Devops 🡪

It is used to develop a software lifecycle from build to deploy.

IN SDLC we have different approaches like waterfall , agile etc.

1. Waterfall -🡪 it is oldest practice.

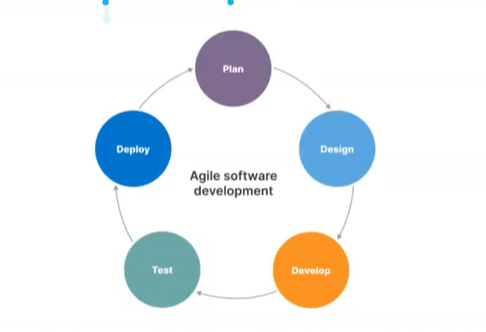
A diagram of a software development process

AI-generated content may be incorrect.

Like we are going to build a website for hotel. We are going to follow above approach.

Here next phase will not be able to go till previous got finished. Also u cannot go back to any step once u got forward.

Agile model🡪



Here we can go back to any phase.

Now come to devop’s 🡪

A list of software components

AI-generated content may be incorrect.

A computer screen shot of a diagram

AI-generated content may be incorrect.

Containerization is a software deployment process that packages an application's code with all its dependencies into a single container. This allows the application to run on any operating system or infrastructure.

Version Control System 🡪

It is a system which will track or record the changes on the file over the period of times.

Two types🡪

1. Centralized 🡪 Rarely used by defense etc.
2. Distributed 🡪 93 % ruling it.

Centralized version control systems (CVCS) rely on a central server to store project files, while distributed version control systems (DVCS) allow each developer to have their own copy of the project files.

GIT 🡪

Let learn git but we are not going to use in our system. We are going to take help of aws.

launch one instance on aws.

Now a days you will automatically find git if not please install.

Sudo yum/apt install git -y (linux/ubuntu)

Use git --version to check version for git. Or else use git bash in windows.

Now first step for git is git init.

Git init 🡪 It initialized a empty git repository.

Now in that folder us ls -a (to see hidden folders.)

A screenshot of a computer code

AI-generated content may be incorrect.

We can see a .git folder got created. This is the empty repository which got created.

This folder will track all the commands which we are going to use.

Now let’s understand the GIT lifecycle.

A diagram of a diagram

AI-generated content may be incorrect.

When we are working on a file it comes under working directory.

Now when we add the file using git add or git add . it comes under staging area.

Now when we use git commit it will move from staging to local repository and from here we can track the changes . who has done the changes what changes he had push.

Now what is the use of that Staging Area, why we cannot direct commit ?

It is like a validation Area . it will show all the files which u want to commit .

If u need to revert some files we can take advantage of this area.

Now let’s do some commands to see demo.

First set username and email to track the commits.

Use below commands to set the username and password.

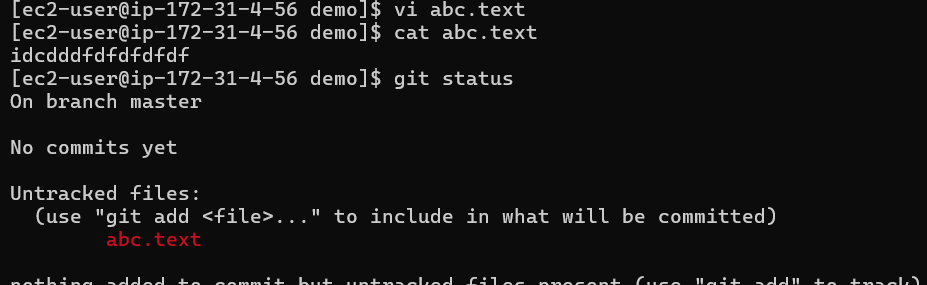
git config –global user.name ‘Manu’

git config –global user.email ‘Manu.raghava@gmail.com’

Now create any file using command vi.

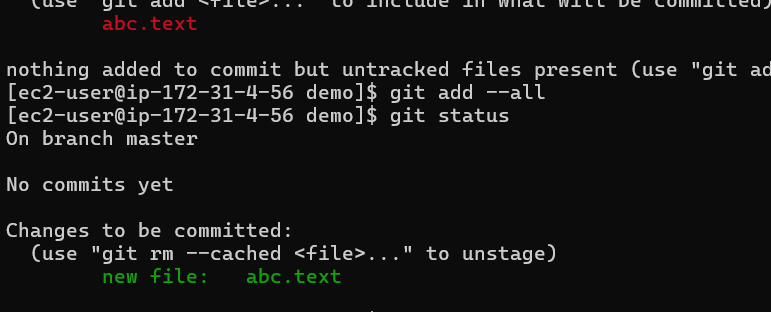
vi abc.text

made some changes in it and then run git status. Git status is the command we use to see our files in working directory or staging area.



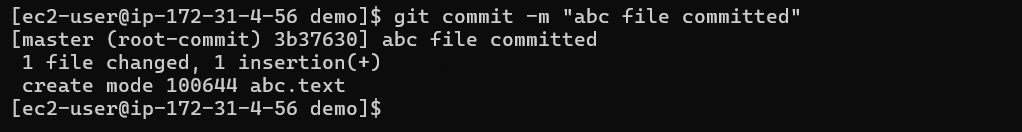
Here we can see our file in red color . It means it is in working directory.

Now run git add abc.text or git add . or git add --all and then run git status.



Now we can see color changes to green means it moved to staging area.

Now we will move our file from staging area to local repository by committing it.



Now we will use the command git log to see all the commits made on this path ever.

A screen shot of a computer

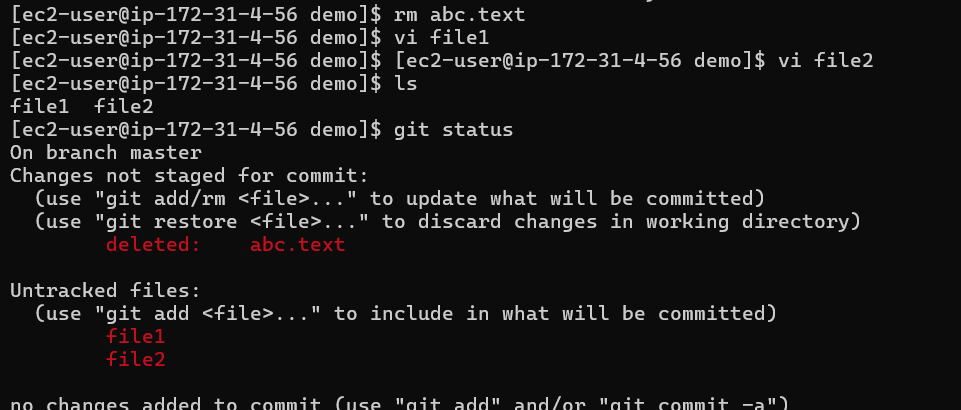
AI-generated content may be incorrect.Here we can see who has made the commit . every time commit id is created.

Git Reflog vs Git Log

The git log is a public record of the commit history of the repository, whereas the reflog is private. This is the key difference between git log and reflog. After a fetch, push, or pull, the git log is duplicated as part of the git repository. In that situation, the git reflog is absent.

Now how to move back files from staging to working directory 🡪

First create two files file1.txt and file2.txt. git add . 🡪 will move these files to staging.



Here we have removed our old file and add two files file1 and file2.

On doing git status we can see all files in red(deleted or modified[in case of update]).

Now run git add .

A screen shot of a computer

AI-generated content may be incorrect.

Now we can see all the files moved to green after using add --all . to revert back the file from staging area we can see command above only which is mentioned by git itself.

git restore --staged file name or we can use git reset.

A screen shot of a computer

AI-generated content may be incorrect.

We can see all the files move back to working directory after reset.

Now we try to connect our local repository with remote one.

We have already done git init in our demo folder.

First go to GitHub account and create one repository.

After creating the repo go into our remote repository and run this command.

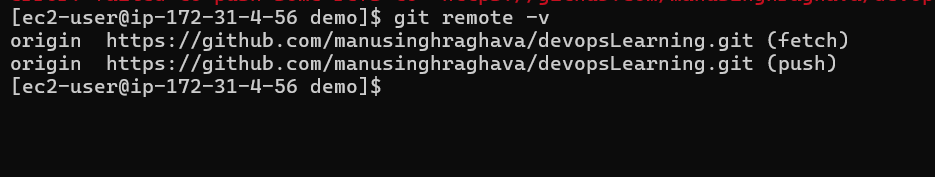
git remote add origin https://github.com/manusinghraghava/devopsLearning.git

git push -u origin main

devopsLearning here is our repo and origin here is alias for whole url .

so we don’t need to use whole URL every time.

We can use git remote -v to check remote location.



Before pushing just run one command.

git branch -m main /// it will create one main branch where we can push.

After that only we should run.

git push -u origin main

Now when we try to push the code it will ask for username and password.

Password here is token which we need to create in GitHub repo settings 🡪developer options.

A screenshot of a computer

AI-generated content may be incorrect.

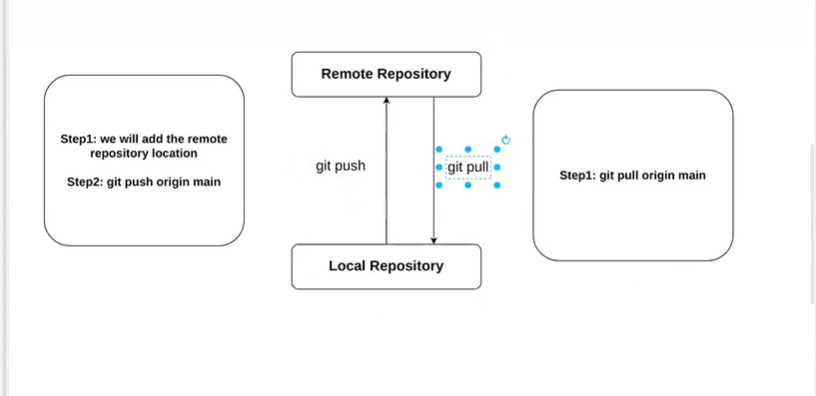
And generate new token save it and use every time to connect. Or we can use ssh keys.

After doing all these steps and pushing the data. We can see all on remote.

A screenshot of a computer

AI-generated content may be incorrect.

We use push/pull to exchange data between remote/local.



Now if we want to give repo to another developer to make changes on it.

How will they get it to push and pull? // they will use git clone.

If we want to delete local repo.

Delete the local git repo using the rm -rf .git // we can use it to revert 1st commit.

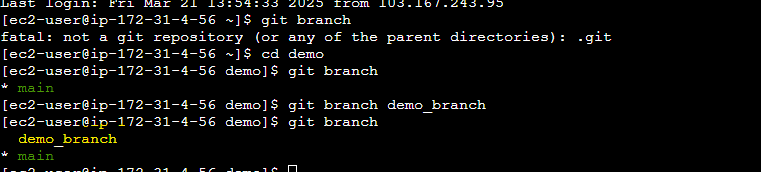
Branching🡪

Branch 🡪 whenever we create a repo and do a first commit , automatically a branch is created some places it is known as master and some known as main.

Git branch is the command we will use to see all branches in a repository.

Now if we use the same command with a name. It will create a new branch.

Git branch demo\_branch [new branch name]



We can see our new branch now.

Now to switch in the new branch we will use checkout command.

Git checkout test // to switch in the new branch.

Git checkout -b new\_branch\_name // it will create a new branch and switch to it.

A black screen with white text

AI-generated content may be incorrect.

<https://drive.google.com/drive/folders/1uXzPibrR-KpD4hAm1EVTBAyRm2sJwCZC>

all useful commands u can find here.