Docker is containerization technology tool.

Kubernetes is container orchestration tool.

**Docker is a container runtime.**

**Containerization:**  
 Its all about deploy application with required dependencies is known as containerization.

**Kubernetes:**

Kubernetes is a platform for running and managing container for many(100’s of) containers runtimes.

**Virtualization:**

-------------------------------------------------------------------------------------------------------------------

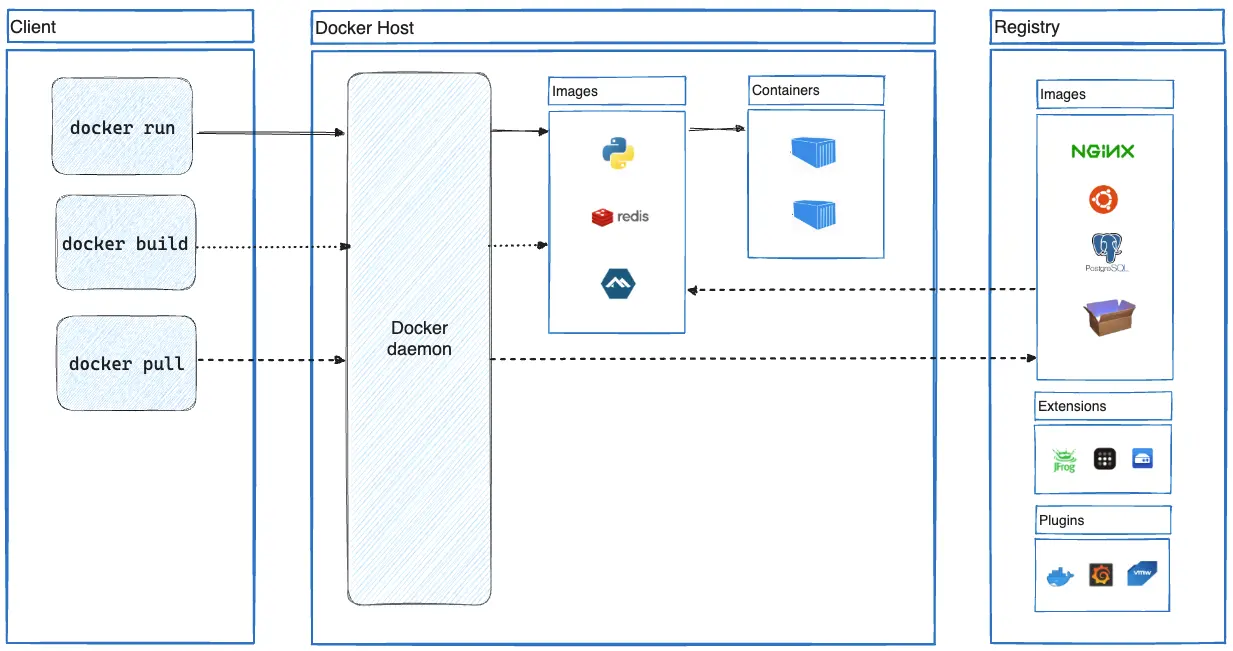
**Container Engine:**

Software which helps to implement containerization on a machine or server is caller container Engine.

Tool:

Docker, Jail, Crio.

**Docker Architecture:**



**Docker CLI/Client:**

Docker client is used to interact with docker hub to interact with images or containers.

* **Docker pull :** It helps to get the images from docker hub into the server.
* **Docker build:** To create docker images.
* **Docker run:** To create the containers from the images.

**Docker Daemon (Docker service):**

Docker daemon manages all the services by communicating with other daemons. It manages docker objects such as images, containers, networks, and volumes with the help of the API requests of Docker.

(The Docker daemon listens for Docker API requests and manages Docker objects such as images, containers, networks, and volumes. )

**Docker Host**

A Docker host is a type of machine that is responsible for running more than one container. It comprises the Docker daemon, Images, Containers, Networks, and Storage.

**Docker Registry**

All the docker images are stored in the docker registry.

**Docker Images**

An image contains instructions for creating a docker container. It helps to create container from docker images and it is used to store and ship applications.

**Docker Containers**

Containers are created from docker images. With the help of Docker API or CLI, we can start, stop, delete, or move a container.

**Docker Networking**

Docker Networking provides complete isolation for docker containers. It means a user can link a docker container to many networks. It requires very less OS instances to run the workload.

**Types of Docker Network**

1. **Bridge:** It is the default network driver. We can use this when different containers communicate with the same docker host.
2. **Host:**When you don’t need any isolation between the container and host then it is used.
3. **Overlay:**For communication with each other, it will enable the swarm services.
4. **None:**It disables all networking.
5. **macvlan:**This network assigns MAC(Media Access control) address to the containers which look like a physical address.

Are you ready to unleash the power of **DevOps**to streamline your **Software Development and Deployment**? Learn about our [**DevOps Live Course**](https://gfgcdn.com/tu/Q8P/) at GeeksforGeeks, created for all professionals in practice with continuous integration, delivery, and deployment. Learn about leading tools, industry best practices, and techniques for **automation**through an interactive session with hands-on **live projects**. Whether you are new to DevOps or looking to improve your skills, this course equips you with everything needed to streamline workflows and deliver excellent quality software in the least amount of time. Learn to take your skills in DevOps to the next level now, and harness the power of streamlined software development!