**Academic Year: 2023-24 Semester: V**

**Class / Branch: TEIT Subject: DevOps Lab**

**Name of Instructor: Prof. Sonal Jain/Prof. Neha Deshmukh**

# Experiment No. 3

**Aim: To understand and perform version control system / source code management using Git.**

GIT is a Version Control System (VCS) (aka Revision Control System (RCS), Source Code Manager (SCM)). A VCS serves as a Repository (or repo) of program codes, including all the historical revisions. It records changes to files at so-called commits in a log so that you can recall any file at any commit point.

To issue a command, start a "Terminal" (for Ubuntu/Mac) or "Git Bash" (for Windows):

$ **git <command> <arguments>**

The commonly-used commands are:

1. **init**, **clone**, **config**: for starting a Git-managed project.
2. **add**, **mv**, **rm**: for staging file changes.
3. **commit**, **rebase**, **reset**, **tag**:
4. **status**, **log**, **diff**, **grep**, **show**: show status
5. **checkout**, **branch**, **merge**, **push**, **fetch**, **pull**

#### **Getting Started with Local Repo**

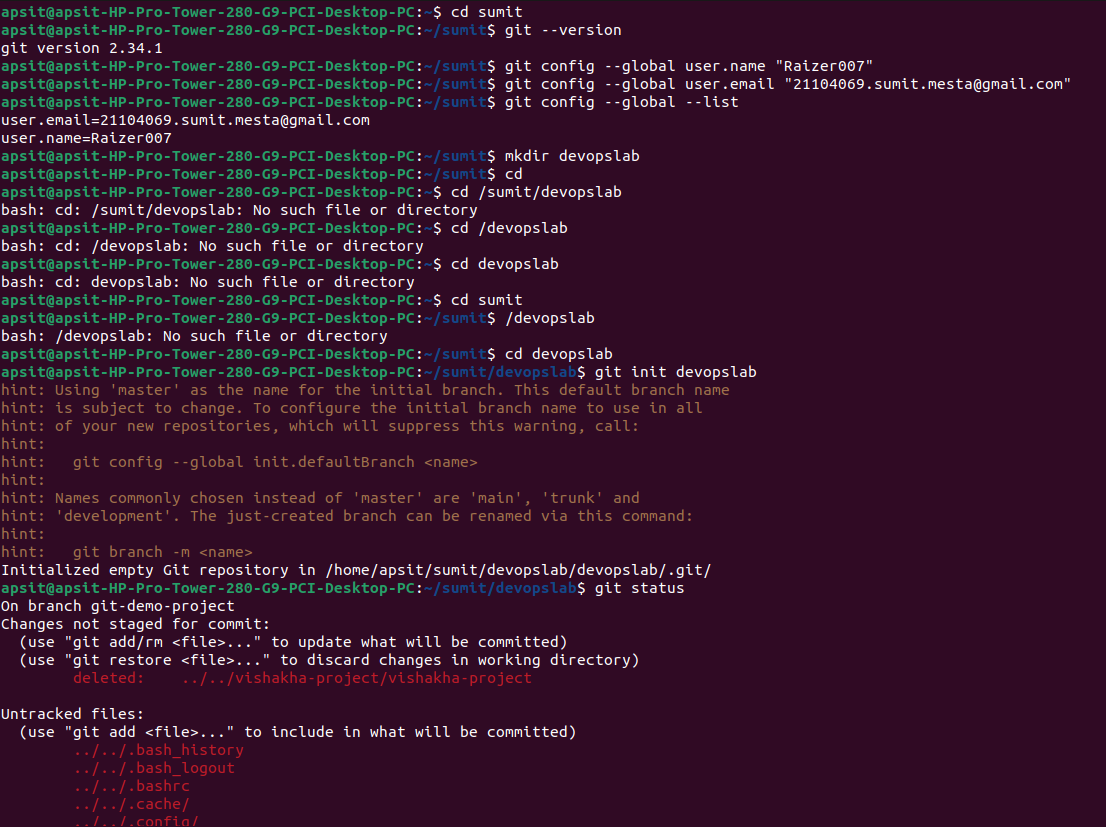
There are 2 ways to start a Git-managed project:

1. Starting your own project;
2. Cloning an existing project from a GIT host.

Git uses two stages to commit file changes:

1. "git add <file>" to stage file changes into the staging area, and
2. "git commit" to commit ALL the file changes in the staging area to the local repo.

**Prerequisite: Commands of Exp 2**



**You need to setup Git on your local machine, as follows:**

**Step1:** Download & Install:

* For Ubuntu, issue command "sudo apt-get install git".

**Step 2.**

* Create a repository on local machine

mkdir Sumit

* Initiate that directory to make it a git repository (.git file must be added inside that folder after initiation)

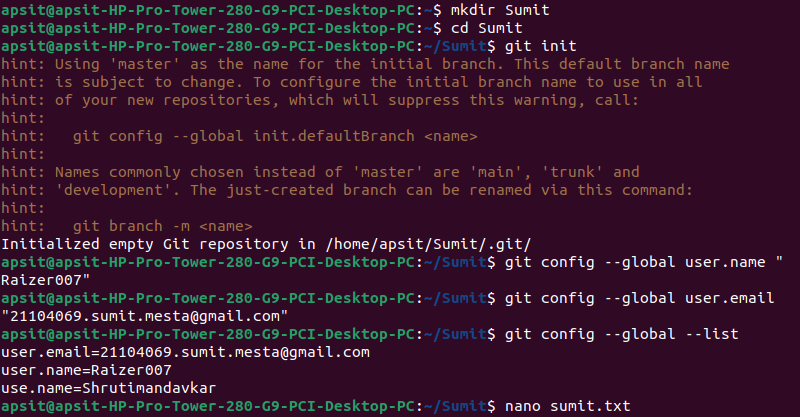
Git init Sumit

**Step 3**: Customize and configure your Git Account:

// Set up your username and email (to be used in labeling your commits)

$ **git config --global user.name "*your-name*"**

$ **git config --global user.email "*your-email@youremail.com*"**



You can issue "git config --list" to list the settings:

$ **git config --list**

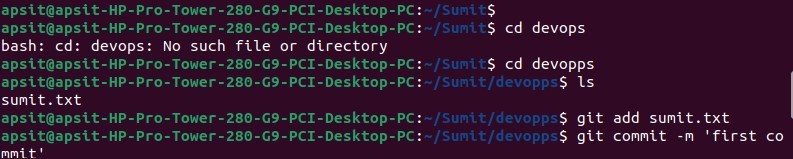
user.email=xxxxxx@xxxxxx.com

user.name=xxxxxx

**Step 4:** Create and Add one file in that folder and write some content inside it

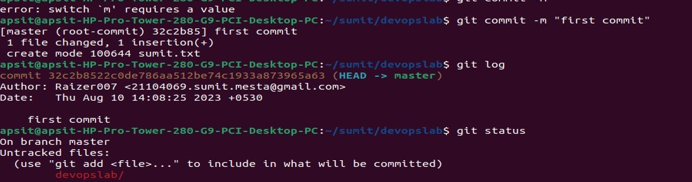
gedit sumit.txt

git add sumit.txt

****

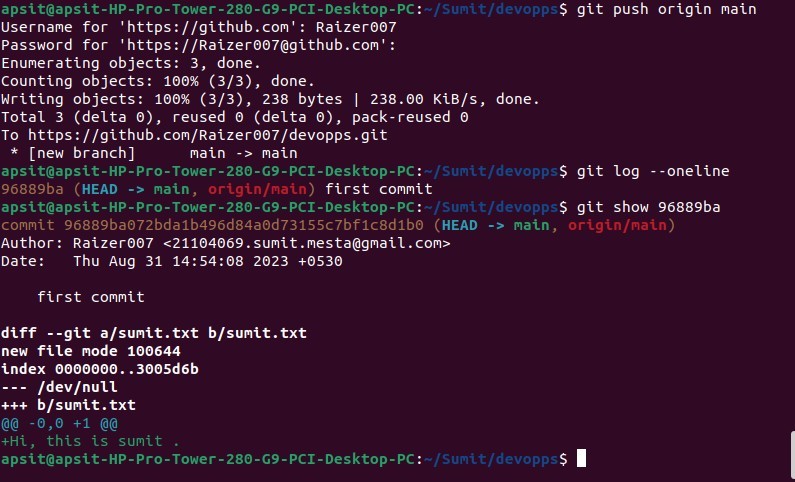
**Step 5**: To start first commit

git commit -m “my first commit”



**Step 6: To push local repository on git account**

**git push origin master or git push origin main**



**Step 7:** Make some changes in the previous file like “sumit.txt” and then add that file again

gedit sumit.txt

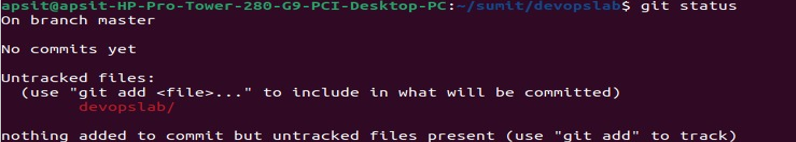
git add sumit.txt

Step 8: Do second commit after making changes in the file as follows

git commit -m “my second commit”

Step 9: Check Git Status after after step to see the staging of git repository

git status



**Step 8:** To see the logs in oneline like username, email -id, date, time of creation.

git log

or

git log –oneline

**Step 9:** To show repository id and other detail

git show

**Step 10:** To see the difference in the content of file between first and second commit.

git diff sumit.txt

**Step 11:** Creating a latter commit and reverting back to see the initial/original content

git commit <secondcommitID>



# Conclusion:

# In this experiment, we understood the use case of Version Control System, its benefits in real time scenario which provides a application of reverting the changes when people are in working in a collaborating environment. Different commands were used for the same such as revert(by using its id),diff for displaying the changes between the initial and latter texts.