<https://github.com/dvk040/myWorkspace>

**Working at client location infosys ITD 2.0**

**Git stash-----------------🡪 u changed but not finished But u have to pull code at that time use --🡪git stash save “give message”**

**Next ------🡪 git pull origin master**

**Git stash list----------🡪 to see stash list**

**Git stash show –p id-Number**

**Git stash pop will take recent stash**

**Git stash pop <stash-id>**

**After pop their will be a conflicts check it**

**Git stash drop---🡪latest stash will be drop**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Git diff ----------------🡪 will show what are the differences**

**Git push origin branch1:branch2-----------🡪 branch2 is maintained but not shown**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Git branch –d branchName-----------🡪 for delete branch**

**Git branch –merged it will give which branches are merged**

**Git branch --no–merged it will give which branches are Not merged**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**To skip staging -----------------🡪 which means “git add” is not doing-----🡪 directly git commit**

**Git commit –a –m “write any comment”**

**After conflicts resolved**

**You have un merge paths---------🡪enter command ------🡪 git add .--------🡪 git commit**

**Branch to master branch merge**

**Git merge branchName**

**You have un merge paths---------🡪enter command ------🡪 git add .--------🡪 git commit**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**SVN**. Also known as **Subversion**, **SVN** represents the most popular centralized version control system on the market. With a centralized system, all files and historical data are stored on a central server. And **developers commit their changes directly to that central server repository**

**Git** is the most commonly used version control system. **Git** tracks the changes you make to files, so you have a record of what has been done, and you can revert to specific versions should you ever need to. **Git** also makes collaboration easier, allowing changes by multiple people to all be merged into one source.

**Git branch**

**To create a branch**

**Git checkout –b branchName**

-b means new branch X **Git branch –d branchName-----------🡪 for delete branch**

$ touch .gitignore

# It will create a gitignore file

# Git gc –auto ---------🡪 for un link files

# Git checkout –f -----🡪 all changes will comeback in files for last commit

# Git checkout filename

# If lost data to retrive

# File to make unstage

# Git restore filename----🡪 un modified to last step

git restore --stage secoundOne.docx

# after add we can make unstage

# Git init its opposite is-------🡪 rm –rf .git folder converted into normal

# Git init X rm –rf .git

# to start git in that folder

# [Ignore modified (but not committed) files in git?](https://stackoverflow.com/questions/655243/ignore-modified-but-not-committed-files-in-git)

git update-index --assume-unchanged dir-im-removing/

git update-index --assume-unchanged config/database.yml

git check-ignore [<options>] <pathname>…​

git check-ignore [<options>] --stdin

I changed observe

I what to check about stat

To know Email

<https://git-scm.com/docs/git-log>

git log --pretty=format:”%h -- %an”

*%h*

abbreviated commit hash

*%an*

author name

Placeholders that expand to information extracted from the commit:

*%H*

commit hash

*%h*

abbreviated commit hash

*%T*

tree hash

*%t*

abbreviated tree hash

*%P*

parent hashes

*%p*

abbreviated parent hashes

*%an*

author name