Git Cheat Sheet

Commits

git commit --amend	Change last commit
git revert <commit-id>	Revert commit (creates new commit)
git reset --hard HEAD	Discard all changes in working directory
git reset --hard <commit>	Move HEAD to commit and discard all changes since
git tag -a <name>	Create tag object for current commit
git clean -fxd	Clean everything not tracked by repository

History

git cherry-pick <rev>	Apply commit on current branch
git cherry-pick <rev1>..<rev2>	Apply range (excludes rev1)
git show <rev>:<path>	Show specified version of file
git blame <file>	Show changes for file with time and author
git grep <regex> $(git rev-list --all)	Search all history
git grep <regex> $(git rev-list <rev1>..<rev2>)	Search some history
git log --diff-filter=A -- */<file-name>	Find commit where file was added

Branches

git branch -f <branch> <commit-id>	Move branch pointer to another commit
git branch -m <new-name>	Rename current branch
git push origin --delete <name>	Delete remote branch
git fetch --prune	Remove deleted remote branches

Merging

git rebase -i <branch>	Rebase interactively on top of branch
git merge --abort	Cancel merge
git reset --merge ORIG_HEAD	Undo merge
git merge-base <branch1> <branch2>	Most recent common ancestor

Stash

git stash save “<message>”	Push changes to stash
git stash pop	Apply stash
git stash drop	Clear stash without applying

Remote

git pull --rebase	Pull using rebase instead of merge
git push --tags	Push tags

Created by Carlos Anderson (www.carlosjanderson.com)