

SETUP & CONFIG

git config --global user.name "[name]"
Sets your commit name

git config --global user.email "[email]"
Sets your commit email

git init
Initialize a new local repo

git clone [url]
Clone a repository

INSPECTION & LOG

git log
Show commit history

git log --oneline
Compact history

git diff
Show unstaged changes

git diff --staged
Show staged changes

git show [commit]
View specific commit

GLOSSARY

Working Directory Current file state.
Staging Area Files ready for commit.
Repository Committed history.
Remote Hosted version (GitHub/GitLab).
HEAD Last commit in current branch.

BASIC SNAPSHOTTING

git status
View changed files

git add [file]
Stage a specific file

git add .
Stage all changes

git commit -m "[msg]"
Commit staged changes

git commit --amend
Edit last commit

UNDOING CHANGES

git reset [file]
Unstage a file

git reset --hard HEAD
Discard all local changes

git checkout -- [file]
Restore file to last commit

git revert [commit]
Create a compensating commit

BRANCHING

git branch
List local branches

git checkout -b [name]
Create and switch to branch

git checkout [name]
Switch to a branch

git merge [branch]
Merge branch into current

git branch -d [name]
Delete a branch

ACME SPECIFICS

Branch Naming

feat/[jira-id]
For new features

fix/[jira-id]
For bug fixes

docs/[description]
Documentation only

Commit Format

[JIRA-ID] description

Pre-flight

npm run lint
Run before committing

npm test
Ensure all tests pass

REMOTE SYNC

git fetch
Get latest from remote

git pull
Fetch and merge changes

git push origin [branch]
Push branch to remote

git remote -v
List remote URLs

STASHING

git stash
Temporarily store changes

git stash pop
Restore last stashed

git stash list
List all stashes

git stash drop
Discard last stash