

Initialize

<code>git init</code>	Initialize repository
<code>git remote add <origin/alternate-name> <repo url></code>	Link local repo with <repo URL>
<code>git push -u origin master</code>	Push committed changes (-u and track branch with remote)

Commit

<code>git add .</code>	Adds all modified and new (untracked) files in the current directory and all subdirectories to the staging area
<code>git commit -m <commit message></code>	Locally commit
<code>git checkout .</code>	Revert changes to local copy
<code>git reset</code>	Revert changes made to the index (i.e., that you have added). Warning this will reset all of your unpushed commits to master!
<code>git revert <commit 1> <commit 2></code>	Revert a change that you have committed
<code>git clean -f</code>	Remove untracked files (e.g., new files, generated files)
<code>git clean -fd</code>	Remove untracked directories
<code>git stash</code>	Stash the changes away in a working directory
<code>git diff <branch 1> <branch 2></code>	Compare two branches
<code>git diff <branch 1> <branch 2> -name-only</code>	Compare two branches, file names only
<code>git diff <branch></code>	Compare <branch> with current branch
<code>git rm <file> --cached</code>	Remove file from remote, while retaining local copy

Branching, Merging

<code>git checkout -b <branch> master</code>	Create <branch> from master
<code>git push -u origin <branch></code>	Push <branch>
<code>git merge <branch></code>	Merge <branch> into current branch
<code>git branch -d <branch></code>	Delete local branch if it is merged, if not -D to force delete

Branching, Merging (cont)

<code>git push <remote_name> --delete <branch></code>	Delete remote branch
<code>git branch <branch></code>	Create branch <branch>
<code>git branch [-a]</code>	List branches, -a including remote, * indicates current branch
<code>git rebase <branch></code>	Rebase from <branch> into current branch, i.e. take <branch> and then apply current branch's commits onto it to create linear history
<code>git branch -m <old> <new></code>	1. Rename your local branch
<code>git push origin :<old> <new></code>	2. Delete the <old> remote branch and push the <new> local branch
<code>git push origin -u <new></code>	3. Reset the upstream branch for the <new> local branch
<code>git diff --name-status <b1>..<b2></code>	Show diffs (filenames only) between two branches
<code>git diff --name-status master</code>	Show diffs (filenames only) between master and current branch

Admin, Trouble-shooting

<code>git log [-p] [-n]</code>	Shows commit history, last n, -p differences
<code>git remote set-url origin <new url></code>	Change remote url
<code>git remote -v</code>	List remote information
<code>git config --global credential.helper cache</code>	Set git to use the credential memory cache
<code>git config --global credential.helper 'cache --timeout=3600'</code>	set the cache to timeout after 1 hour (setting is in seconds)
<code>git config [--global] user.name "FIRST_NAME LAST_NAME"</code>	Set (global/repo) username for commits
<code>git config [--global] user.email "MY_NAME@example.com"</code>	Set (global/repo) email for commits

