

# VERSION CONTROL

Version control, also known as source control, is the practice of tracking and managing changes to software code.

Version control systems are software tools that help software teams manage changes to source code over time. They help software teams work faster and smarter. Version control software keeps track of every modification to the code in a special kind of database. If a mistake is made, developers can turn back the clock and compare earlier versions of the code. Most widely used modern version control system in the world today is Git.

## GIT STATES

- **The Modified State** – This is when you add, remove, or change something in a file. Git has noticed the change but you have not informed Git of the change yet.
- **The Staged State** – This is when you inform Git of the change and tell Git to take note of that change. Commonly known as the "Staging Area", files in this stage are called the Staged Files. You can modify a file even after staging it, which lets you see the file both in a modified state and a staged state. The modifications made after the staging will not appear after a commit has been made.

- **The Committed State** – This is when Git has saved your modified changes.

## **GIT COMMANDS**

- **git clone <repository url>** - used to point to an existing repo and make a clone or copy of that repo in a new directory.
- **git fetch** - used to download content from a remote repository.
- **git checkout <branch name>** - lets you navigate between the branches created by git branch.
- **git add** - used to add file contents to the Staging Area.
- **git commit -m "<commit message>"** - Prior use of the is to select the changes that will be staged for the next commit. Then git commit is used to create a snapshot of the staged changes along a timeline of a Git project's history.
- **git push** - used to upload local repository content to a remote repository.
- **git pull** - used to fetch and download content from a remote repository and immediately update the local repository to match that content.
- **git status** - used to display the state of the repository and staging area. It allows you to see the tracked, untracked files and changes.
- **git log** - used to view the history of committed changes within a Git repository.
- **git init** - used to convert an existing, unversioned project to a Git repository or initialize a new, empty

repository.

- `git remote add origin <repository url>` - used to connect your local repository with remote repository.

## **STEPS TO FOLLOW WHEN CREATING A NEW REPOSITORY**

1. Create a GitHub profile.
2. Create a new repository on github.
3. Install Git.
4. Setup your username and email (git config).
5. Create a new empty folder for your project.
6. Navigate to your folder and run the `<git init>` command to convert your folder to repository.
7. Add new files and changes to your local repository
8. Push your changes and check your updates and history on GitHub.