**Docker Networking**

**Managing Networks**

10 docker network ls

11 docker network create -d bridge network1

12 docker network ls

13 docker network create network2

14 docker network ls

15 docker network rm network1 network2

**Connecting User Defined Network with container**

|  |
| --- |
|  |

docker network create network1

docker pull somidocker/ubuntu\_devops

docker run -td --name container1 --network network1 -p 8080:8080 somidocker/ubuntu\_devops /bin/bash

docker inspect container1

docker network inspect network1

**Establishing connection between containers using bridge network**

docker run --name container2 --network network1 -td -p 8081:8080 somidocker/ubuntu\_devops

docker run --name container3 --network network1 -td -p 8082:8080 somidocker/ubuntu\_devops

docker inspect container2 | grep IPAddress

docker inspect container3 | grep IPAddress

docker exec -it container2 /bin/bash --> Ping commands

docker exec -it container3 /bin/bash --> Ping commands

docker network create network2

docker network connect network2 container2

docker network connect network2 container3

docker inspect container2 | grep IPAddress

docker inspect container3 | grep IPAddress

docker exec -it container2 /bin/bash

docker network disconnect network1 container1

docker network disconnect network1 container2

docker network disconnect network1 container3

docker container stop container2 container3

docker container rm container2 container3

docker network rm network2

docker network ls

**Host Network**

docker network ls

docker run -td --network host --name jenkins -p 8080:8080 somidocker/ubuntu\_devops

yum install httpd -y

service httpd start

docker exec -it jenkins service jenkins start

curl localhost:8080

docker run -td --network network1 --name jenkins8080 -p 8080:8080 somidocker/ubuntu\_devops

docker exec -it jenkins8080 service jenkins start

curl <IP>:port