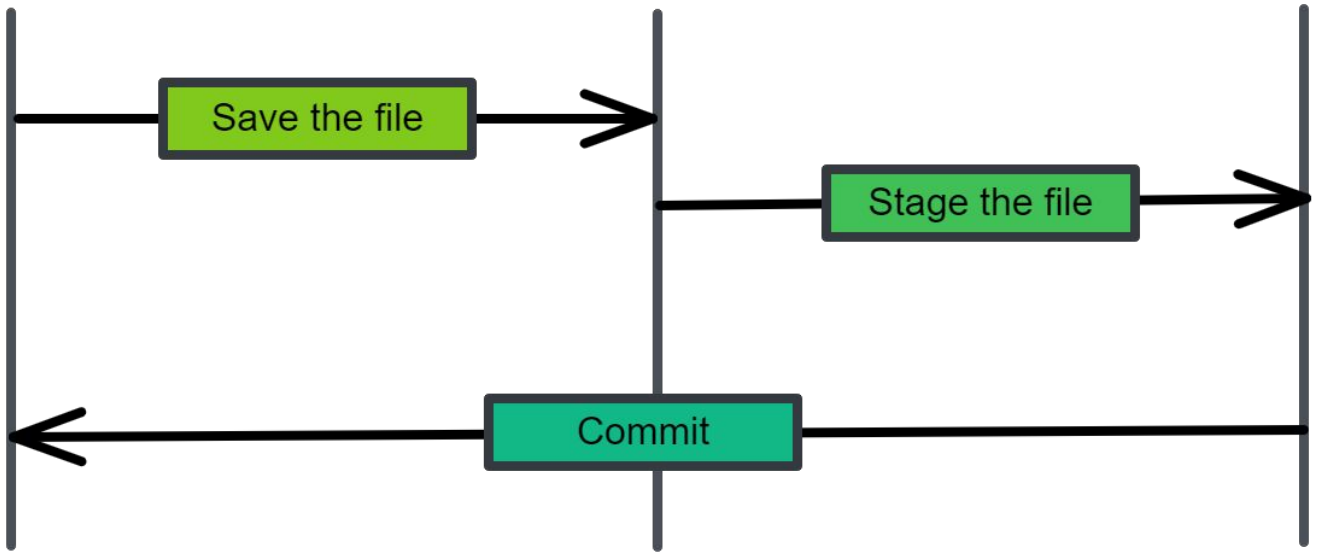
Commits

There are three states of files in Git: modified, staged, and commit. When you make changes in a file, the changes are saved in the local directory. They are not part of the Git development history. To create a commit, you need to first stage changed files. You can add or remove changes in the staging area and then package these changes as a commit with a message describing the changes.

Unmodified

Modified

Staged



Github

GitHub is a cloud software development platform. It is commonly used for saving files, tracking changes, and collaborating on development projects. In recent years, GitHub has become the most popular social platform for software development communities. Individuals can contribute to open-source projects and bug reports, discuss new projects and discover new tools.

Account setup and collab

Show on github

Commands

git init create a Git repository in a local directory.

git clone <remote-repo-address>: copy the entire repository from a remote server to remote directory. You can also use it to copy local repositories.

git add <file.txt>: add a single file or multiple files and folders to the staging area.

git commit –m “Message”: create a snapshot of changes and save it in the repository.

Contd..

git status shows the list of changed files or files that have yet to be staged and committed.

git push <remote-name> <branch-name>: send local commits to remote branch of repository.

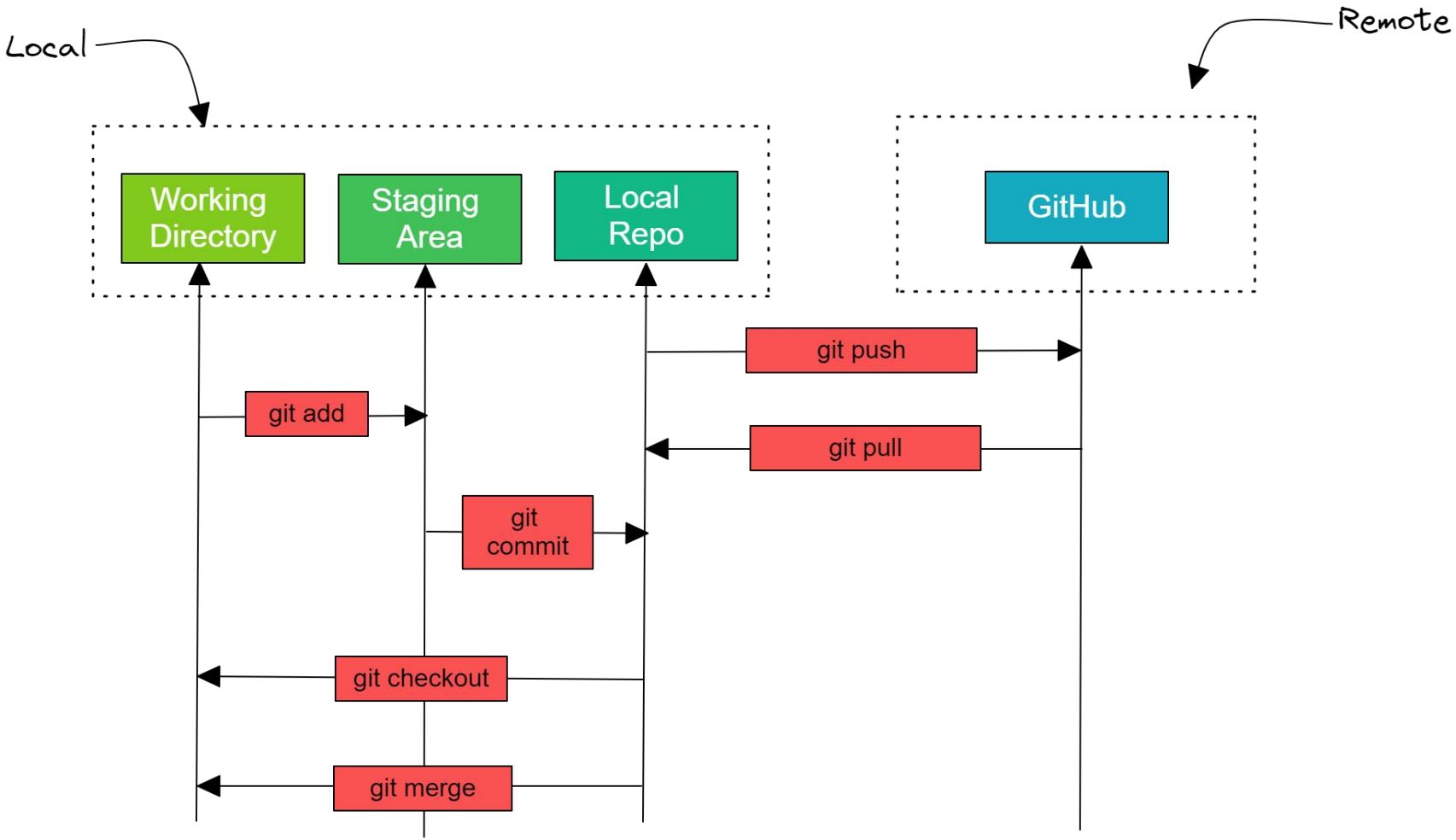
git checkout -b <branch-name>: creates a new branch and switches to a new branch.

git remote –v: view all remote repositories.

git branch –d <branch-name>: delete the branch.

git pull merge commits to a local directory from a remote repository.

git merge <branch-name>: after resolving merge conflicts the command blends selected branch into the current branch.


Local
Remote
Working Directory
Staging Area
Local Repo
GitHub
git push
git add
git pull
git commit
git checkout
git merge