Git fetch – This command checks are there any changes in remote repository which are not available in local repository..

Local repository we ll have , github will have remote repository

Git pull – this command is used to download latest changes from central repository to our local repository…

CI (continuous integration) – multiple commits or multiple changes that is been done by different developers can be integrated in the shared repository without affecting the live application or production environment

Types of environment

dev environment - when dev commit changes and build on successfully

QA – quality assurance(basically testing)

PPD – pre release environment

Prod – main environment (in which we deploy )

DR – Disaster recovery (when we have issue with prod environment we can use prod environment)

Continuous deployment process

1. Deployment pipeline
2. Automated deployment
3. Testing
4. Release and roll back
5. monitoring

Devops benifts

Time saved

Early bug detection

Increased collaboration – with operation team + dev team members

Improved quality

Agility and flexibility – apply and deploy changes in less time

Questions

What is Devops – it is a process

CI and CD – 5 environment

Merge

Merging is the process of combining changes from one branch into another. This is typically done to incorporate the changes made in a feature branch back into the main branch

. Version Control System (VCS)

 Git allows multiple developers to collaborate on a project by keeping track of changes to the source code over time. It provides a history of changes, facilitates collaboration, and helps manage different versions of a project.

How GIT works:

Refer gfg notes

GIT VS GITHUB

Git is a distributed version control system used for tracking changes in source code. GitHub, on the other hand, is a web-based platform that hosts Git repositories and provides collaboration tools such as pull requests and issue tracking

And can easily revert it back to older

Repository – it is a central location where version control s/w stores all the files and data related to project

It contains the history of all the changes made to the file along with the metadata such as author , date and commit message.

Revision

Branch – copy of main code , developers work on branches and bug fixes

Git – Git is a powerful distributed version control system used to manage source code efficiently , collaborate seamlessly and track changes effectively

Git hub is a web based platform to manage repositories and make collaboration easier

Version control – same file with different changes with

Git init – initialize the simple reposity with the git repository

Open git bash here

Git status

Git add .

**7) What is  the fundamental differences between DevOps & Agile?**

The key difference between Agile versus DevOps is that Agile is a philosophy about how to develop and deliver software, while DevOps describes how to continuously deploy code through the use of modern tools and automated processes.

8)Why do we need Devops?

Devops is a process of improving the application delivery by ensuring proper Automation, with a code quality, ensuring Continuous Monitoring and Continuous Testing.

9)Why Devops?

To improve the process of delivery, Earlier multiple teams are involved and delivery would more time than expected .

* Lower failure rate of new releases

Increase deployment frequency

* In case of new release crashing, have a faster mean time to recovery

10) Introduce you as a Devops Engineer?

I was working as a Embedded Software Engineer

Roles and responsibility

I have worked on Automation

Quality

Monitoring

Testing

11)What tools ur working on

Ansible, terraform, Kubernetes add more

12)SDLC - Process followed in SDLC

End goal of SDLC – Deliver a high quality product

12) **What do you understand by anti-patterns of DevOps?**

An organization needs to have a separate DevOps group

* Agile equals DevOps
* DevOps is a process
* DevOps is development-driven release management

13)  **what are the major benefits of implementing DevOps automation?**

* Removal of the possibility of human error from the CD equation (Core benefit)
* As tasks become more predictable and repeatable, it is easy to identify and correct when something goes wrong. Hence, it results in producing more reliable and robust systems.

**12) Please explain the core operations of DevOps in terms of development and infrastructure.**

* Application development -Developing a product that is able to meet all customer requirements and quality.
* Code coverage
* Code developing
* Configuration – Allowing the product to be used in an optimum way
* Deployment – Installing the software to be used by the end-user.
* Packaging – Activities involved when the release is ready for deployment
* Provisioning – Ensuring that the infrastructure changes arrive just-in-time with the code that requires it
* Unit testing – Meant for testing individual units or components

**13)I type http://www.yahoo.com in my browser’s URL bar and I press enter. What happens ? (discuss at every OSI layer - Physical, data link, network, transport, session, presentation, application)**

In summary, when you type http://www.yahoo.com and press enter, your computer goes through these OSI layers, initiating a process that involves translating the human-readable URL into an IP address, establishing a connection, and requesting the desired web page from the Yahoo server. The server responds by sending back the requested web page data, and your browser then interprets and displays it to you.

**20) What is a virtual IP address ?**

A virtual IP address (VIP) is an IP address that is not assigned to a specific physical network interface or device but is instead associated with a virtual resource or service.

Git commands and their uses (init, clone, status, diff, log, etc.)

**git init**

The **git init** command is used to start a new repository. It creates a new **.git** subdirectory in your current working directory.

git init

git clone

The **git clone** command is used to create a copy of an existing repository

### ****git status****

The **git status** command shows the status of the working directory and the staging area.

### ****git add****

### ****git commit****

The **git commit** command saves the staged changes to the local repository

### ****git diff****

The **git diff** command shows the differences between the working directory and the staging area

### ****git pull****

The **git pull** command fetches changes from the remote repository and merges them into the current branch of your local repository.

$ git pull origin main

### ****git push****

The **git push** command is used to push committed changes to the remote repository.

$ git push origin main

### ****git merge****

The **git merge** command is used to merge changes from one branch into another.

Code review:

Code review  benefits

 Error Detection: Identifying bugs, logic errors, and potential security vulnerabilities before the code is merged.  
- Knowledge Sharing: Encouraging collaboration and knowledge transfer among team members.  
- Code Consistency: Enforcing coding standards and best practices throughout the codebase.  
- Quality Assurance: Enhancing code quality, readability, and maintainability.

2. Setting Up a Code Review Process:  
To establish an efficient code review process using GitHub, the following steps can be followed:  
- Create a pull request (PR): Developers initiate a pull request to propose changes and request a review from their peers.  
- Reviewers' Role: Reviewers carefully examine the proposed changes, provide feedback, and suggest improvements.  
- Collaboration and Discussion: Developers and reviewers engage in discussions within the PR to address feedback and clarify any questions.  
- Iterative Improvements: Developers make necessary changes based on the feedback received until the code is deemed ready for merging.

In the code review process: (peer review)

We will review the code of multiple developers and if correction is required in any of the function we will make and changes and create pull request , so that the other developers can get to know there is something as change in the code , if he finds that the changes made is correct or finds okay , he can merge the changes

**CI/CD pipeline**

The CI/CD pipeline is a series of automated steps that enable developers to continuously integrate, build, test, and deploy software changes in a controlled and efficient manner. It ensures that software is always in a deployable state and reduces the risk of introducing errors into the production environment.

**Jenkins Initial Setup**

**Install Jenkins.msi file**

In the Jenkins login first , install necessary plugins, git, github, maven plugin which are required , then

Go to the tools section - > click on Maven Installation - > give name MAVEN\_HOME

Then select latest version, click on install , then it will be installed locally in the Jenkins

Go to New Item - > Enter an item name example (GFG\_JOB) select Maven project -> click OK

Go to Configuration page ->

Description – my first Jenkins job

Source Code Management – yes (GIT)

Repository URL – copy from code space in GITHUB and select URL and paste it here

Specify the branch

pom.xml -- **everything should be in small case**

Goals – install

Click on save

Now click on Build NOW

You can see build no 1 is processing

When I want to automate the build process,

Go to Configure in GFG-JOB

In build Triggers -> select Poll SCM

Give pattern \* \* \* \* \* - every minute it will look into source code , if changes is made then it ll build every minute

Give pattern H \* \* \* \* run build every hour

**CI/CD with Jenkins and GitHub Actions (Build Triggers, CI/CD Pipeline Project)**

Jenkinsfile -> written in Groovy

Create new Jenkinsfile in eclipse project (simple file without any extension)

Pipeline {

agent any

tools {

}

}

In Jenkins

GFG\_B17-Pipeline -> select pipeline

Configure -

2 option -> pipeline script form SCM

SCM -> Git

Repository URL – give that URL here https:// .git

Scriptfile - > Jenkinsfile

Save it

Click on Build Now

Poll SCM - \* \* \* \* \*

Save it

Open blue ocean - plugin

What is Jobs and Pipeline

What is Jenkins?

Jenkins is a open source continuous integration server written in Java that allow continuous development, test and deployment of codes, it has got very good pluglins for Source code management , build and deployment , Testing ..

what is differnce between Jenkins Job and Jenkins Pipeline

A Jenkins Job is a single task or a series of tasks that can be executed on Jenkins.

t is typically configured using the Jenkins web interface by specifying build steps, post-build actions, and other configurations.

A Jenkins Pipeline is a suite of plugins that enables the definition and automation of build, test, and deployment workflows as code. Pipelines are defined using a domain-specific language (DSL) called Groovy DSL.

Comparison:

Complexity: Jobs are simpler and suitable for straightforward build and deployment tasks, while Pipelines are more powerful and better suited for complex, multistage workflows.

Configuration Management: Pipelines provide better configuration management as they are defined as code and can be version-controlled, promoting collaboration and traceability.

Flexibility: Pipelines offer more flexibility, making them suitable for a wide range of scenarios, from simple to highly complex build and deployment processes.

### Continuous Delivery

What is the difference between continuous delivery and continuous deployment

Continuous Delivery is a software development practice where code changes are automatically prepared and made ready for release at any time. The goal is to ensure that the software can be released reliably and efficiently at any point in the development process.

### Continuous Deployment

**Definition:** Continuous Deployment is an extension of Continuous Delivery in which every code change that passes the automated testing phase is automatically deployed to production without manual intervention

**Docker**

**Docker Commands**

Docker –version

Docker run -it ubuntu /bin/bash

Exit –

docker images – display images which are downloaded

docker pull python – download python image

docker run ubuntu – make container in machine

docker run -it ubuntu / bin/bash – to go inside container

docker run wordpress – it is used to pull and run using this command

docker deamon should be running – like docker rngine should be runinning , it act as a server

docker cli will ask the docker daemon for particular image and once it is there it pull the image and download

Docker Contatiner commands

Docker run <imagename>

docker exec <contd-id>

docker ps -> to list running container

docker ps -a -> to list running / not running container

docker start <container id>

docker exec -it <container id> /bin/bash -> when u want to go to the container already created

docker stop <cont id>

docker rm -> delete container

docker ps -a

docker run - d -p <host port>: <cont -port> <img- name>

docker run -it –name=myubuntu ubuntu/bin/bash – my image name

volume mapping

Any data in host machine is mapped with data in remote machine

Sharing the data between two or more containers

docker run -it -v d :/test : /data openjdk:11.0 /bin/bash

we can edit the container and changes will be reflected in host machine folder or file

docker run - d -p 8008 : 80 nginx – port mapping -d -> detached mode

go to localhost:8008

Welcome to nginx

docker run - d -p 8009 : 80 httpd

1. docker volume
2. build own docker image
3. push our docker image on dockerfile
4. docker network
5. cicd pipeline using github actions

docker volume is used for data persistant (copy of folder will be there in host machine which is mapped to container

docker ps -aq

docker run -it –name=myubuntu ubuntu /bin/bash

create folder in host name - test

to map the folder which is local machine with folder in host machine

docker run -it - -name=myubuntu -v D: /test:/gfg ubuntu /bin/bash

Dockerfile

FROM httpd (image name)

WORKDIR . /usr/local/apache2/htdocs

docker build -t webapp:1.0 .

docker images

docker run -it mywebapp:2.0 /bin/bash

.dockerignore - file name

Password.txt

\*.mp4

Dockerfile

Docker run -it mywebapp:3.0 /bin/bash

docker tag <imageid of webapp> hk/gfg17:1.0 <docker hub repository name>

docker login

docker push hkshitesh/gfgb17

docker pull hkshikesh/gfgb17

docker run -p 8006 :80 -d hkshikesh/gfgb17:1.0

Docker network

Kubernetes

One pod can contain more than 1 container

Always 3 pods for replicaset

We can run multiple multiple container in pod

Each container given the label

Start 1 .pod

2. service

When we Wnt to spin up pod mininmum requiremnts should be given are

Pod -name , label , docker images

Spin services - 1> service Name

2> pod label

3>

kubectl get all

kubectl apply -f pod.yml

kubectl apply -f service.yml f – folder name

kubectl get all

localhost:30001 welcome to nginx webpage

What are containers?

Containers is a package which is a combination of application libraries , system dependencies

Containers are light weight .

Docker containers doesn’t have a full operating system and they use the resources from base os, it has only minmum os which is required to run containers

We can easily deploy the container to Kubernetes platforms

Life cycle of docker

* 1. write a dockerfile
  2. Create a image ,

Let us consider one official docker image i.e., ubuntu

This ubuntu base image consists of files and folders like /sbin ,/ bin , /etc, /var , /root

And the file and folders this container use from host operating system are network stack , system calls, Namespace . control groups

Docker client –whenever we are running docker cli commands it is recived by docker daemon and it is executed , they create docker image , containers

Deamon is listening to containers

Docker Deamon helps in build , pull , run

Docker registry sytores Docker images , registry can be private , public registry

Brigde network – veth, docker0 network

Devops(2024)

1.What , why (SDLC)?

2. learn linux

how to create a linux machine in Aws account

Shell commands

3.What is Virtualization

4. Shell scripting (basic shell scripting)

5. Git

6. CI/cd -> complicated

Jenkins /CD

7.Configuration Management,

What is configuration management

Playbook -> 2VM (httpd)

8. Infrastructure -> Terraform

9. Docker(Containerazation)

Vm diff from containers

Why

How to build , create container, volume , networking

10. Kubernates – How conatiners are different from ks8

From devops zero to hero -abhishek

Things need to be learned

Advanced Concepts

Ci/cd

Shell scripts

Monitoring

K8s

Cloud platform

Aws is a cloud platform

Aws zero to hero

Terraform Advanced

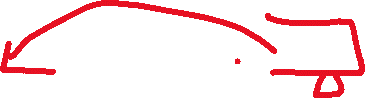
Python

AWS

Aws cloud platform -> EC2

S3

VPC



Redhat redhat

cat /etc/group

List all the groups present

cat /etc/passwd

how to create group

sudo groupadd gfggroup

cat /etc/group – we can see new group added gfggroup

sudo useradd -m -G gfggroup gfguser

cat /etc/passwd

sudo passwd gfguser

give password minimum 8 charracters

open visudo shuld be opened from root user

root All=(ALL) ALL

gfguser All=(ALL) NOPASSWD: ALL

sudo usermod -aG wheel gfguser

su -gfguser

password :

sudo yum install httpd

systemctl start httpd

create redhat instance 2

same gfg-redhat key .ppm

use same default security group

launch 2 instance

now inside 2 instance

create new user in 2 machine

sudo su

sudo adduser gfg

passwd gfg

enter

Now open visudo file

Add gfg user

gfg ALL=(ALL) NOPASSWD: ALL

su – gfg

gfg@ip -172-31-1-73 ]$

swich to 1 account

sudo su

adduser gfg

passwd gfg

give password

enter

open visudo file

Add gfg user

gfg ALL=(ALL) NOPASSWD: ALL

and save using wq

su – gfg

gfg@ip -172-31-15-229 ]$