**Git** and **GitHub** is essential for version control, collaboration, and managing code effectively. Here's a step-by-step guide to get you started:

**🔧 What is Git and GitHub?**

| **Git** | **GitHub** |
| --- | --- |
| A distributed version control system for tracking changes in code. | A cloud-based platform that hosts Git repositories and enables collaboration. |

**📘 Git Basics: Core Concepts**

**1. Repository (Repo)**

A project directory tracked by Git.

**2. Commit**

A snapshot of your changes.

**3. Branch**

A separate version of your codebase used for features or fixes.

**4. Merge**

Combines changes from one branch into another.

**5. Remote**

A GitHub (or other) server where your code is pushed.

In Git, there are **three primary stages** that files go through during the version control process:

**✅ 1. Working Directory (Working Tree)**

* This is your local file system where you write and edit code.
* Any changes you make (new files, modifications, deletions) appear here first.
* These changes are **not yet tracked** by Git.

**✅ 2. Staging Area (Index)**

* Files added here are *prepared* for the next commit.
* You explicitly stage changes using git add.

**✅ 3. Repository (Local Repository)**

* Once you commit staged changes, they’re saved to your local Git database.
* This is your project history.

🔁 Flow Summary:

[Working Directory] → git add → [Staging Area] → git commit → [Local Repository] → git push → [Remote Repository]

**🔧 Git & GitHub Workflow: Step-by-Step**

**✅ Step 1: Initialize Git in Your Project**

git init

This creates a .git folder and starts tracking the project.

✅ Step 2: Check Git Status

git status

See untracked/modified files.

**✅ Step 3: Stage Files for Commit**

git add .

Stages all files for the next commit.

✅ Step 4: Commit Your Changes

git commit -m "Initial commit"

Saves the changes in your local repository with a message.

**✅ Step 5: Create a New Repository on GitHub**

* Go to <https://github.com>
* Click **"New Repository"**
* Give it a name (e.g., my-project)
* Click **"Create repository"**

✅ Step 6: Link Local Repo to GitHub

git remote add origin <https://github.com/yourusername/my-project.git>

✅ Step 7: Push Code to GitHub

git branch -M main

git push -u origin main

**✅ Step 8: Make Further Changes and Repeat**

# After editing files

git add .

git commit -m "Updated feature"

git push

**✅ Step 9: Pull Latest Changes (if working in team)**

git pull