GIT REFERENCE NOTES
By Devayush Bajaj

A Git repository tracks and saves the history of all changes made to the files in a Git project. It saves this data in a directory called . git , also known as the repository folder

GIT Repository Types

1) Local Repository - Git local repository is the one on which we will make local changes, typically this local repository is on our computer.

2) Remote Repository - Git remote repository is the one of the server, typically a machine situated at 42 miles away.

GIT BASH GENERAL QUERIES (For Local Repository)

Step wise queries:-

S1) add global user email :-
 git config --global user.email devayushbajaj@gmail.com

S2) add global username:-
git config --global user.name "devayush-bajaj-coditas"

S3) To verify user :-
 git config --list

S4) To make a new folder

mkdir Folder_Name

S5) To enter in that folder

cd Folder_Name

S6) To make that folder repository

git init

S7) To create a file using GitBash

touch FileName.txt

S8) To open that file:-

notepad FileName.txt

S9) Add message or update as per work performed using commit:-

git commit -m "This is my first commit and I have updated this repo"

```
coditas@DBajaj-LAP-0765 MINGW64 ~/Git_Demo (master)
$ git add .

coditas@DBajaj-LAP-0765 MINGW64 ~/Git_Demo (master)
$ git commit -m "This is my first commit and I have updated this repo"
[master 0065b54] This is my first commit and I have updated this repo
 2 files changed, 3 insertions(+), 1 deletion(-)
```

S10) To get the status of files in the repository (Which file is added, updated and deleted):-

git status

```
coditas@DBajaj-LAP-0765 MINGW64 ~/Git_Demo (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        master.txt

nothing added to commit but untracked files present (use "git add" to trac
```
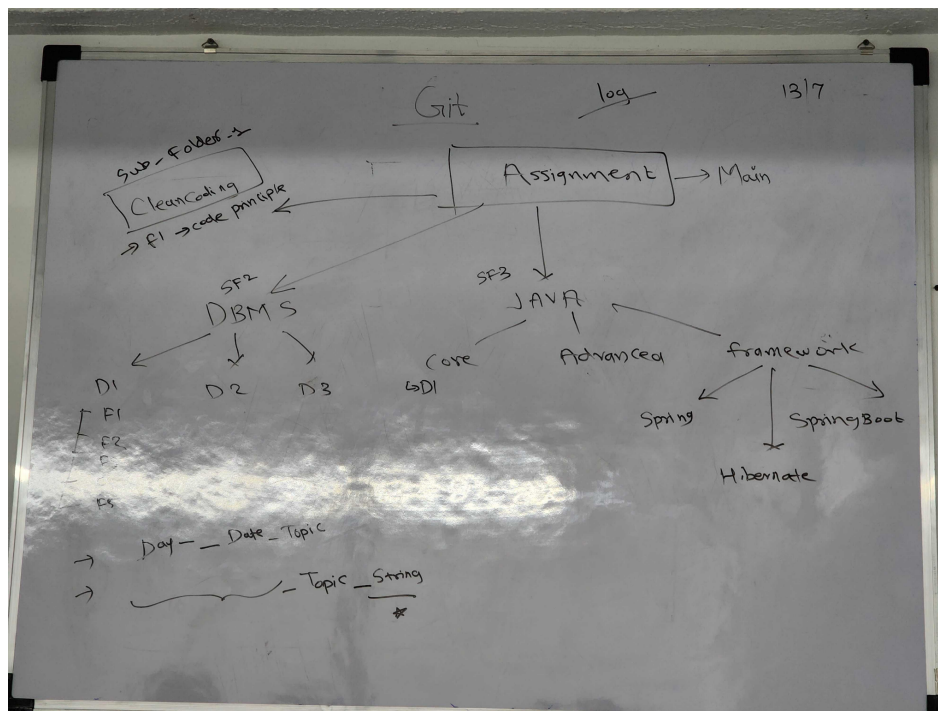
Q1 create folder structure

As per the above folder flow we need to create folders in our local repository

```
coditas@DBajaj-LAP-0765 MINGW64 ~ (master)
$ cd assignments

coditas@DBajaj-LAP-0765 MINGW64 ~/assignments (master)
$ mkdir JAVA
m
coditas@DBajaj-LAP-0765 MINGW64 ~/assignments (master)
$ mkdir GIT

coditas@DBajaj-LAP-0765 MINGW64 ~/assignments (master)
$ mkdir DBMS

coditas@DBajaj-LAP-0765 MINGW64 ~/assignments (master)
$ mkdir frameworks

coditas@DBajaj-LAP-0765 MINGW64 ~/assignments (master)
$ cd java

coditas@DBajaj-LAP-0765 MINGW64 ~/assignments/java (master)
$ touch .gitkeep

coditas@DBajaj-LAP-0765 MINGW64 ~/assignments/java (master)
$ cd..
```

Queries used to create the structure in local repository

mkdir FolderName —> create empty folder

cd FolderName —> Navigate to other folder present in that directory.

touch .gitkeep —> used when there is empty folder but still need to be uploaded in remote repository

cd .. —> used when we need to go back to the previous folder (as per directory)

Q2 how to add an existing folder into the github repository (using GitBash).

NOTE - Before adding a folder we need to make sure that it is not empty. If it is then use .gitkeep

Step wise explanation:-

S1 - First make the repository (Name=RepositoryName) on github.
S2 - Open the terminal and make the new directory (mkdir NewDirectory).
S3 - Copy your ProjectFolder to this NewDirectory.
S4 - Change the present work directory to NewDirectory.

S-5 Run these commands

git init   Making folder master(Repo)

git add ProjectFolderName   Adding files(We can use . to add all files)

git commit -m "first commit"   Committing

git remote add origin https://github.com/YourGithubUsername/RepositoryName.git

git push -u origin master   Pushing the files into the remote repository

<u>GROUP ACTIVITY</u>

Lets-have-fun-git

In this activity, we have understood how conflict problems occured.

<u>Task: -</u> There was an shared repository between all the team and there were two files

1) Name.txt   2)Character_Name.txt

We need to type the name in the first file and any fictional character in the character file.

In the very first round, each team member is allocated one time to insert his data and push these files to the remote repository. After that all the remaining 4 rounds everyone has to update without any sequence randomly.

Observation:-

We have observed that in the end while tely the data. Few data seems to be mismatched. Means one name is assigned to the character of another name.

This explains the problem of conflict when multiple users are working on a particular file.

STEPS TO FOLLOW

1) Cone the data from the github repository
        git clone Repo_Link

2) Edit the file and fill the details
3) git add .
4) git status'
5) Git commit -m"msg"
6) Git push origin master