Git Cheatsheet

A Distributed Version Control System.

CODE2BITS

Create a Repository

Create new local repository
$ git init

Create new local repository with name
$ git init {repo name}

Download remote repository
$ git clone {repo url}

Managing Branches

List the branches, current one highlighted
$ git branch

Create a new branch with name
$ git branch {branch_name}

Delete the branch
$ git branch -d {branch_name}

Switch to specified branch and update working directory
$ git checkout {branch_name}

Join changes from branch into current branch
$ git merge {branch_name}

Edit Repository

Add file contents to the index
$ git add {filename}

Add all files (new, modified, deleted) to staging
$ git add -A

Record changes to the repository with message
$ git commit -m "{Message}"

Resets the index and working directory from HEAD.
$ git reset --hard HEAD

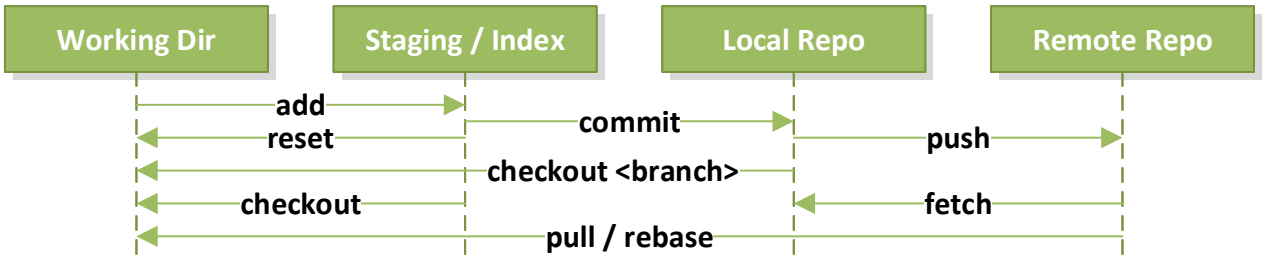
Resets file in index, keep working tree
$ git reset {file}

Resets the file in index and working tree
$ git reset --hard {file}

Remove file from the working tree and index.
$ git rm {file}

Remove file from the index and not index.
$ git rm --cached {file}

Move or rename a file, directory or symlink
$ git mv {old} {new}



Inspect Repository

Show the working tree status
$ git status

View changes: Working Dir & Index
$ git diff

View changes: Index & Local Repo
$ git diff --cached

View changes: Working Dir & Local Repo (HEAD)
$ git diff HEAD

View changes in Local Repo between HEAD & HEAD-1
$ git diff HEAD HEAD^

View changes between Index & Local Repo (HEAD)
$ git diff --cached HEAD

View changes in Local Repo between id1 & id2
$ git diff {commit1} {commit2}

View changes in Local Repo between tag1 & tag2
$ git diff {tag1} {tag2}

Show the commit history
$ git log

Show commit history
$ git log --oneline --graph --decorate –all

Show log message and textual diff for the commit
$ git show {commit}

Show what was changed in file by whom
$ git blame {filename}

Git Cheatsheet

A Distributed Version Control System.

CODE2BITS

Git Configuration

Set user name property in global config
$ git config --global user.name “{name}”

Set user email property in global config
$ git config --global user.email “{email}”

Display user name property value
$ git config user.name

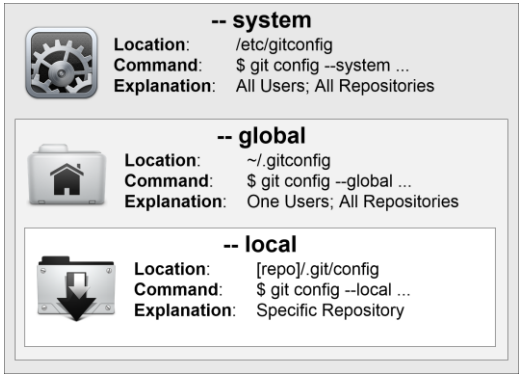
Display user name property value
$ git config user.email

Display all property values in global config
$ git config --global --list

Open default editor to modify local config file.
$ git config -e --local

Open default editor to modify global config file.
$ git config -e --global

Open default editor to modify system config file.
$ git config -e --system



Sync Remote Repository

Shows a list of existing remote repositories
$ git remote

Add a remote name for repository at url
$ git remote add {name} {repo_url}

Shows list of existing remote repositories with urls
$ git remote -v

Download objects and refs from another repository
$ git fetch

Fetch from and integrate with another repository or a local branch
$ git pull {alias} {branch_name}

Push branch to upstream remote repository
$ git push {alias} {branch_name}

Manage Tags

Create a tag reference
$ git tag {tag name}

Create an “annotated” tag reference
$ git tag -a {tag name}

List all the tags
$ git tag --list

Delete specific tag
$ git tag --delete {tag name}

Save / Stash Code Fragments

Record the current state of working directory
$ git stash

Record the current state of working directory (Include untracked)
$ git stash -u

List the stash entries with its name.
$ git stash list

Remove stash state and apply to working tree
$ git stash apply

Remove a single stash entry from the list of stash entries
$ git stash drop

Same as git stash apply + git stash drop
$ git stash pop

Remove all the stash entries.
$ git stash clear

Git Help

Provide help on the different git commands
$ git help

Provide help on the specific git command
$ git help {command}

Provide help on the specific git command
$ git {command} --help