Devops: Devops is a collaboration b/w development and operation teams which enables CD (continuous delivery) of applications and services to our end users.

Dev- planning and codebase (git and jira), building (Gradle, Maven), testing (selenium)

In middle Jenkins will come.

Ops-deploy and operate (Docker, chef), monitor (Nagios)

1.What is Docker?

A) Docker is a tool which is used to automate the deployment of applications in lightweight containers so that applications can work efficiently in different environments.

(Container is a s/w package that consists of all the dependencies required to run an application)

**Or**

Docker is an open-source software platform to create, deploy and manage virtualized application containers on a common operating system (OS), with an ecosystem of allied tools.

Disadvantage 🡪Daemon is running in backend, if we’re using with root, then if any using others cli then problem comes

2. What is Podman?

A) pod manager tool

1.Podman is **a daemonless container engine for developing, managing, and running OCI (open container** Initiative**) Containers on your Linux System**. Containers can either be run as root or in rootless mode.

Simply put alias docker=podman.

2. Podman is an open-source container management tool for **developing, managing and running OCI containers**. Let's take a look at some of the advantages of Podman, in comparison with other container management tools: Images created by Podman are compatible with other container management tools

(Docker daemon is **a persistent background process that manages the containers on a single host**. It is a self-sufficient runtime that manages Docker objects such as images, containers, network, and storage. Docker daemon listens for REST API requests and performs a series of container operations accordingly.)

3. what is skopeo?

A)

1.Skopeo is **a tool for moving container images between different types of container storages**. It allows you to copy container images between container registries like docker.io, quay.io, and your internal container registry or different types of storage on your local system.

4.what is build-ah?

A)

Buildah is **a command-line tool for creating Open Container Initiatives (OCI) or traditional Docker images**. And then building working containers from those images is up to you. It allows the containers to be mounted and modified. You can also save images based on the updated containers.

### Buildah and Podman Relationship

Buildah plays very nicely with Podman. There is some overlap in functionality between Buildah and podman. But the split-up of core responsibilities is clear. Podman itself can also build images. It actually uses Buildah on the backend while creating them. For instance, podman build calls the buildah bud command.

5. what is container?

A)

A container is **a standard unit of software that packages up code and all its dependencies, so the application runs quickly and reliably from one computing environment to another**.

6.what is pod?

A)

Collections of containers or grouping of containers are called pods.

A Pod represents **a single instance of a running process in your cluster**. Pods contain one or more containers, such as Docker containers. When a Pod runs multiple containers, the containers are managed as a single entity and share the Pod's resources.

7.what is registry?

A)

The registry or Windows registry is **a database of information, settings, options, and other values for software and hardware installed on all versions of Microsoft Windows operating systems**.

Or

The Windows Registry is a collection of databases of configuration settings for Microsoft operating system.

8. what is public and private registry?

A)

**Public registries are for public usage and open-source projects**. Examples include in like Docker Hub.

Whereas private registries are for organizational use or for your own. Examples for private registries include Azure Container Registry, EC2 Container Registry etc.

9. difference b/w container and pod?

10.difference b/w docker, postman, skopeo and buildah?

11.what is difference with container file and docker file?  
A)

Dockerfile is **a text document that contains all the commands a user could call on the command line to assemble an image**. Using docker build users can create an automated build that executes several command-line instructions in succession. This page describes the commands you can use in a Dockerfile.

Every Safe works with an encrypted **Container File**. It is the image of the virtual drive stored in a big file. Your encrypted files and folders are actually stored in this Container File.

The Container File cannot be accessed if it resides in the Default Container Folder. The access to this folder is blocked to ensure a greater security (mostly against accidental deletion). You can use the information stored in an Container File only through SafeBit program. The Container File is encrypted using the strong 256-bit AES encryption algorithm.

### Docker Image:

A Docker Image is a read-only file with a bunch of instructions. When these instructions are executed, it creates a Docker container.

### Dockerfile:

Dockerfile is a simple text file that consists of instructions to build Docker images.

Or

It is basically a text file which contains some set of instructions. Automation of docker image cration.Always D is capital letter on Docker file and Start component also be capital letter.

docker file docker build docker image

Text file

https://www.youtube.com/watch?v=3c-iBn73dDE

<https://birthday.play-with-docker.com/pods-with-docker/>

https://www.simplilearn.com/tutorials/docker-tutorial/what-is-dockerfile

https://www.youtube.com/watch?v=72LG34wIWI8

|  |  |  |
| --- | --- | --- |
| --all , -a |  | Show all containers (default shows just running) |
| --filter , -f |  | Filter output based on conditions provided |
| --format |  | Pretty-print containers using a Go template |
| --last , -n | -1 | Show n last created containers (includes all states) |
| --latest , -l |  | Show the latest created container (includes all states) |
| --no-trunc |  | Don't truncate output |
| --quiet , -q |  | Only display container IDs |
| --size , -s |  | Display total file sizes |