

Mastering Git



by

➤ Naveen Jayachandran





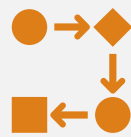
What is Git?

- **Git** is a tool that helps track changes in your files.

Key Points:

- Keeps a history of your work.
- Allows you to collaborate with others.
- Handles projects of all sizes.

Why Git?



Track Changes: See who changed what and when.



Collaborate: Work with others without overwriting each other's work.



Recover: Roll back to previous versions if needed.

Git Basics



Repository (Repo): Where your project lives.



Commit: A snapshot of your project.



Branch: A separate line of development.



Merge: Combining changes from different branches.

Getting Started with Git



Installation:



Windows: [Download Git](#)



macOS: brew install git



Linux: sudo apt install git or sudo yum
install git

Core Commands

- **Initialize Repo:** `git init`
- **Clone Repo:** `git clone <url>`
- **Check Status:** `git status`
- **Stage Changes:** `git add <file>`
- **Commit Changes:** `git commit -m "message"`
- **Push Changes:** `git push`
- **Pull Changes:** `git pull`

Branching Made Simple

- **Create Branch:** `git branch <name>`
- **Switch Branch:** `git checkout <name>`
- **Merge Branches:** `git merge <name>`
- **Delete Branch:** `git branch -d <name>`

Collaborate with Ease

- **Fork Repositories:** Make your own copy.
- **Pull Requests:** Suggest changes to the main project.
- **Conflict Resolution:** Resolve conflicts with visual tools or manually.

Best Practices



Commit Often: Regularly save progress.




Clear Messages: Write descriptive commit messages.



Branch Strategically: Use branches for new features or fixes.



Push Regularly: Keep remote repositories updated.



Resources for Learning



Interactive Tutorials:

[GitHub Learning Lab](#)



Codecademy Git
Course



Documentation: [Git Documentation](#)

Thank you

