

GIT COMMANDS

1. Setup and Configuration

Commands:

- `git config --global user.name "Your Name":` set git username
- `git config --global user.email "you@example.com":` set git password
- `git config --list:` show all git configurations.

2. Repository operations

Commands:

- `git init:` Initialize a new Git repository.
- `git clone <repo-url>:` Clone a remote repo to your local machine.
- `git remote -v:` View connected remote repositories.
- `git remote add origin <url>:` Add a remote repository.

3. Basic file operations

Commands:

- `git status:` Show current repo status.
- `git add <file>:` Stage a file for commit.
- `git add .:` Stage all changes.
- `git commit -m "message":` Commit staged changes with a message.
- `git rm <file>:` Delete a file and stage the removal.
- `git mv oldname newname:` Rename a file and stage it.

4. Work with new branches

- `git branch:` List all branches.
- `git branch <branch-name>:` Create a new branch.
- `git checkout <branch-name>:` Switch to another branch.
- `git checkout -b <branch-name>:` Create and switch to a new branch.
- `git merge <branch-name>:` Merge branch into current branch.
- `git branch -d <branch-name>:` Delete a branch.

5. Working with remote

Commands:

- `git fetch:` Fetch changes from remote (doesn't merge).
- `git pull:` Fetch and merge changes from remote.
- `git push:` Push changes to remote repository.
- `git push -u origin <branch>:` Push and set upstream for a branch.

6. History and logs

Commands:

- `git log:` Show commit history.
- `git log --oneline:` Show short commit history.
- `git diff:` Show changes not staged.
- `git diff --staged:` Show staged changes.

7. Advanced / recovery

Commands:

- `git stash:` Save uncommitted changes temporarily.
- `git stash pop:` Reapply stashed changes.
- `git checkout <commit>:` Checkout a specific commit (detached state).