Introduction to Git and GitHub

What is Git?

Git is a **distributed version control system** used to track changes in source code during software development. It lets multiple developers work on a project at the same time, without overwriting each other's work.

Key Concepts:

- **Repository (repo):** A project tracked by Git. It can be local (on your machine) or remote (like on GitHub).
- **Commit:** A snapshot of your project at a given point. You commit to save changes and write a message explaining what you changed.
- **Branch:** A parallel version of your code. Useful for adding features or fixing bugs without affecting the main codebase.
- **Merge:** Combining changes from different branches.
- **Staging Area:** A buffer between your working directory and your commit. You "stage" changes you want to include in the next commit.

What is GitHub?

GitHub is a **cloud-based hosting platform for Git repositories**. It adds collaboration features like pull requests, issue tracking, code reviews, and team management.

GitHub = Git + Collaboration + Cloud

You can:

- Store your code online
- Collaborating with teammates
- Contribute to open-source projects
- Use GitHub Actions to automate CI/CD workflows

Typical Git Workflow (Local to GitHub):

```
git init                    # Initialize a local repo
git clone <repo-url>        # Clone a remote GitHub repo
git status                  # Check what's changed
git add <file>              # Stage a file for commit
git commit -m "Message"     # Save the snapshot
git push origin main        # Push changes to GitHub
git pull origin main        # Get latest changes from GitHub
```

Example Scenario:

You're working on a web app. You create a new branch called feature/login, develop the login page, commit your changes, and push to GitHub. Then, you create a **pull request** to merge your changes into the main branch after review.

Final Thoughts

- Git is for version control.
- GitHub is for collaboration.
- Together, they form the backbone of modern software development and DevOps.

Visual Diagram of Git