Name: Apoorv Kashyap

INTERN ID: MST03-004

Git vs. GitHub

Git

- **Definition**: Git is a distributed version control system that allows multiple developers to work on a project simultaneously without overwriting each other's changes.
- **Purpose**: It helps track changes in source code during software development.
- **Local Repository**: You work on a local copy of the project.
- **Commands**: Git commands are used through the terminal or command line interface (CLI).

GitHub

- **Definition**: GitHub is a cloud-based hosting service that manages Git repositories.
- **Purpose**: It provides a graphical interface to manage Git repositories, collaboration tools, and additional features such as issue tracking, project management, and CI/CD.
- **Remote Repository**: You push your local Git repository to GitHub for remote access and collaboration.
- **Web Interface**: You can use GitHub's web interface to manage repositories, view commits, and collaborate with others.

Important Git Commands

1. **Basic Commands**
    - git init: Initialize a new Git repository.
    - git clone [repository_url]: Clone an existing repository.
    - git status: Check the status of your files in the working directory and staging area.
    - git add [file]: Add files to the staging area.

- git commit -m "commit message": Commit changes in the staging area with a descriptive message.
- git push [remote] [branch]: Push your changes to a remote repository.
- git pull [remote] [branch]: Fetch and integrate changes from a remote repository.

2. **Branching and Merging**
   - git branch: List all branches in the repository.
   - git branch [branch_name]: Create a new branch.
   - git checkout [branch_name]: Switch to a different branch.
   - git merge [branch_name]: Merge a branch into your current branch.
3. **Undoing Changes**
   - git reset [file]: Unstage a file.
   - git checkout -- [file]: Discard changes in the working directory.
   - git revert [commit_hash]: Create a new commit that undoes changes made in a specific commit.