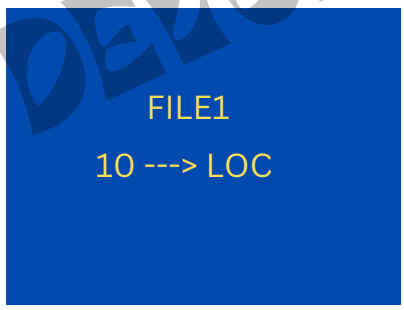
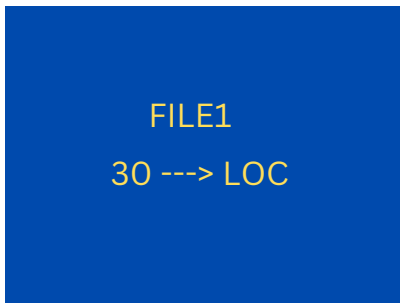
GIT: GLOBAL INFORMATION TRACKER

WHY GIT?

- IT IS USED FOR SOURCE CODE MANAGEMENT.
- GIT IS USED TO MAINTAIN MULTIPLE VERSIONS OF THE SAME FILE.



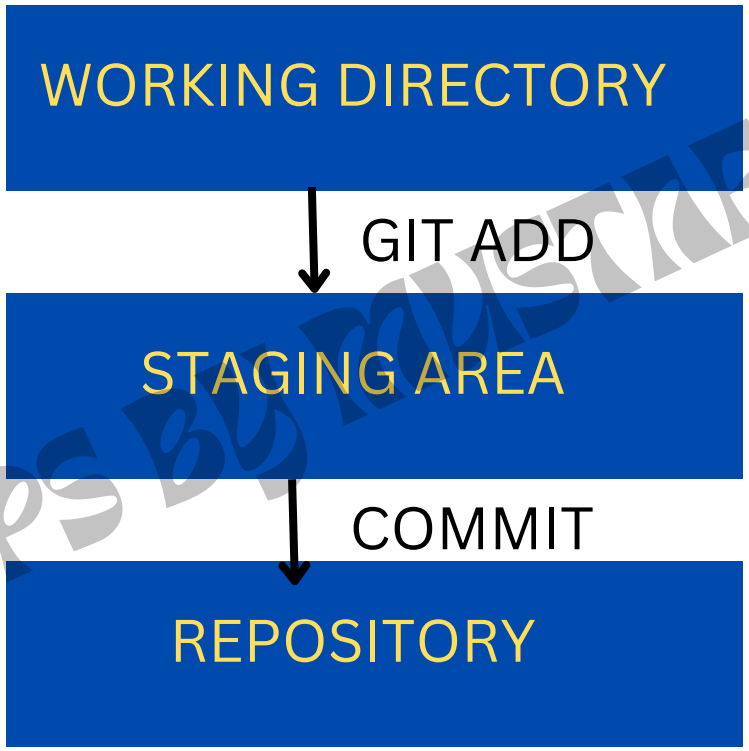


- GIT IS FREE AND OPEN SOURCE.
- GIT IS PLATFORM INDEPENDENT.

IT SUPPORTS LINUX, MAC OS, AND WINDOWS.

- GIT IS USED TO TRACK THE FILE.
- GIT WAS INTRODUCED IN THE YEAR OF 2005.

GIT STAGES:



DEVOPS BY MUSTAFA

WORKING DIRECTORY:

- In this stage git is only aware of having files in the project.
- It will not track these files until we commit those files.

STAGING AREA:

- The staging area is like a rough draft space, it's where you can git add the version of a file or multiple files that you want to save in your next commit.
- In other words, in the next version of your project.

REPOSITORY:

- Repository in Git is considered as your project folder.
- A repository has all the project-related data.
- It contains the collection of the files and also history of changes made to those files.

We have 3 repositories in GIT

- LOCAL REPO
- REMOTE REPO
- CENTRAL REPO

DEVOPS BY MUSTAFA
DEVOPS BY MUSTAFA

LOCAL REPO:

The Local Repository is everything in your .git directory. Mainly what you will see in your Local Repository are all of your checkpoints or commits. It is the area that saves everything (so don't delete it).

REMOTE REPO:

The remote repository is a Git repository that is stored on some remote computer.

CENTRAL REPO:

This will be present in our GITHUB

INSTALL GIT:

yum install git -y

yum: yellowdog updater modifier

To check the git version:

git --version

git init . : to get empty repo

To track the file:

git add file_name : single file

git add aws azure gcp: multiple files

git add * : all regular files

git add . : including hidden files

DEVOPS BY MUSTAFA

GIT INSTALL: yum install git -y
TO INITILIZE AN EMPTY REPO: git init .

here .(dot) represents the current directory (~). so it will creates .git folder (hidden)

to check the version: git --version
it returns: git version 2.38.1

GIT is used to track the file
to track a file

1. touch file_name : create a new file
2. git add file_name : add git to file
3. to check : git status

green color file represents tracking files
red color files represents untracking file

tracking means: we have a record that what we made the changes on a file.
it will track the each and every change in a file.

to commit a file: git commit -m "message" file_name

once we commit the file, it will goes to repository.
we can see the history of a file by using git log

DEVOPS BY MUSTAFA
DEVOPS BY MUSTAFA

to track multiple file: git add f1 f2 f3
to track all files: git add *
to track all files including hidden files: git add .

to commit multiple files: git commit -m "msg" f1 f2
to commit all tracking files: git commit -m "msg" .

.(dot) represents the all tracking files
notes: we can't commit the untracking files.

to untack a file: git rm --cached file_name
once we untrack a file, that files goes to working directory.

DEVOPS BY MUSTAFA
DEVOPS BY MUSTAFA

to untrack the files:

git rm --cached file_name

to commit file:

git commit -m "message" file_name

git commit -m "message" aws azure gcp

git commit -m "message" .

to check the history

git log

git log:

git log -2: latest 2 commit

git log --oneline: used to see only commit id's and commit messages

git config: it is used to configure the user with their mail id in git

command: **git config user.name "user_name"**

git config user.email "user@email.com"

git restore: used to get back the deleted files

command: **git restore file-name**

git show: it is used to show the file names in log

command: git show commit_id --name-only

DEVOPS BY MUSTAFA