



Hey 🙌 Guys


Introduction to **Git** and **GitHub**

 **Git:** A version control system **that helps track changes in code**, similar to how a bank account tracks transactions.


 **GitHub:** A platform that allows developers to **store and manage their code using Git**.

Key Concepts

 **Version Control System:** A tool that tracks the history of changes in a project, allowing developers to revert to previous versions if needed.

 **Collaboration:** Git enables multiple developers to work on the same project without overwriting each other's changes.


Git Basics

 **Installation:** Git can be installed on various operating systems like Windows and Mac. Visual Studio Code is a recommended code editor.


💡 Basic Commands:

- **git clone <repository-url>:** 📁 Copies a repository from **GitHub to your local machine.**
- **git status:** 🔍 Shows the current status of the repository, including modified and untracked files.
- **git add <file>:** ➕ Stages changes to be committed.
- **git commit -m "message":** 📝 Records the staged changes with **a message describing the changes.**
- **git push origin main:** 🚀 Uploads local changes to the remote repository on GitHub.




Working with Repositories

-  **Repository (repo):** A storage space for your project on GitHub, where you can upload your code and track changes.
-  /  **Public/Private Repositories:** Repositories can be public (visible to everyone) or private (restricted access).

Commit History


 **Commits:** Each commit represents a snapshot of your project at a specific point in time. View commit history to understand project evolution.

Branching in Git


-  **Branches:** Used to develop features independently. The default branch is often called "main".
-  **Creating a Branch:** `git branch [branch-name]`.
-  **Switching Branches:** `git checkout [branch-name]`.

Merging Branches

 **Merging:** Combines changes from one branch into another.
Use `git merge [branch-name]`.

 **Pull Requests (PR):** A way to propose changes to a repository, allowing for code review before merging.

Handling Merge Conflicts

 **Merge Conflicts:** Occur when two branches have changes in the same line of a file. Git prompts you to resolve these conflicts manually.

◀ Undoing Changes

- ♻️ **Undoing Commits:** `git reset HEAD~1` to undo the last commit.
- ↺️ **Resetting Changes:** `git reset [file]` to unstage changes.

🔪 Forking Repositories

- 🍴 **Forking:** Creates a copy of someone else's repository in your GitHub account, allowing you to make changes without affecting the original.

🧠 Conclusion

- Understanding Git and GitHub is essential for modern software development, enabling efficient collaboration and version control for projects of all sizes. Familiarity with commands and workflows enhances productivity and code management.

Created by Imtiaz 🥰 Ali