Git vs GitHub

Git:

Git is a distributed version control system (VCS) used to manage and track the changes in your codebase.

It allows you to work on your code locally and track changes (commits), revert to previous versions, create branches for features, and collaborate with other developers.

GitHub:

GitHub is a platform that provides hosting for software development and version control using Git.

It's a cloud-based service where you can store your Git repositories and collaborate with other developers.

GitHub offers features like issue tracking, pull requests, code reviews, and more, making it a social platform for code sharing.

Git Workflow :

The general Git workflow typically follows these steps:

1)Clone a repository (if you're starting from an existing repository on GitHub):

This creates a local copy of the repository on your machine.

git clone https://github.com/username/repository-name.git

2)Create a branch (to work on a new feature or bugfix):

It's a good practice to create a new branch so you don't mess with the main codebase.

git checkout -b your-branch-name

3)Make changes (edit, add, or delete files):

After working on your code, you can check the status of the modified files:

git status

4)Add changes to the staging area (prepare for commit):

This step tells Git which files you want to commit.

git add filename

Or to add all modified files:

git add .

5)Commit the changes (save changes to your local repository):

Committing saves the changes with a message describing what you’ve done.

git commit -m "Describe your changes"

6)Push the changes to GitHub:

This sends your local commits to the remote GitHub repository.

git push origin your-branch-name

7)Create a Pull Request (PR):

On GitHub, open a PR to merge your branch into the main branch. This allows others to review your changes before they are merged.

Git Commands to Push Your Code from Your Machine to GitHub

1)Initialize a Git repository: If you're starting from scratch, initialize a Git repository in your project folder.

git init

2)Link the local repository to GitHub: You need to link your local repository to the remote GitHub repository.

git remote add origin https://github.com/username/repository-name.git

3)Add files to staging area:

git add .

4)Commit your changes:

git commit -m "Initial commit or description of changes"

5)Push to GitHub: Push your code to GitHub using the git push command. If you're pushing to the main branch for the first time, you'll need to set the upstream.

git push -u origin main

After the first push, you can just use:

git push

Summary of Common Git Commands:

git status: View changes and the status of files.

git add .: Stage all changes.

git commit -m "message": Commit the staged changes.

git push origin branch-name: Push your branch to GitHub.

git pull origin branch-name: Fetch changes from the remote GitHub repository.

git clone repository-url: Clone a remote repository to your local machine.

git checkout -b branch-name: Create a new branch and switch to it.

This covers the basics of the Git workflow and pushing code from your local machine to GitHub.