|  |
| --- |
| **GitHub Notes:** |
| * git --version  OR git version // Show git version * git help // Get general help * PRESS ‘q’ OR ‘Q’  // Exit to terminal * pwd // Show current directory (folder) (present working directory (folder)) * git config --global user.name "Nadir Shah" // Configuring git user name globally * git config --global user.email ["lonewolfnadhu@gmail.com"](mailto:\"jason@jasongtaylor.com\") // Configuring git email id globally * git config --global --list // List out all global configuration settings * cat ~/.gitconfig // Seeing git's user-based configuration file * git init git-demo // Initialize/create empty git repository * cd workspace/github/ // Change current directory (folder) to github directory (folder) * ls // Show current files in that directory (folder) * ls -a  // Show current filesin that directory (folder) including hidden files * vim README.md // Open README.md file * git status // Changes/modifications in the current repository // Shows which files have been modified in the working directory (folder) vs git's staging area * git add README.md // Adds the new or newly modified file-name to git's staging area * git commit -m "Initial Commit" // Commits all files currently in git's staging area.  // The -m parameter allows for a commit message directly from the command line * clear // Clears all previous commands from the terminal screen -- just a bit of clean up * git commit -am "Adding some more items" // Directly commit newly modified tracked files * git add . // Add all new and newly modified files * git reset HEAD README.md // "Unstage" the specified file from Git's staging area * git checkout -- README.md // Back out any changes made to the specified file and replace it with the version last committed in git * git log // Git commit history * git log --oneline --graph --decorate --color // Much better compact view of git commit history * touch Demo.txt // Creating a new ‘Demo.txt’ file * git rm Demo.txt // Deleting ‘Demo.txt’ file * mkdir subfolder // Creating a new directory (folder) * mv Demo.txt subfolder/ // Move ‘Demo.txt’ to ‘subfolder’ directory (folder) * touch .gitignore // Create ‘.gitignore’ file * ssh-keygen -t rsa -C ["nadhu.uk@gmail.com"](mailto:\"nadhu.uk@gmail.com\") //Generating an SSH Key * tail id\_rsa.pub // Seeing inside ‘id\_rsa.pub’ file (SSH Key) * ssh -T git@github.com  // Verify SSH authentication and to connect to GitHub over the SSH protocol * git remote add origin git@github.com:scm-ninja/git-demo.git // SYNTAX: ‘git remote add remote-name remote-repository-location’ // Creating a remote repository reference * git remote -v // List out all git repositories * git push -u origin master SYNTAX: ‘git push -u origin master’ AND git push origin master SYNTAX: ‘git push origin master’ // Send Changes to Remote. The git push sends all your local changes (commits) on branch branch-name to the remote named remote-name. The ’-u’ parameter is needed the first time you push a branch to the remote. * git pull origin master SYNTAX: git pull remote-name branch-name // Receive Changes from Remote. The git pull receives all your remote changes (commits) from the remote named remote-name and on branch branch-name. * which git // Shows git location |