|  |
| --- |
| **GitHub Udemy Notes:** |
| * git --version  <<<OR>>> git version // show git version. * git help // get general help. * git help config // get configuration help. * PRESS ‘q’ OR ‘Q’  // exit to terminal. * 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 in terminal. // seeing ‘.gitconfig’ file in terminal. * vim ~/.gitconfig // edit ‘.gitconfig’ file in terminal. * pwd // shows current directory (folder) (present working directory (folder)). * cd workspace // move to ‘workspace’ folder. * ls // show files and folders in current directory (folder) without parameters, will list non-hidden folders and files. * ls -a  // show current filesin that directory (folder) including hidden files. * git init git-demo // initialize / create empty git repository. * cd workspace/github/ // change current directory (folder) to github directory (folder). * touch README.md  // create a new file name “README.md”. * vim README.md // Open README.md file in Terminal. PRESS ‘i’ to edit ‘README.md’ file. PRESS ‘esc’ to go back to terminal file / exit from terminal. ENTER ‘:wq’ to save ‘README.md’ file. * git status // changes / modifications in the current repository. // shows which files have been modified in the working directory (folder) and git's staging area. * git add README.md // adds the new or newly modified ’README.md’ file 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. * 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.md // creating a new ‘Demo.md’ file * git rm Demo.md // deleting ‘Demo.md’ 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 ["lonewolfnadhu@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 |

|  |
| --- |
| Setting up SSH Authentication with GitHub |
| * Open Terminal and Type following: * cd ~ // go to the main user directory / home directory * pwd // make sure you are in main directory * cd .ssh // make sure there are no .ssh file * mkdir .ssh * cd .ssh/ * pwd * ssh-keygen -t rsa -C [“lonewolfnadhu@gmail.com”](mailto:\“lonewolfnadhu@gmail.com\”) * ls -al  // show all files in .ssh folder. you will find ‘id\_rsa’ file and ‘id\_rsa.pub’ file. * vim id\_rsa.pub  <<<OR>>> cat id\_rsa.pub // open ‘id\_rsa.pub’ file and copy everything and paste in github website where show setup ssh key. * ssh -T [git@github.com](mailto:git@github.com) // connect to github over ssh protocol. |