

GIT Cheat Sheet

Know

<code>git status</code>	View tracked, untracked, modified of local repo
<code>git log</code>	Detailed log
<code>git log --oneline</code>	Single line log
<code>git diff <hash1> <hash2></code>	Difference between two commits. Shows to the move happened from <hash1> to <hash2>
<code>git tag</code>	List all the tags
<code>git branch</code>	List of all the branches
<code>git branch -a</code>	Detailed listing of branches
<code>git help <action/verb></code>	git's help
<code>git log --follow <filename></code>	Lists the file changes including renames
<code>git remote -v</code>	Shows the remote repositories with fetch and pull remote urls

Create

<code>git init</code>	Initiate a local Repository
<code>git clone <url></code>	Get remote repo along with its revision history. Creates a folder with the repo name.
<code>git branch b1</code>	Creates a branch b1
<code>git checkout -b b1 <hash1></code>	Creates a branch b1 with changes from <hash1>

Make changes

<code>git add <file name></code>	Adds a file/modification to staged area
<code>git commit -m "<commit-message>"</code>	Commit a change to a local repository
<code>git push</code>	Push the current to remote repository

Sync

<code>git merge <b1></code>	Merge the changes of branch b1 into current
<code>git pull</code>	Pull the changes from remote and merge into local
<code>git fetch</code>	Get the changes from remote. Does not merge
<code>git rebase <b1></code>	Rebase from b1
<code>git checkout <b1></code>	Checkout branch b1 into working directory

Delete

<code>git rm <filename></code>	Deletes from working directory and stages the delete
<code>git rm -cached <file></code>	Untrack the file

<code>git mv <file1> <file2></code>	Rename the file and stage the rename
<code>git branch -d <b1></code>	Delete a branch

Undo changes

<code>git reset <hash></code>	Resets history to the commit version. Preserves changes in working director
<code>git reset --hard <hash></code>	Discards history and working directory and moves the current state to <hash>
<code>git revert <hash></code>	Takes the hash, merges to HEAD and makes a commit. History is preserved
<code>git amend</code>	Amend the previous commit