

Git Commands Sheet

Starting with Git

- `git --version`: Check if Git is installed on your computer.
- `git init`: Start Git in your project folder. It tells Git to track this project.
- `git clone <repo-url>`: Copy a project from GitHub to your computer.

Checking File Status

- `git status`: Shows which files are new, changed, or ready to save.
- `git add <file.txt>`: Tell Git to save this specific file.
- `git add .`: Add all files.
- `git commit -m "message"`: Save the changes with a message.
- `git diff`: See exactly what you changed in the file.

Working with Branches

- `git branch`: Show all branches (different versions of code).
- `git branch <name>`: Create a new branch, e.g., `git branch dev`.
- `git checkout <name>`: Move to another branch, e.g., `git checkout dev`.
- `git checkout -b <name>`: Create a branch and move to it, e.g., `git checkout -b feature`.
- `git branch -d <name>`: Delete a branch you don't need.

Combining Code

- `git merge <dev>`: Join one branch into another.
- `git rebase`: Put your changes on top of the latest code for a clean history.

GitHub Online Repo

- `git remote -v`: See GitHub link connected to your project.
- `git remote add origin <repo-url>`: Connect your project to GitHub.
- `git push origin main`: Send your code to GitHub.
- `git pull origin main`: Get latest code from GitHub.
- `git fetch`: Check new updates on GitHub without adding them.

Undo and Fix Mistakes

- `git reset --soft`: Undo last commit but keep the code.
- `git reset`: Remove file from staging but keep changes.
- `git reset --hard`: Delete last commit and code completely.
- `git checkout -- <file>`: Undo changes in a file.

See Old History

- `git log`: See all saved changes history.

- `git log --oneline`: Short history in one line.
- `git show`: See details of one commit.

Additional Commands

- `git stash`: Hide your work temporarily.
- `git stash apply`: Bring hidden work back.
- `git tag`: Mark a version like v1.0.
- `git push --tags`: Send version tags to GitHub.
- `git config --global user.name "Your Name"`: Set your name
- `git config --global user.email "email@gmail.com"`: Set your email.

Daily Simple Git Workflow

- `git status`
- `git add .`
- `git commit -m "save work"`
- `git pull`
- `git push`