

**AIM:**Containerdecode onproductionusingDocker.

DESCRIPTION:

Dockeroverview:

Docker is an open platform for developing, shipping, and running applications. Dockerenables you to separate your applications from your infrastructure so you can deliversoftware quickly. With Docker, you can manage your infrastructure in the same waysyou manage your applications. By taking advantage of Docker’s methodologies forshipping, testing, and deploying code quickly, you can significantly reduce the delaybetweenwriting codeandrunningitinproduction.

DockerArchitecture:

Docker uses a client-server architecture. The Docker *client* talks to theDocker *daemon*, which does the heavy lifting of building, running, and distributing yourDocker containers. The Docker client and daemon *can* run on the same system, or youcan connect a Docker client to a remote Docker daemon. The Docker client and daemoncommunicate using a REST API, over UNIX sockets or a network interface. AnotherDockerclientisDockerCompose,thatletsyouworkwithapplicationsconsistingofasetofcontainers.

BasicDockerCommands:

1.finding the version:

docker--version

2.downloadingimage:

dockerpullhttpd

1. Images:

dockerimages

1. run:

dockerrun-it-dhttpd

1. what’srunning?

dockerPs

1. Ps–a

dockerPs-a

1. exec

dockerexec–itfilenamebash

1. removingcontainer

dockerrmcontainername

1. Removingimage

dockerrmiimageid

1. restartdocker

dockerrestartcontainerid

1. commit

dockercommitfilename

1. push

dockerpushimagename

1. kill/stopping/starting

dockerkillcontainerid