**Docker Swarm**

Docker Swarm is an orchestration management tool that runs on Docker applications. It helps end-users in creating and deploying a cluster of Docker nodes.

Each node of a Docker Swarm is a Docker daemon, and all Docker daemons interact using the Docker API. Each container within the Swarm can be deployed and accessed by nodes of the same cluster.

There are five critical elements within a doctor environment:

1. Docker container
2. Docker daemon
3. Docker images
4. Docker client
5. Docker registry

**Features of Docker Swarm**

Some of the most essential features of Docker Swarm are:

* Decentralized access: Swarm makes it very easy for teams to access and manage the environment
* High security: Any communication between the manager and client nodes within the Swarm is highly secure
* Autoload balancing: There is autoload balancing within your environment, and you can script that into how you write out and structure the Swarm environment
* High scalability: Load balancing converts the Swarm environment into a highly scalable infrastructure
* Roll-back a task: Swarm allows you to roll back environments to previous safe environments

**How Does Docker Swarm Work**

In Swarm, containers are launched using services. A service is a group of containers of the same image that enables the scaling of applications. Before you can deploy a service in Docker Swarm, you must have at least one node deployed.

There are two types in Docker Swarm:

1. Manager node. Maintains cluster management tasks
2. Worker node. Receives and executes tasks from the manager node

**Docker Networking**

Docker networking enables a user to link a Docker container to as many networks as he/she requires. Docker Networks are used to provide complete isolation for Docker containers.

**Advantages of Docker Networking**

Some of the major benefits of using Docker Networking are:

* They share a single operating system and maintain containers in an isolated environment.
* It requires fewer OS instances to run the workload.
* It helps in the fast delivery of software.
* It helps in application portability.

**How Does Docker Networking Work**

* Docker file builds the Docker image
* Docker Image is a template with instructions, which is used to build Docker Containers.
* Docker has its own cloud-based registry called Docker Hub, where users store and distribute container images.
* Docker Container is an executable package of an application and its dependencies together.

**Functionalities of the different components:**

* Docker File has the responsibility of building a Docker Image using the build command
* Docker Image contains all the project’s code.
* Using Docker Image, any user can run the code to create Docker Containers.

Once Docker Image is built, it’s either uploaded in a registry or a Docker Hub

**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.

Mentioned below is the syntax of a Dockerfile:

Syntax

*# comments*

*command argument argument1...*