GIT AND GITHUB

\*Git is the most widely used control system

\*from college students to large companies to open source project,spanning hundreds of developers all over the world.

\*Git has all basic feature of version control system,inorder to keep track of source course development.

\*Distributed.

\*Includes useful constructs Branching & merging.

\*Github is an ecosystem which makes it very simple to setup a source code & collobirate with developers who use Git.

INTRODUCING THE GIT CLIENT:

--git –version

\*git config –list

\*git config --global user.name "poorvi"

\*git config --global user.email [poorvi@gmail.com](mailto:poorvi@gmail.com)

\* git config –list

\* git config user.name

\* git config user.email

CREATING A LOCAL GIT REPOSITORY:

Create a project directory

\* mkdir company\_web

\* cd company\_web

LOCAL REPOSITORY

\* git init

Initialized empty Git repository in C:/Users/DELL/company\_web/.git/

\* cd .git

\*ls

\* nano file.txt

\* echo "Brocadero webpage">>README.md

\* ls

DIFFERENCES BETWEEN THE CURRENT PROJJECT AND LOCAL GIT REPOSIT SINCE LAST COMMIT

\* git status

\* git add . TO KEEP IT TRACK

COMMIT THE SAVED FILES

\* git commit -m "this is my first commit"

\* git status

CREATING A REPOSITORY ON GITHUB:

REMOTE REPO THROUGH WHICH THAT LOCAL REPO WILL BE MAPPED

LINKING A LOCAL REPO TO REMOTE REPO:

\* git remote -v

\* git remote add origin [https://github.com/poorvih01/web--page-repo-->LInk](https://github.com/poorvih01/web--page-repo--%3eLInk)

\* git remote -v

SSH KEY GENERATION

Id\_rsa

123

\* cat id\_rsa.pub

TEST CONNECTION TO GITHUB

\* ssh -T [git@github.com](mailto:git@github.com)

03/04/2025

PUSHING FROM LOCAL REPOSITORY TO REMOTE

\*git push origin master

ADDING FILES TO REPO

\*git status .

\*git add

\*git status

\* git commit -m "Added web page"

MODIFIYING FILES IN REPO

Create one html file in vs in same direcortory

\* git status

\*git add .

\* git status

\* git status –short

\*git diff –staged

\* git commit -a -m "Added about html"

\* git status

VIEWING THE COMMIT LOG

\* git push origin master

\*GIVES US DETAILS OF ALL COMMITS WHICH HAVE BEEN RECORDED- git log

\* git log -1—recent commit

\* git log --oneline

\* git log –stat

\* git log –patch

VISULAIZING GIT BRANCHES

Go to visualizing git website

\* git checkout -b new\_branch

\* git checkout maste

\* git commit

\* it checkout new\_branch

DELETING FILES FROM REPO

\* ls

\* git rm file.txt

\*ls

\*delete the working repo but not from the working tree-- git rm --cached README.md

\* git status

\*rename-- git mv company\_web.html about1.html

\*ls

\* git checkout -b new\_branch

\* git status

\* git commit -m "Rename company\_web and remove"

\* git status

\* git checkout master

\* git add .

\* git stash

\* git status

\* git stash list

\* git stash show

VISULAIZING A GIT MERGE:

\* git status

\* git add history.html

\* git commit -m "added history page"

\* ls

\*git checkout master

\* git merge new\_branch

\* nano about1.html

\* git status

\* git add about1.html

\* git commit -m "modidfied about"

IMPLEMNETING A MERGE OF TWO BRANCHES:

\* git checkout master

\* git merge new\_branch

\* git status

\* git log --oneline

\* git push origin master

USING GITHUB FOR SOURCE CODE MANAGEMENT