DevOps

**What is DevOps ?**

DevOps is the union of people, process and products to enable continuous delivery of values to our end user.

You cannot buy DeVops and install it. DevOps is not a automation or infrastructure as a code. Devops is people following a process enabled by products to deliver values to end user.

**Core componets of devops:**

People –

* Collaborate more
* Share common goles
* Focus on improment
* Bringing people together

Process –

* Eliminate waste
* Increase efficiency
* Stream line feedback
* Delivering value faster

Tools –

* Enhance productivity
* Enable collaboration
* Facilitate experimentation
* Execuating a devops strategy

**Fundamental principles of DevOps:**

* It is a culture of SDLC model.
* A new way in delivering of software projects.
* Automation using tools.
* Faster changes with stability.
* Mesure of success is, that we do more for our end users and for that we need the feedback and continuous feedback from those users, so that we can deliver value fast and continuously.

**Why DevOps use tools for Automation: (\*\*\*)**

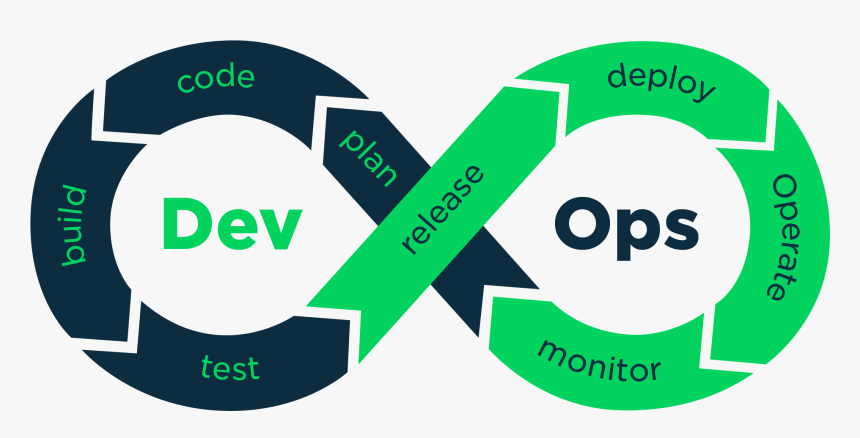
* The tools provide automation.
* The automation eliminates mistakes or human errors.
* It eliminates waste.
* It eliminate rework.
* It allows us to measure the feedback, so that we can enhance productivity.
* We can improve the collaboration experimentation.
* We can test whether the things we belive in advance.

**Defination of DevOps:**

DevOps is the practice of Operations, Development engineers, all other stakeholders participating together in the entire service lifecycle from design, development process to production support.

Simple Dev+ops = worktogether DevOps

“from code to product”



**Phases of DevOps:**

1. Plan
2. Code
3. Build
4. Test
5. Release
6. Deploy
7. Operate
8. Monitor
9. Feedback (new phase industry added)

**plan:-**

When client comes to us first we have to understand the client idea.

There are two activities in plan

1. Startegic Analysis
2. Capacity Analysis

Startegic Analysis:- Analyging weather, we can do that project or not.

Capaciy Analysis:- what I have and what I have to get.

**Code:-**

* Coding can be done by developer according to the plan.
* In devops python is the demanding language.

**Build:-**

Bulild is an activity which involves multipul things. right from compilation, logical testing,converting packages and hosting.

* Build process are happening queckly now.
* We are able to cutdown the time.
* We are able to eliminate mallparactice.

**Test:-**

* Here testing engineer test the code and identify the bug. Not only identifing the bug also fixed by the tester if the bug is in minor level.
* In devops automation testing is the citeria.
* Using tools such as Selenium, Junit etc.

**Release:-**

At this point, the build is prepared to be deployed in the operational environment. The DevOps department prepares updates or sends several versions to production when the build satisfies all checks based on the organizational demands.

**Deploy:-**

At this point, Infrastructure-as-Code assists in creating the operational infrastructure and subsequently publishes the build using various DevOps lifecycle tools.

**Operate:-**

The version is now convenient for users to utilize. With tools including chef, the management department take care of server configuration and deployment at this point.

**Monitor:-**

DevOps monitoring allows teams to respond to any degradation in the customer experience, quickly and automatically. More importantly, it allows teams to “shift left” to earlier stages in development and minimize broken production changes. An example is better instrumentation of software to detect and respond to errors, both manually through on-call and also automatically whenever possible.

**Continuous integration:-**

CI is all about integrating live changes of code into the master branch or release branch or product branch after being validated and tested.

Metric 1:- The changes we are pushing should be small.

Metric 2:- Frequency of deploying those changes should be high.

Metric 3:- The changes should be version controled and capabul enough to roll back if release fail.

**Lab setup:-**

**What is Oracle VMBox ?**

Oracle VM VirtualBox is a cross platform virtulation software. It allows users to extend their existing computer to run multiple operiting systems including microsoft windows, Mac OS X, linux and oracle Solars at the same time.

Step 1:- Download and install oracle VMBox.

*Download link –* [*https://www.virtualbox.org*](https://www.virtualbox.org)

* Click on windows host and it will download.
* After download right click on VMBox and click on open or run as adminstractor.
* At some point of installation, we may get a pop-up, that says do you want to disconnect the network and configure “select yes”.
* Challenges, if getting errors like download microsoft c++ or python just downlad it.

Step 2:- Image congiguration.

* Open oracle VMBox.
* Click on new.

Give, Name: ex:- centos base image

Type: Linuex

Version: Red Hat (64-bit)

Click on next

* Memory size – the recommanded size is 1024, click on next.
* Virtual Hard disk – select “Create a virtual hard disk now”

Hard disk size – 8Gb

Click next

* Click on finesh.

**What is CentOS 7 ?**

CentOS7 is an open source operiting system based on the Red Hat Enterprise linux source code and built on the Linux kernal.

**Centos-7:-**

Now download centos-7.

* Go to google.com and search for centos 7 download.
* Open offical website [*https://www.centos.org*](https://www.centos.org)*.*
* Select centOS Linux.
* Under centos linux click on “x86\_64”.
* After open the link *–*

[*http://centos.excellmedia.net/7.9.2009/isos/x86\_64/*](http://centos.excellmedia.net/7.9.2009/isos/x86_64/)

* Click on this link – [CentOS-7-x86\_64-Minimal-2009.iso](http://centos.excellmedia.net/7.9.2009/isos/x86_64/CentOS-7-x86_64-Minimal-2009.iso)
* It will download.

Step 3:- Mapping

* Open oracle VMBox.
* Select, created image.
* Click on settings --> click on storage
* In storage we can see left hand side “storage devices” and right hand side “Attributes”.
* Under storage devices
* Click on controller : IDE
* Click on empty disk.
* Under attributes
* Click on disk point -->choose disk file -->select [CentOS-7-x86\_64-Minimal-2009.iso](http://centos.excellmedia.net/7.9.2009/isos/x86_64/CentOS-7-x86_64-Minimal-2009.iso).
* Click on ok.

Step 4:- installation process.

* Click on start.
* Red hat will be open.
* We can see some timer will running, before the timer end click on esc button.
* First we can see boot:
* Type “linux text”.
* We can see installapion

Here, [x]-already configured.

[!]-have to configure – mandatory

[ ]-have to configure – optional

* Before installaption it wii ask to do some settings like time zone setting, selecting root password etc.
* Read the indtractions and do it.

**Setup network:-**

* Open oracle VMBox.
* Select the image.
* Make shoore the meashine is off.
* Go to settings and select network.
* We can see four adapters, choose any one.
* Put check mark to “Enable Network Adapter”.
* For Attached to select “Bridge Adapter”.
* For name it will take automatically.
* Expand Advanced.
* For promiscuous mode select “Allow All”.
* For mac address we can see number, click on refresh the value. The number will change.
* Cable connected should be enabled.
* Click on ok.

**Log into the meashine:-**

* Now start the meashine.
* We can see ***localhost login***: **root**
* Type the password what we have set.
* We see **[*root@localhost~*]#** that means we login successful.
* Now type the command “nmtui”.
* This is the command for centos and red hat to turn on the internet for the meashine.
* Press enter, we can see a pop up.
* Select “Activate a connection” press enter.
* Navigate key to activate, and press enter it will activate.
* After that come out back and quict.
* Now type the command “hostname -I” and press enter. It will show the ip-address.
* Copy the ip-address.
* Type the command to check internet is working or not “ping google.com”.
* We can see number of lines will come call as hope.
* To stop this, we use *ctrl+c.*
* Lets confirm our network is stable or not.
* We see packed loss, if we use it will be grater than “0% upto 10%” it will work normally greater than 10% it will make some trable.
* Now we are seeing 0% that means stable.
* Our network setting is ok.
* Command to power off the meashine “shutdown”.

**Create clone copy:-**

* When we creating a clone, the meashine have to be power off.
* Right click on created image.
* Click on clone, we can see name, path, mac address etc.
* For name:ex:git practice
* Change only name, click next.
* Clone type: full clone.
* Click on finesh.

**Note:** whenever we created a clone copy the MAC address of the VM meashine should be refreshed.

* To refresh select clone copy.
* Go to settings -->network -->click on refresh symbol (only one time)
* Click ok.

**Start the clone meashine:-**

* Now start the meashine.
* Login with root & password.
* Type command “nmtui” to activate network.
* Type command “hostname -I”.
* Note the ip address.

**What is MobaXterm ?**

MobaXterm is a toolbox for remote computing. In a single windows application it provides lodes of function that are tailored for pragrammers, webmasters, IT Administrators and pretty much all users who need to handle their remote jobs in a more simple fassion.

**Download MobaXterm:-**

* In google search MobaXterm download.
* Go to offical website.
* Download the free.
* It will take to another page.
* Click on green colour one and download it.
* The jip file will download.
* Right click on jip-file.
* Click on extract here.
* Extract the file in program files and install.

**How to connect red hat to mobaxterm:-**

* Now open the mobaxterm.
* Click on “session” left hand side top icon.
* Click on “SSH” for linux.
* Under remote host type ip-address.
* Click ok.
* Now we get window worning is called as ssh finger printing click on accept.
* Now login with root & password.

**What is git ?**

Git is a DevOps tool used for source code management. It is a free and open source version control system used to handle small to very large projects efficiently. Git is used to tracking changes in the source code, enabling multiple developers to work together on non-linear development.

**Installation of git:-**

* Open MobaXterm and login with credentials.
* Type command “yum install git”.
* Now press Y and again praess Y.
* Now verify git is install or not, by typing command “git --version”.

**Version Control System (VCS):-**

* The version controal systems evolved in 1960.
* In 1960 – 1990 we called it as revesion control system (RCS).
* In 1990 – 2010 we called it as source code management (SCM).
* In 2010 we called it as version control systems (VCS).

**Defination:-**

Version control, also known as source control, is the practice of tracking and managing changes to software code. Version control systems are software tools that helps software teams to manage changes to source code over time.

* VCS helps to the code changes.
* Experientations of code changes.
* VCS helps to redusing the storage.
* Maintain the different versions of the code.

In version controal systems we have two types.

1. Centralised vcs
2. Distributed vcs

**Centralised VCS:-**

* Centralised VCS came up in 1990’s era.
* CVCS products are most popolar in agile era.
* This centralised VCS introduced concept called client and server architecture.
* It is a 2-tier system.
* Reopsatory (repo) tracking our changes.
* Reposatory on server side.
* Operates in sync model.
* Dependency on network.
* Number of users connected to the repo, if users increases server slow.
* Data upload and download with server.
* Encryption is missing --- security flaw.
* If server go down – repo will be gone, to overcome need to take backup.

**Distributed VCS:-**

* DVCS came up in 2003.
* After 2010 it was well developed.
* It was used in devops era.
* 2-repo concept.

1. Repo sets on server side called as remote repo.
2. Repo sets on client side called as local repo.

* It is a 2-tire architecture.
* Belive in client and server technology.
* At the time of uplod and download we need network.

Version control software tools.

1. Git
2. CVS
3. SVN
4. Bazaar. Etc

**Git:-**

* Git is an open source.
* Git implements DVCS technology.
* That means it contains Local repo and Remote repo.
* Git encryption of SHS (secure Hash Algorithm).
* Git can work on windows/linux/mac.
* It will work on operating systems 32 and 64-bit.
* Language written in c-->peral-->python.
* Official site – [*https://git-scm.com/*](https://git-scm.com/)*.*
* No system requirements to install git.

**Git installation on windows:-**

Go to the official web site [*https://git-scm.com/*](https://git-scm.com/)

Screenshots of installation is availabe in below link

[*https://freshersnotes.com/how-to/how-to-install-git-on-windows/*](https://freshersnotes.com/how-to/how-to-install-git-on-windows/)

[*http://robertovormittag.net/ebooks/git-and-github/git-for-windows-installation-screenshots/*](http://robertovormittag.net/ebooks/git-and-github/git-for-windows-installation-screenshots/)

**Note:-** while installation, under screenshot “adjust path environment”, select “use git from git bash only”.

**Git installation on linux:-**

Q. Command to install git.

# yum install git

Q. Command to verify the installation.

# git --version

Expected output

[root@localhost~]#git --version

git version 1.8.31

**Git – configuration:-**

**Q. How to check the configuration of git ?**

**#** git config --list

* There are 7 or 8 configurations which can be configured.
* Out of this two configurations are mandatory.
* There are : user.email & user.name.

**Q. How to configure a user name ?**

# git config --global user.name <name>

Ex: git config --global user.name akhil

**Q. How to configure a user email id ?**

# git config --global user.email <email\_id>

Ex: git config --global user.email abc@gmail.com

**Configuration practically:-**

* Go to oracle vmbox, open the image and start the image.
* Log in with your credentials.
* To activate network type command “nmtui” and activate it.
* After activating network, type command “hostname -I” it will show the ip-address note it.
* Now open the mobaxterm, click on session which was on left side.
* Now click on ssh, it will ask for hostname type the ip-address and click on ok.
* Log in with same credentials.
* Type the command “git --version” to check weather git is install or not.
* Check weather we configur name & email by typing command “git config --list”.
* On next line we see nothing because we not configured.
* Now configur name by typing command –

“git config --global user.name Akhil”

* Now configure email by typing command –

“git config --global user.email [akhil@gmail.com](mailto:akhil@gmail.com)”

* After configuring user name and email check weather we configure or not by typing command “git config --list”. Now we can see user name and email.

**Git configurations use cases:-**

Q. What if the information user.name and user.email are incorrect ?

- Fix: re-run the command with correct information.

Q. How to change/update user.name and user.email ?

# git config --global user.name <name>

Q. Do you think the email id that was given in configuration is getting validated ?

- No validations.

**Git progect or local reposatory creation:-**

**Q. How to create the project or loacal reposatory in git ?**

# git init <project\_name>

Or

# git init <localRepo\_name>

**Project or local repo creating practically:-**

* To create poject or local repo type command “git init <project name>.

Ex:. git init project-1

* If we want to create folder or directory type command “mkdir folder\_name”.

Ex:. mkdir folder-1

* In linux we call folder as directory.
* To check weather the project or repo and folder is created or not, type command “ls -l”.
* We can see that repo and folder are looking same.
* To identify what is repo and folder, type command-

“cd <repo\_name or folder\_name>

* cd is the command to open the repo and folder.
* After taking in to the repo or folder type command –

“git status”

* If it is folder, gives message as “not a git reposatory”.
* If it is repo, what ever happening in the repo we will get status.
* To come out from repo or folder type command –

“cd ..”

**Q. How do you verify the git folder or repo ?**

Method.1:.

**#** cd <folder\_name or repo\_name>

# git status

Folder – not a git reposatory.

Reposatory – gives some information what happening in repo.

Method.2:.

# cd <folder\_name or repo\_name>

# ls -la

We should see .git folder it is a reposatory

**Note:-** In entire git two commands will respond if it is repo or folder i.e

“git --version” and “git config --list”. Rest of all comends will respond to repo.

Q. How many projects or reposatories we can create on the client side ?

- unlimited

Q. Will you create project or repo inside of another repo on client side ?

- No

Q. What are the two commands which can be executed irrespective of repo or project ?

# git --version

# git config --list

**Q. How do you find .git folder which is hidden exists in a project or localrepo ?**

# ls -l will show all files and folders.

# ls -la will show all files and folders including hidden files and folders also.

Q. Do we have to create the repo in the root folder only or can we create repo at any location ?

- Location does not matter.

**Git architecture:-**

****

* I created a repo, that repo was looking like a folder and it contains something .git.
* Physically it looks like a folder, but technically it is not a folder.
* That folder is started in three layer format.
* First layer is called working copy or working directory.
* Second layer is called staging or index area.
* Thard layer is called as commit area.
* The entire three layers in a box called local repo.
* We start creating files in working directory.
* If we want to move file from working directory to staging area, we have to use command “git add”.
* After that we have to move file from staging to commit area or localrepo, we have to use command “git commit”.
* We always start working from working directory after there travelled to staging after that commit.
* The file must travelled to commit area then only we can send to server.
* To send the file to server we have to type the command “git push”.
* If there any updates in server, we can get back the file from server by using command “git pull”.
* The repo which we are getting from the server is called as remote repo.
* The entire process we want to know what going on, type command “git status”.

**Working directory:-**

**Creating files:-**

**Q. How to create file and open the file ?**

# vi <file\_name>

**Q. How to write information in that file ?**

# esc+i

**Q. How to save the file ?**

# esc+:w

**Q. How to come out from the file ?**

# esc+:q

**Q. How to save and come out from the file at a time ?**

# esc+:w

**Creating text file practically:-**

* Open mobaxterm
* Lets go to the repo which was created by using command –

“cd project-1”

* Now type the command “ls -l” to check, if there is any thing in that repo.
* We will see nothing.
* Type command “vi states.txt”.
* By using this command we can open exising files also.
* This command will create the file and open the file at a time.
* To type the information in that file click “esc+I”.
* After clicking we can see, At bottom ‘Insert’ that means we can write some information.
* After writing some information save the file by clicking “esc+:w”.
* After saving the file come back from the file by clicking “esc+:q”.
* We can save and come out from the file at a time by clicking “esc+:w”.

**Now we created file in working directory.**

**Staging area:-**

**Q. How to take the file into staging area?**

# git add <file\_name>

* After saving the file type command “git status”, it will show whats happening in the repo.
* Now we will see that, message coming like below –

*Untracked files:*

*(use "git add <file>..." to include in what will be committed)*

* Type the command “git add states.txt” and click enter.

**Now we move the file to staging area.**

**Commit area:-**

**Q. How to take the file into commit area ?**

# git commit -m “message” <file\_name>

**Q. How to unstage the file?**

# git rm --cached <file\_name>

* Now again check the status by typing command “git status”.
* We can see message like this

*Changes to be committed:*

*# (use "git rm --cached <file>..." to unstage)*

*#*

*# new file: states.txt*

* In the message we can see that, if we want to unstage the file use command “git rm --cached states.txt.
* We want to commit for that use command –

“git commit -m “states names in file” states.txt”

* Now again check the status by typing command “git status”.
* We will get message like this,

*On branch master*

*nothing to commit, working directory clean*

**Now we move the file to commit area.**

Q. How to convert a folder which is already has some files (50) into a repo ?

* Go into the folder using command cd “<folder\_name>”.
* Type command “git init” with out clicking name.
* Because already have folder, it will convert to git reposatory.

**Q. How to create multipul files at a time ?**

# touch <file\_name1 file\_name2 file\_name3 ……… file\_name n>

**Benificts of using “touch” command in linux –**

* We can create a file which is empty.
* We can create multipul files.
* We can update time stamp of the files.

**Q. How to stage multipul files at a time ?**

# git add <file\_name1 file\_name2 ……file\_name n>

**Q. How to stage n-number of files with out using file names ?**

# git add .

**Q. How to commit multipul files at a time?**

# git commit -m “message” <file\_name1 file\_name2 ….. file\_name n>

**Q. How to commit n-number of files with out using file names ?**

# git commit -m “message” .

**Practically creating and commiting number of files at a time:-**

* Open mobaxterm and login with credentels.
* Now create files using a linux command –

“touch file1 file2 file3”

* Type command “ls -l” to check weather the files are created or not.
* Check the status of the files using command “git status”.
* We can see un tracked files name.
* To stage that files at a time use command “git add file1 file2 file3”.
* We can stage the files at a time using same command like this also –

“git add .”

* To commit the files at a time using command –

“git commit -m “files are commited” file1 file2 file3”

* We can commit the files at a time using same command like this also –

“git commit -m “files commited” .”

* Now check git status, we can see nuthing to commit.

**See changes in git:-**

**Q. How to know the files in working copy ?**

# ls -l

**Q. Command to see files in staging ?**

# git status

* We don’t have any specific command.
* “git status” can help in this regard.
* Based on the message help, have to understand about the files instaging.

**Q. Command to see commited files or in commited area ?**

# git ls-files

**Q. Why I am getting this message “nuthing to commit, working directory is clean” ?**

**-** we getting this message under below condations.

1. When number of files or content that is sitting in working copy, if it is equal to commit area.
2. If the number of files or what ever in the staging area, it should be equal to zero.

When condetion number 1 and 2 is true then only we get working directory is clean.

**Q. How to see the last commit or recent commit of the repo ?**

# git show

We get details like –

* Commit id,
* Author,
* Date-Time-Time zone,
* Commit message

#git show <commit\_id>

**Q. How to see all the commits so far on the repo ?**

# git log --all

Start to end

**Q. How to see recent commits done in repo ?**

# git log

It will show recent 10 commits.

**Q. How to see all the commits information ina short format ?**

# git log --oneline

**Q. How to find commits done by a perticular user in a project ?**

# git log --author=<name or email>

**Q.How to see specific details of a commit ?**

1st # git show <commit id>

2nd # git log -p

**Git unstage:-**

**Q. How to unstage a file ?**

# git reset HEAD <file\_name>

It will roll back the file into WC/WD from staging.

**Git revert:-**

**Q. Git revert command ?**

# git revert <Commit id>

* It helps to undo a commit.
* Git revert would be applied after you done the commit on a file.
* Applicable on the commited files.

**Revert pratically:-**

* Open mobaxterm and login with credentials.
* Go to the reposatory using command “cd project-1”.
* Type command “ls -l” to check what is there in the repo.
* We can see file states.txt.
* Open that file using command “vi states.txt”.
* We entered some states there. Now add some more states.
* Now we want to delete the added states.
* We have to know commit, for that type command “git log --oneline”.
* Now type command “git revert <commit Id>.
* Now go and see the file last commit will not there.

**Git checkout:-**

**Q. How to undo changes before commit ?**

# git checkout -- <filename>

**Q. How to delete a file or folder from repo ?**

# git rm <filename>

Any file that we delete, that will be auto-staged.

* Select a file that we want to delete ex – states.txt
* Type command “git rm states.txt”
* When we use “git rm” command to delete file, that file automatically staged.
* If we want to delete, then go for commit using command –

“git commit -m “delete states” states.txt”

* Now type the command “ls -l” to see files in reposatory, we will not see the states.txt file.

**Q. How to recover the deleted file ?**

- we can recover the deleted files in two ways.

Method 1.

* First we have to know the commit id for the deleted file.
* So, type command “git log --oneline”. It will show the commit id for the detete file.
* Type command “git revert <commit id>.
* It will ask reason for changes, click “esc+i” remove firstline and type message and save by clicking “esc+shift:x”.
* Now type command “ls -l” to check the reposatory. We can see the file again.

Method 2.

* After deletion of file it will be auto staged.
* Check the states using command “git status”. We can see (use command git reset HEAD <file\_name>) to unstage.
* We have to unstage the file using command “git reset HEAD states.txt”.
* Now type command “git checkout -- states.txt”.
* Type command “ls -l”, we can see that file states.txt.

**Cloud Hoting of Remote Repo:-**

**Use case 1:-**

Remote repo

server

local repos

If your organization has own server then we did not need git, bit bucket etc.

**Use case 2:-**

Remote repo

Local repos

* If your organization wants to host the repo on a thart party server then need cloud vendors. Who support remote repo on lease.

Ex:- github, bit bucket, assembled, gitlab etc.

**Challenges of hosting Remote Repo on own server:-**

1. Security
2. Availability
3. Down time
4. Network issues
5. Maintenance
6. Cost
7. Os licence
8. Resources to manage
9. Backup

**Repo types with cloud vendors:-**

Remote repoes are of two types

1. Public repo
2. Private repo

**Public repo:-**

* Free
* Limit in space
* Repo count
* Accessed by any persion who has github account
* Sutable for open source
* It is searchbul on internet

**Private repo:-**

* Pay for using private repo.
* All features apply
* Private repo can be accessed by only authorized persion

**Real time:-** When you join any project, they share your github id and password.

**Note:-**

* All non-cretical and medium critical repos are generally hosted on cloud.
* All critical data of repos are hosted on there own server.

**Github introduction:-**

1. Github created on 2007.
2. Purchased by microsoft in 2018.
3. Operates on principle of “pay per use model”.
4. Support two types of repos -public and private.
5. Features wise ranked-1 in the market.
6. Head quarters: San Francisco.

**Github features:-**

1. Free account
2. Public repo unlimited
3. Private repo with limitations
4. Pay per user
5. Largest
6. Network speed
7. Universel

**Github account creation:-**

* Go to google.com and type github in the search bar.
* Once we go to site just click sineup.
* Provide your Email id and create password.
* Enter your username.
* Solve the puzile.
* Click create account, we get otp for the email and verify it.
* Account is created.

**How to create the repo in the github ?**

* Login into the github.
* Then click on new.
* Give reposatory name.
* Write the discription it is optional.
* Select public or private.
* Put a check mark for “add readme file”.
* Click on “create reposatory”.
* Remote repo created.

**How to get url ?**

* Once open the repo in github.
* We see “code button” click on it.
* We ca see HTTPS is selected, under url is there just copy it.

**Interview questions on version controal syatem:-**

1. **What is version controal syatem ?**

Version controal systems are the software tools used to manage software team source code modification with time. Version controal software tracks each and every cahanges in the code. Developers can easily debug his privious work and compare earlier version of the code to fix the mistake. Version controal tools also helps developers to easily work in collebrating environment.

1. **What is GIT ?**

Git is a DevOps tool used for source code management. It is a free and open source version control system used to handle small to very large projects efficiently. Git is used to tracking changes in the source code, enabling multiple developers to work together on non-linear development.

1. **How will you know in git if a branch has been already merged into a master ?**

# git branch --merged master ( lists branches merged into master ).

# git branch --merged (lists branches merged into head).

# git branch --no-merged (lists branches that have not been merged).

By default this applies to only the local branches. The -a flag will show both local and remote branches, and the -r flag shows only the remote branch.

Or

You can use the “git merge-base” command to find the latest common commit between the two branches. If the commit is same as your branch head, then the branch has been completely merged.

1. **How can you bring a new featute in the main branch ?**

To bring a new feature in the main branch, you can use a command “git merge” or “git pull”.

1. **I want to delete branch both locally and remotely ?**

To delete the local branch use one of the following:

# git branch -d <branch\_name>

# git branch -D <branch\_name>

The -d option is an alies for --delete, which only deletes the branch if it has already been fully merged in its upstream branch.

Delete remote branch

You can delete a remote branch using

# git push <remote\_name> --delete <branch\_name>

1. **What is the diggrnce between “git remote” and git clone” ?**

“git remote add” just creates an entry in your git config that specifies a name for a particular URL. While ‘git clone’ creates a new git reposatory by copying and existing one located at the url.

git clone:

will physically download the file into your computer. It will take space from your computer. If the repo is 200mb, then it will download that all the place it in the diretory you cloned.

git remote add:

wont take space, its more like pointer, it doesint increase your disk consumption.

1. **How can you fix a broken commit ?**

To fix any broken commit, you will use the command “git commit—amend”. By running this command, you can fix the broken commit message in the editor.

1. **Why is it advisible to create an additional commit rather than amending an existing commit ?**

There are couple of reasons

1. The amend operation will distroy the state that was previously saved in a commit. If it’s just the commit message being changed then that’s not an issue. But if the contents are being amended then changes of eliminating something important remains more.
2. Abusing “git commit- amend” can cause a small commit to grow and acquire unrelated changes.
3. **Can you tell me layman terminologies diffrence betweeen git and git hub ?**

Git is a revison controal system, a tool to manage your source code history.

GitHub is a hosting service for git repositories.

1. **How do you find a list of files that has changed in a particular commit ?**

# git diff-tree -r {hash}

**List of questions:**

1) What is GIT?

2) What is a GIT repository?

3) What is the command to write commit message?

4) What is the difference between GIT and SVN?

5) What are the advantages of using GIT?

6) What language is used in GIT?

7) What is the role of GIT PUSH command?

8) Why GIT Subversion is better?

9) What is Staging Area in GIT?

10) What is GIT stash?

11) What is GIT stash drop?

12) How will you know in GIT if a branch has already been merged into master?

13) What is the function of git clone?

14) What is the function of ‘git config’?

15) What does commit object contain?

16) How can you create a repository in Git?

17) What is ‘head’ in git and how many heads can be created in a repository?

18) What is the purpose of branching in GIT?

19) What is the common branching pattern in GIT?

20) How can you bring a new feature in the main branch?

21) How can you fix broken commit?

22) How can conflict in git resolved?

23) What is the command to delete branch?

24) What is another option for merging in git?

25) WWhat is ‘git add’ is used for?

26) What is the difference between ‘git remote’ and ‘git clone’?

27) What is GIT version control?

28) Mention some of the GIT client for LINUX?

29) What is Subgit?

30) What is the function of ‘git diff ’ in git?