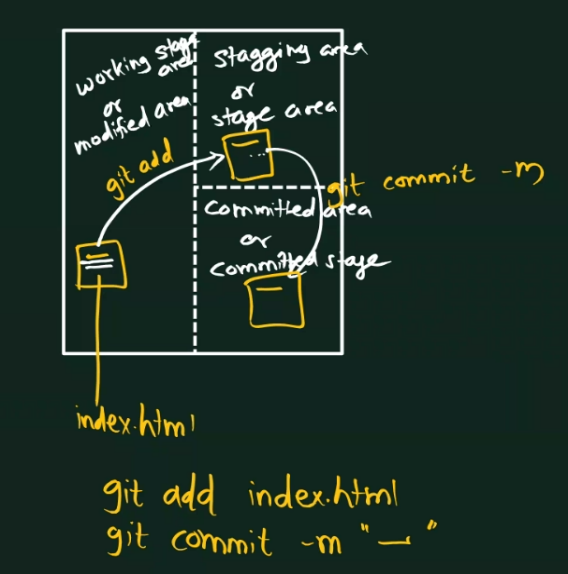
**Cheat sheet for GIT**

* **Git** is a distributed version control system designed to help developers track changes in their code, collaborate with others, and manage different versions of their projects efficiently.
* Unlike centralized version control systems, Git allows every developer to have a full copy of the repository, including its history.
* Git doesn’t need internet for their services.
* Git is a source code management tool

1. we have to install GIT in EC2-instance or Laptop
2. The git init command initializes a new Git repository. It sets up the necessary metadata and directories for Git to manage version control in a project. It does the logical sepration of directory
3. Then we have to config it

The git config command is used to configure Git's settings at different levels (system, global, and local). These settings control how Git behaves and manages repositories.

1. The git add command stages all changes (new, modified, or deleted files) in the current directory and its subdirectories for the next commit. It’s a shorthand for telling Git to include all relevant changes in the working directory into the staging area. (You’ve made multiple changes in your project and want to stage all of them for a commit.)
2. The git commit command is used to save changes from the **staging area** into the **local repository**. Each commit represents a snapshot of your project's state at a specific point in time, complete with a unique identifier and metadata like the author, timestamp, and a descriptive message.



* Git fetch command is used to download new data (commits, branches, tags) from a remote repository(origin) without altering your working directory or current branch.
* Git clone command downloads an entire repository (including all branches and history) from a remote source to your local machine.
* Branch – It allows you to work on different versions of your code simultaneously without interfering with the main or other branches. This is especially useful for managing features, bug fixes, and development workflows.

**Commands**

* git init – To initialize the git
* git config --global user.name "Sumit2606" – To config the username
* git config --global user.email [Sumitpedgulwar5@gmail.com](mailto:Sumitpedgulwar5@gmail.com) - To config the email
* git config --list – to show list of all config
* git pull <https://github.com/Sumit2606/gitrepo.git> - To pull the files from repo
* git remote add origin <https://github.com/Sumit2606/gitrepo.git> - To store the working repository link in origin variable
* git remote rename old-name new-name – To rename the remote repository
* git status – To check the status of files
* git add index.html – To add the files in staging area
* git commit -m "first commit" – To commit the changes
* git push origin master – To push the files to repo
* git log – To check commits
* git show commit-id – To check the details about commit
* git diff – To check the difference between file content of working area & staged area
* git diff --staged – To check the difference between file content of file present in staged and committed area
* git diff HEAD – To check the difference between file content present at committed area and working area
* gut difftool – For visual representation of difference
* git reset file – To move file from staging area to working area
* git reset HEAD -
* git fetch - To download updates for review without merging.
* git rebase - To align your branch with upstream changes cleanly.
* git clone - To copy a repository to your local system.
* Git branch – To check the branches
* Git remote – To check the remote directories
* Git branch branch-name – To create the branch
* Git checkout branch-name – To switch the branch with all content
* Git branch -D branch-name – To delete the branch
* Git checkout -b branch-name – To create and switch the branch
* Git merge branch-name – To merge the branches
* Git stash apply – To switch the branch without making any changes to working branch
* Git tag – To give tags to commit
* Git ignore – To ignore