GIT MY Work

1. Create repository on git with name : IntellipaatStudyMaterial and URL: <https://github.com/abhishekraics001/IntellipaatStudyMaterial.git>
2. Create one local folder with same name (IntellipaatStudyMaterial)
3. Initalize IntellipaatStudyMaterial folder in local system as git repogitory by typing **git init**
4. Create & Add some file into this folder (for creating file through command you can use **touch xyz.java**)
5. Check the file status inside local folder(whatever we have done the chnages & which file, added or chnages deleted, modify by typing **git status**

Add all newaly created file which are inside untracked area to staging area in local repository by typing **git add xyz.java** or **git add .**

1. Commit all change (which all are inside staging area) by typing to save it into local repogitory **git commit –m”your commit message”**
2. Note:
   1. Make sure to add authors details
      1. Add Author Name: **git config --global user.name “Abhishek Rai”**
      2. Add Author Email ID **git config --global user.email** [**abhishekraics001@gmail.com**](mailto:abhishekraics001@gmail.com)
   2. if you have already respogitory on git with old codebase then you 1st need to get clone of your repository at before the 1st step

**git clone https://github.com/abhishekraics001/IntellipaatStudyMaterial.git**

* 1. If you created newaly repo on git then you 1st need to run below command

**git remote add origin** [**https://github.com/abhishekraics001/IntellipaatStudyMaterial.git**](https://github.com/abhishekraics001/IntellipaatStudyMaterial.git)

* 1. First time it’s ask pass to push details on remote server
     1. Go github account details Profile => Setting=> Developer Setting => Parsonal Access Tocken => Generate new Tocken <https://github.com/settings/tokens?type=beta>
     2. Copy the tocken and put it as password

1. Push all local/staging area file/chnages into remote repository by typing **git push --set-upstrem origin master**
2. Get the update code from remote repositiry by typing **git pull**

10.Git the comit log by typing **git log or git log –oneline**

**11.** create new local branch in same resposity branch by typing **git branch DevTesting**

**12.** Get all branch list inside local or remote by typing **git branch**

**13.** How to check active/current branch**: It’s simply shoing \* with green color**

**14.** Swich from one brach to another branch by typing git branch **git switch DevTesting** or **git checkout branchName**

**15.** Add/Do The chnages, Commit and push your branch on remote

**Git add .**

**Git commit –m”** **Add New DevTesting Branch”**

**Git push –u origin DevTesting**

**17.** Get Checkout from remote respository **git checkout master**

**18**. Merge your DevTesting Branch into/to Master Branch by typing

**git merge DevTesting**

**git push**

**19.** If you want to change the message of last commit

**git commit --amend -m"Changing commit message: Add chnages for RestHead"**

**20.** Reset Your last or any **git reset HEAD~1**

**21.** Clear/clean the local **caches git clean –n**

**22.** Delete branch **git delete –D Dev1(your branch name)**

**23.** Delete file **git rm ABC.text(filename)**

**24.** Tag in Git: create tag by typing **git tag rlease-1**

**25.** push the tag into remote repository **git push --tag**

**26**

**22. In case of Conflict fllow bellow step to get it resolved:**

**How conflict happen:** Supposes you have one respository and now inside **master** branch & there is one file like **ABC.text**. After that you just create two more branches from master **Dev1** and **Dev2. 1st** you checkout **Dev1** branch and do some chnages inside the **Abc.text** file & Add + Commit it inside the same(Dev1) branch and after that you checkout **Dev2** branch alos do some chnages insdie **Abc.text** file & Add + Commit it inside the same(Dev2) branch. After that we checkout the **master** branch and marge Dev1 branch with master(will done successfulyy) but when we try to mager Dev2 branch with master then conflict will accoure

touch A.text

git a .

git add .

git commit -m"Adding A.text file into master branch"

git branch Dev1

git branch Dev2

git branch

git checkout Dev1

nano A.text

git add .

git commit -m"Modifinig A.text file into Dev1 Branch"

git checkout Dev2

nano A.text

git add .

git commit -m"ModifingA.text file into Branch Dev2"

checkout master

git checkout master

git merge Dev1

git merge Dev2

git mergetool --tool=vimdiff

git commit -m"Resolving conflick and masger into Master"

**How to resolved**

* **git mergetool --tool=vimdiff** => Editor will open then => For view the conflict file content
* Review the connect whare conflick is happed and edit with atual onc and type below command multiple time(4/5 time)
* **Esc** => press Escape keyword button
* **:wq** => for save & quit
* Then type the commit command to done the changes **git commit –m”Conflick resoluton done”**