```
Docker Engine Installation codes:
vi docker.sh
[sudo yum update -y
sudo yum search docker
sudo yum info docker
sudo yum install -y docker
sudo systemctl enable docker.service
sudo systemctl start docker.service
sudo systemctl status docker.service
docker version
sudo curl -L
https://github.com/docker/compose/releases/download/1.22.0/dockercompose-$(uname
-s)-$(uname -m) -o /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
docker-compose version]
su filemane.sh---------------------is used to install the docker

Docker codes to do operations in linux connect:
docker images
docker ps
docker pull imagename or container id
docker run -it -d imagename or container id --------------it is used to
convert the image into container
docker stop imagename or container id
docker ps
docker ps -a
docker start imagename or container id
docker ps -a

first create an instance
connect that instance with mobextreme
in dockerhub create a new repository
in mobextereme install the docker
pull an image------docker pull image_name
convrt it into a container--------docker run -it -d image_name
commit the image by using---- docker commit containerid repository_name
push the image into the repository using----- docker push repository_name
to login to the dockerhub using instance use docker login
check the dockerhub we will get the pulled image from the docker

to pull webserver
docker pull nginx
docker images
docker run -it -d -p 8000:80 nginx
docker ps
go to instance copy the ipv4 address and paste it with :8000
we will get the welcome page of nginx

🡅------------to pull apachetomcat------docker pull amazonlinux
docker images
docker run -it -d container_name
docker ps
docker exec -it container_id bash -------------------------to go inside the container
go to
copy the link of tar.gz
wget(to get the details of the mentioned url) tomcatlink
yum install wget(to install wget)
wget link of tar.gz
download tar and gzip -----------------yum install tar
ls
tar -zvxf tared_file
cd bin
ls
./startup.sh
yum install java -y
./startup.sh
```

To build own image

```
mkdir Dockerfile
cd Dockerfile
vi dockerfile ----i----inset the commands
[FROM amazonlinux
MAINTAINER <Enter Docker hub mail_id>
RUN yum update -y
RUN yum install -y wget
RUN yum install -y tar
RUN yum install -y gzip
RUN wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.73/bin/apachetomcat-9.0.73.tar.gz
RUN tar -xf apache-tomcat-9.0.73.tar.gz
RUN yum install -y java
RUN sh apache-tomcat-9.0.73/bin/startup.sh]
press esc :wq
cd..
go for dockerhub and create a new repository
docker build -t repository_name Dockerfile
docker run -it -d repository_name
docker ps
docker exec -it container_id bash
```

🡅Docker Compose

Multiple containers working in a single sevice

```
vi docker-compose.yml
insert the codes inside the file
[version : '3'
services:
web:
image: nginx
ports:
```

```
- 4000:80
db:
image: redis]
esc button +:wq
cat docker-compose.yml
docker-compose up -d
docker ps
docker-compose down
docker ps

Docker volume
there are 2 types of volumes they are
a) named volume
b) anonymous
volumecd /var
ls
cd lib/
cd docker/
cd volumes/
docker volume create name------------------to create a new volume
docker volume ls---------------------------to get the list of volumes present
docker volume inspect name-----------------to inspect the volume
docker volume rm name----------------------to delete an volume
docker pull httpd
docker images
docker run -it -d httpd
docker ps
docker exec -it container_id bash
ls
cd htdocs
ls
cat index.html
cd
exit
docker volume create newname
docker volume ls
docker images
docker run -itd -v virus:/usr/local/apache2/htdocs/ -p 6000:80 httpd
docker ps

⬆docker exce -it container_id bash
ls
cd htdocs/
ls
make duplicate the window
sudo su cd /var/lib/docker/volumes/
ls
cd virus
ls
```

```
cd _data
ls
in main window type
touch sample.txt
ls
again in duplicate window
ls
we will get the newly created file

Docker swarm
lanch 2 instances
install docker in both the instances
docker version
in masternode
docker swarm init--------------to initiate swarm in master_node
-in workernode
docker swarm join --token........copy this command and paste
we will get swarm as a worker
-in masternode
docker node ls-----------------it is used to ckeck how many nodes are
present
docker node promote hostname(which is not having as leader)
docker node ls
docker node demote hostname(it is used to demote the worker from manager)
docker swarm leave --force ------------------------in workernode
again in managernode type docker node ls(now it will show status of
worker as down)
```