

# Git Documentation

Documentation on Git along with GitHub, and its use in VS Code.  
From beginner to advanced user.

- [Glossary](#)
- [What is Git?](#)

# Glossary

## Basic Terms

Term one is a more simple version, while term two is a more advanced explanation. It is recommended to read both to get a good understanding of what each term is.

### Repository

1. A repository is the home for the source code and other related files.
2. The git repository is the .git folder in your root folder and hosts all the git information that is crucial for git's features to work properly.

### Branch

1. Branches are different versions of your code in your repository that run in parallel to the others.
2. Git branches are effectively a pointer to a snapshot of your changes.

### Stage

1. Stage, or to stage means to stage your file to go to the next commit.
2. Staged means that you have marked a modified file in its current version to go into your next commit snapshot.

### Commit

1. A commit is a version of your code that you save to be pushed from your local database to the repo.
2. Commits can be thought of as snapshots along the timeline of a Git project.

### Push

1. How you transfer commits from your local repository to a remote repository.
2. Pushing your commits to the centralized repository, comparing them, then adding them to the timeline if there is no merge conflicts that need to be satisfied.

### Pull

1. Pulling changes down from the centralized repository to your local repository, usually to sync the two.
2. Pulling the most recent commits and other history from the centralized repository to your local repository.

## Sync

1. Git fetching, pulling, then pushing, just automated into one command.

## Checkout

1. Switches branches if there are no modified files present on the current branch.
2. If a branch is specified, then all changes will be carried over to the target branch, allowing it to be committed to that branch instead.

## Rebase

1. Changing the base of your branch from one commit to another making it appear as if you'd created your branch from a different commit.
2. Doing this reorders the commit history and can break immediate merges, causing merge conflicts.

# Other Terms

## Terminal

A terminal, or a shell is a window that normally consists of a command prompt that allows you to interact with your pc in a command line interface.

There is a lot that can be done with a terminal, most of which won't be explained here.

# What is Git?

## Description

Git is a version/source control software that tracks any changes across a set of source files usually used for tracking source code among collaborating programmers.

It's goal is to be fast and light. It also supports distributed non-linear workflows aka branches.

It is widely efficient at what it does, being able to support huge projects!

For example, if you looked at the code for Epic Game's Unreal Engine 5, you'll see that the code is over hundreds of gigabytes in size!

Git also has great security measures and methods that you can use to protect and secure your code from unwanted insertions of code, or pushes.

Git keeps a history of everything that happens across the source code, or the repository.

## How Git Runs

It's possible that you've seen other version control software in the past. Git does not function like other software.

How Git saves and tracks your files is via snapshots.

Git takes snapshots of each file as it changes over time.

Git treats the repository like a mini filesystem, like how your PC would.

When you commit, Git takes a snapshot of what your filesystem looks like, aka all of the files that have changed, and stores a reference to that snapshot.

To be efficient, Git does not store files that are unchanged, only modifying the changed files.

### REMEMBER

Git has three states it'll be in: modified, staged, and committed.

Modified means that you have edited the file, but you have not submitted it to the database, or the repository yet.

Staged means that you have staged your files to be committed to the repository.

Committed means that you have committed your code to your local repository and are ready to push said changes to the centralized repository.

Remembering these are a must when it comes to working with this.