Git Commands

SETUP

git config –global user.name "<ravi raushan>"	*-set username*
git config –global user.email "<raviraushan@gmail.com>"	*-set email*
git config –list	*-view all the config*
git config user.name	*-view the name of the user*
git help	*-git help menu*
git help commit	*-description of commit cmd, replace the commit with the cmd you want*

INIT & CLONE

git init	*-initialize an existing directory as a git repo*
git clone <repo_url>	*-clone an entire repo from a hosted location via url*
git clone --branch <branch_name> <repo_url>	*-clone a specific branch from a hosted location via url*

STAGE & SNAPSHOT & COMMIT

git status	*-show modified files in working directory, staged for your next commit*
git add <file_name>	*-add a file to the stage for your next commit*
git add .	*- add all modified files to the stage for your next commit*
git reset <file_name>	*-unstage a file while retaining the changes in working directory*
git commit -m "<my first commit>"	*- commit your staged content as a new commit snapshot*
git commit -a -m "<msg>"	*- commit all the changed files which are added to the git add and changed since then*
git diff	*- diff of what is changed but not staged*
git diff –staged	*- diff of what is changed but not staged*
git diff <first_branch> <second_branch>	*- differences between the first_branch and second_branch*

BRANCH & MERGE

git branch	*- view all the branches*
git checkout <branch_name>	*-checkout to the branch_name specified*
git branch <branch_name>	*-create a new branch named branch_name*
git checkout -b <branch_name>	*-create the new branch and checkout to the newly created branch*
git branch <branch_name><hash>	*-create a branch by defining a commit with its hash*
git merge <branch_name>	*-merge the specified branch with current active branch*

PUSH & ORIGIN

git remote	*- view the locally set name for remote repo*
git remote -v	*- view the remote url of the locally set alias repo name*
git remote add origin <remote_repo_url>	*-set the repo url with alias origin*
git push origin <branch_name>	*-push to the alias origin to the specified branch*
git remote set-url origin <remote_repo_url>	*-change the repo url of origin alias*
git push <repo_name> <branch_name>	*-push to the repo in a specified branch*

FETCH & PULL (PULL = FETCH +MERGE)

git fetch <alias>	*-fetch down all the branches from that git remote (ex, alias = origin)*
git fetch -all	
git merge <alias> <branch>	*-merge a remote branch into your current branch to bring it to up to date*
git pull <alias> <branch>	*-ex, git pull origin master*

TEMPORARY COMMITS

git stash	*- save modified and staged changes*
git stash list	*- list stack-order of stashed file changes*
git stash pop	*- write working from top of stash stack*
git stash drop	*- discard the changes from top of stash stack*

LOG & COMPARE

git log	*- show the commit history for the currently active branch*
git log branchB..branchA	*- show the commits on branchA that are not on branchB*
git log –follow <file_name>	*- show the commit that changed file, even across renames*
git diff branchB...branchA	*- show the diff of what is in branchA that is not in branch*
git show <SHA>	*- show any object in Git in human-readable format*

REWRITE HISTORY

git rebase <branch>	*- apply any commits of current branch ahead of specified one*
git reset –hard <commit>	*- clear staging area, rewrite working tree from specified commit*