GIT

**http://192.168.1.6/Appzillon/3.6.0/Studio**

**http:**[**//username@192.168.1.6**](mailto://prudhvi.kumar@192.168.1.6)**'**

**username@192.168.1.6: Appzillon/3.6.0/Studio**

**Setting-up the User Profile :**

**Open Terminal.**

* **Set a Git username: [ same as your login id & passwd ]**

=> **git config --global user.name "username"**

=> **git config --global user.email "username**[**@i-exceed.com**](mailto:prudhvi.kumar@i-exceed.com)**"**

**Cloning the Existing Project :**

=> **git clone http:**[**\\192.168.1.6\Appzillon\3.6.0.S1\studio**](smb://192.168.1.6/Appzillon/3.6.0.S1/studio)

**#asks for username and Pass for Authentication.**

**Inittializing Git :**

=> **git init**

**Creating New Branch**:

**=> git branch <branch name>**

**Checkout the branch :**

**=> git checkout <branch name eg: Master>**

**Steps to push a file to Remote Repo:**

1\* Edit the file

2\* add the file to staging

**=> git add <file name> or git add \*** [for multipple files]

3\* check the status of the file

**=> git status**

4\* commit the changes to the local repo :

**=> git commit <file name> -m “Commit message”**

**5**\* push the commited changes to the remote REPO :

**=> git push remote <branch name>**

By default it pushes to the origin[master branch]

**ignoring files :**

**craete a .gitignore file in the current working directory and save the file extensions to be ignored.And this will not track the files which has been specified in .gitgnore file [eg tmp, name.sh~] etc..**

**Check the previous Commits in to REPO:**

**=> git log**

**Last two commits => git log -2**

**=> git log -n**

**Track with Author name :**

**=> git log --author= <name>**

**=> git log --since= <date>**

**=> git log --until= <date>**

**Logs with oneline**

**=> git log --oneline**

**Each commit will have a SHA value looks like :**

**SHA value : b3br4rfuu4br8d0993be77ydwoib7ewo**

**Unstaging a file :**

**=> git reset<filename>**

**eg: git reset index.html**

**Checking the diff between two files :**

**=> git diff <file1> <file2>**

**eg: git diff index.html index1.html**

**Delete a file :**

**=> git rm <filename>**

**eg : git rm index.html**

**# it will delets file and stages the deletion**

**Renaming A file :**

**=> git mv <filename>**

**eg : git mv index1.html index.html**

**Stashing :**

**=> git stash**

**leave your work temporarly and can comeback to the same**

**=> git stash pop**

**RESTORES the recent stashed files**

**=> git stash list**

**Track a particular file :**

**=> git log --follow <file name>**

**Diff between Branches :**

**=> git diff <branch 1> <branch 2>**

**UNDO a commit:**

**=> git reset <commit>**

**=> git reset --hard <commit>**

**Pull the changes :**

**=> git pull URL**

**it pulls the changes and merge it with current work**

**Fetch the changes :**

**=> git fetch URL**

**it pulls the changes but won't merge it with current work**

**Merging the branch :**

**=> git merge <branchname>**

**eg: git merge Development**