Experiment No. 9

Aim: To learn Docker file instructions, and build an image for a sample web application using a Docker file.

LO No. & Statement: (LO5): Explains the concept of containerization and Analyzes the Containerization of OS images and deployment of applications over Docker.

Theory:

Importance of Docker file:

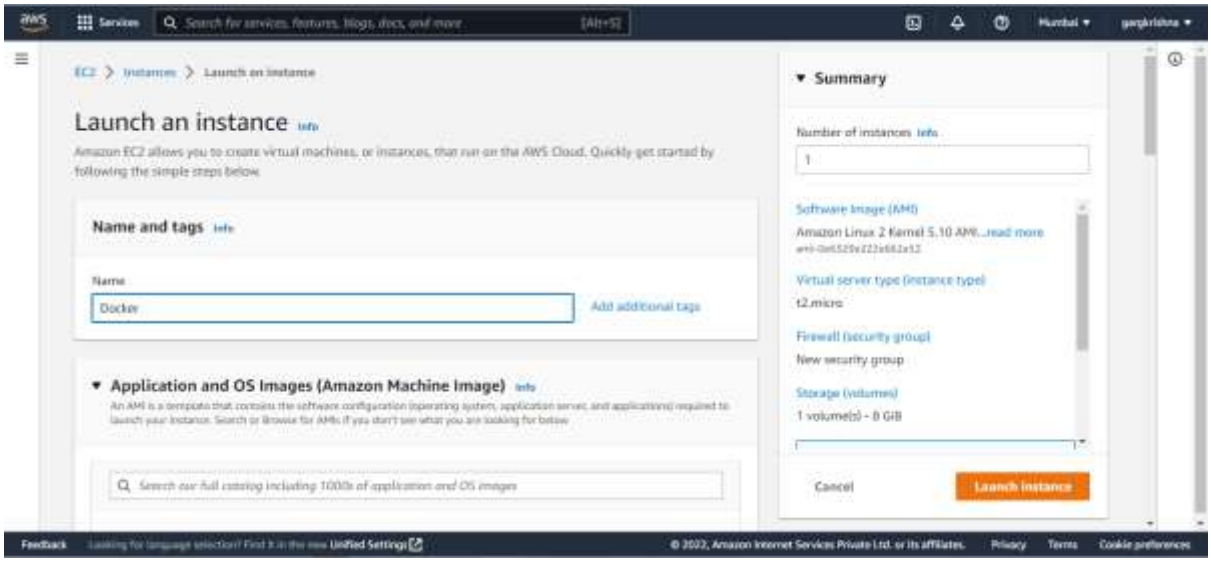
A docker file contains a set of instructions that are executed step by step when you use the docker build command to build the docker image. It contains certain instructions and commands that decide the structure of your image, the amount of time taken to build the image contains instructions related to the docker build context, contains information related to the packages and libraries to be installed in the container, and much more. Hence, it becomes very important to create an efficient, reusable, clean docker file as it contains the blueprint of the image that you will build.

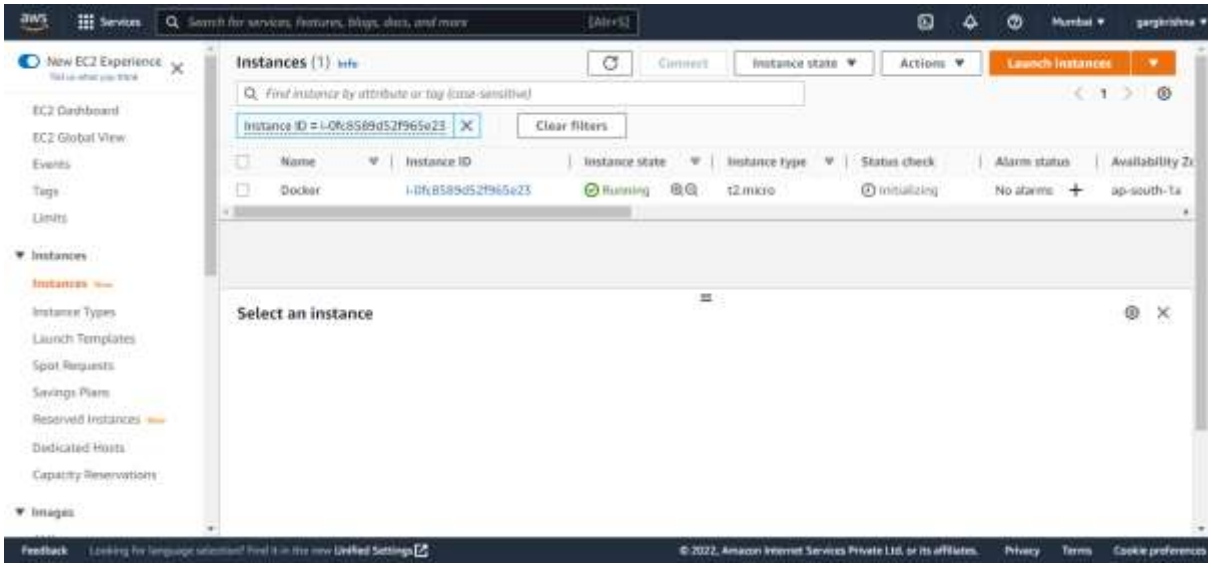
Some Docker file instructions:

- Run: A RUN instruction is used to run specified commands. You can use several RUN instructions to run different commands. But it is an efficient approach to combine all the RUN instructions into a single one. Each RUN command creates a new cache layer or an intermediate image layer and hence chaining all of them into a single line, becomes efficient. However, chaining multiple RUN instructions could lead to cache bursts as well.
- Pull: A pull instruction is used to pull an image from the daemon.
- Stop: A stop instruction is used to stop an image.

Findings and Requirements:

Launch an Instance





Connect the Instance:

```
login as: ec2-user
Authenticating with public key "Krish"

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-27-180 ~]$ sudo su
[root@ip-172-31-27-180 ec2-user]# yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No packages marked for update
[root@ip-172-31-27-180 ec2-user]# sudo amazon-linux-extras install docker -y
Installing docker
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Cleaning repos: amzn2-core amzn2extra-docker amzn2extra-kernel-5.10
17 metadata files removed
6 sqlite files removed
0 metadata files removed
```

```
  Installing : runc-1.1.3-1.amzn2.x86_64                                   1/5
  Installing : containerd-1.6.6-1.amzn2.x86_64                             2/5
  Installing : libcgroup-0.41-21.amzn2.x86_64                              3/5
  Installing : pigz-2.3.4-1.amzn2.0.1.x86_64                               4/5
  Installing : docker-20.10.17-1.amzn2.x86_64                              5/5
  Verifying  : docker-20.10.17-1.amzn2.x86_64                              1/5
  Verifying  : runc-1.1.3-1.amzn2.x86_64                                   2/5
  Verifying  : pigz-2.3.4-1.amzn2.0.1.x86_64                               3/5
  Verifying  : containerd-1.6.6-1.amzn2.x86_64                             4/5
  Verifying  : libcgroup-0.41-21.amzn2.x86_64                              5/5

Installed:
  docker.x86_64 0:20.10.17-1.amzn2

Dependency Installed:
  containerd.x86_64 0:1.6.6-1.amzn2       libcgroup.x86_64 0:0.41-21.amzn2
  pigz.x86_64 0:2.3.4-1.amzn2.0.1         runc.x86_64 0:1.1.3-1.amzn2

Complete!
```

```
[root@ip-172-31-27-180 ec2-user]# sudo service docker start
Redirecting to /bin/systemctl start docker.service
[root@ip-172-31-27-180 ec2-user]# service docker status
Redirecting to /bin/systemctl status docker.service
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; disabled; vendor pres
et: disabled)
   Active: active (running) since Tue 2022-10-04 08:34:38 UTC; 1min 50s ago
     Docs: https://docs.docker.com
  Process: 3470 ExecStartPre=/usr/libexec/docker/docker-setup-runtimes.sh (code=
exited, status=0/SUCCESS)
  Process: 3469 ExecStartPre=/bin/mkdir -p /run/docker (code=exited, status=0/SU
CCESS)
```

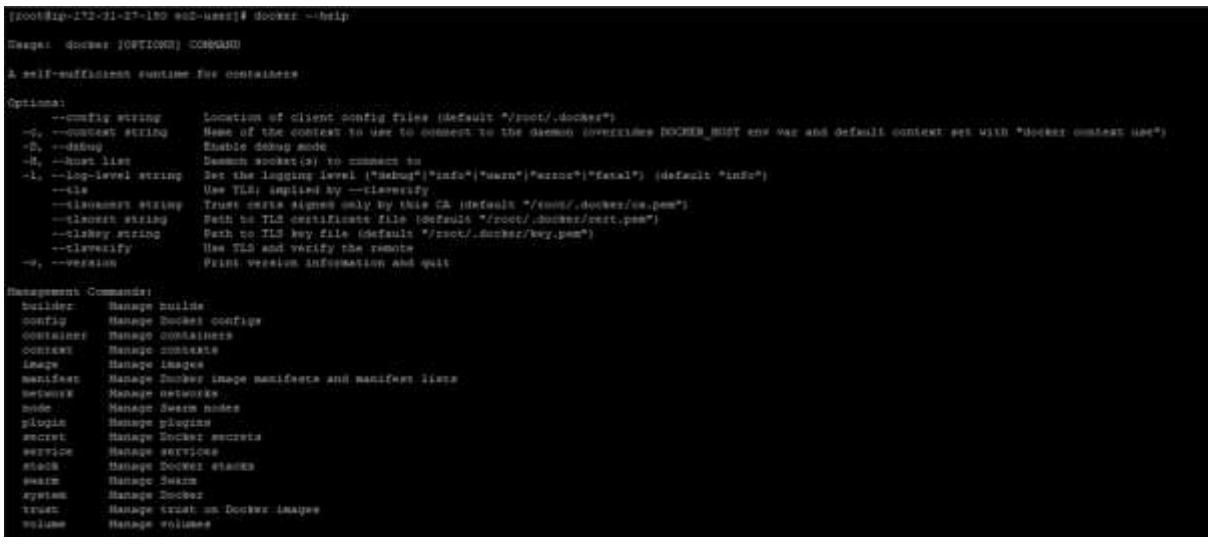
```
[root@ip-172-31-27-180 ec2-user]# usermod -a -G docker ec2-user
[root@ip-172-31-27-180 ec2-user]# systemctl enable docker
Created symlink from /etc/systemd/system/multi-user.target.wants/docker.service
to /usr/lib/systemd/system/docker.service.
[root@ip-172-31-27-180 ec2-user]# docker info
Client:
 Context:    default
 Debug Mode: false

Server:
 Containers: 0
  Running: 0
  Paused: 0
  Stopped: 0
 Images: 0
 Server Version: 20.10.17
 Storage Driver: overlay2
  Backing Filesystem: xfs
  Supports d_type: true
  Native Overlay Diff: true
  userxattr: false
 Logging Driver: json-file
 Cgroup Driver: cgroupfs
 Cgroup Version: 1
 Plugins:
  Volume: local
  Network: bridge host ipvlan macvlan null overlay
  Log: awslogs fluentd gcplogs gelf journald json-file local logentries splunk s
yslog
 Swarm: inactive
 Runtimes: io.containerd.runc.v2 io.containerd.runtime.v1.linux runc
 Default Runtime: runc
 Init Binary: docker-init
 containerd version: 10c12954828e7c7c9b6e0ea9b0c02b01407d3ae1
 runc version: 1e7bb5b773162b57333d57f612fd72e3f8612d94
 init version: de40ad0
 Security Options:
  seccomp
   Profile: default
 Kernel Version: 5.10.135-122.509.amzn2.x86_64
 Operating System: Amazon Linux 2
 OSType: linux
 Architecture: x86_64
 CPUs: 1
 Total Memory: 965.8MiB
 Name: ip-172-31-27-180.ec2.internal
 ID: 3H5V:7T4Q:5GOQ:YQJB:AENF:DOMT:2RCP:Y5JB:B7I6:WAGT:BFMD:3266
 Docker Root Dir: /var/lib/docker
 Debug Mode: false
 Registry: https://index.docker.io/v1/
 Labels:
```

```
[root@ip-172-31-27-180 ec2-user]# docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
2b55860d4c66: Pull complete
Digest: sha256:20fa2d7bb4de7723f542be5923b06c4d704370f0390e4ae9e1c833c8785644c1
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
[root@ip-172-31-27-180 ec2-user]# docker images
REPOSITORY   TAG       IMAGE ID       CREATED       SIZE
ubuntu       latest    2dc39ba059dc   4 weeks ago   77.8MB
[root@ip-172-31-27-180 ec2-user]# docker run -it -d ubuntu
23dacf9e505276eb362c3c04fd5b8eb380bc8e4327258410a1a8a5bfd321101b
[root@ip-172-31-27-180 ec2-user]# docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED          STATUS          PORTS     NA
MES
23dacf9e5052   ubuntu    "bash"    10 seconds ago   Up 10 seconds             be
autiful_northcutt
[root@ip-172-31-27-180 ec2-user]# docker exec -it 23dacf9e5052 bash
root@23dacf9e5052:/# ls
bin   dev  home  lib32  libx32  mnt  proc  run   srv  tmp  var
boot  etc  lib   lib64  media   opt  root  sbin  sys  usr
root@23dacf9e5052:/# mkdir docker
root@23dacf9e5052:/# ls
bin   dev     etc   lib    lib64   media  opt   root  sbin  sys  usr
boot  docker  home  lib32  libx32  mnt    proc  run   srv   tmp  var
```

```
[root@ip-172-31-27-180 ec2-user]# docker --version
Docker version 20.10.17, build 100c701
[root@ip-172-31-27-180 ec2-user]# docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't
have a Docker ID, head over to https://hub.docker.com to create one.
Username: ^C
[root@ip-172-31-27-180 ec2-user]# ^C
[root@ip-172-31-27-180 ec2-user]# docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:62af9efd515a25f84961b70f973a798d2eca956b1b2b026d0a4a63a3b0b6a3f2
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.
```



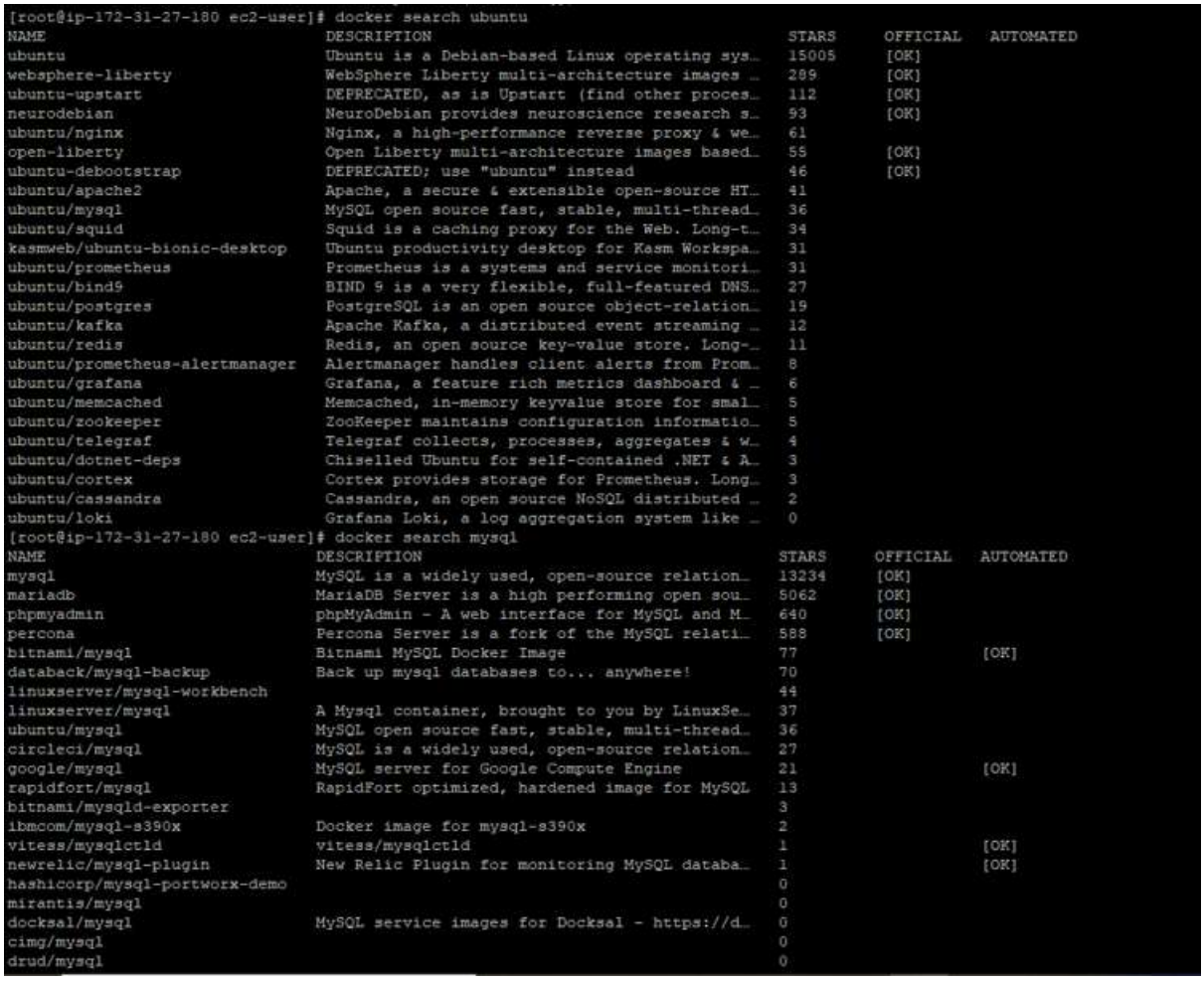
```
[root@ip-172-31-27-180 ec2-user]# docker pull alpine
Using default tag: latest
latest: Pulling from library/alpine
213ec9aee27d: Pull complete
Digest: sha256:bc41182d7ef5ffc53a40b044e725193bc10142a1243f395ee852a8d9730fc2ad
Status: Downloaded newer image for alpine:latest
docker.io/library/alpine:latest
[root@ip-172-31-27-180 ec2-user]# docker images
REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
ubuntu        latest    2dc39ba059dc   4 weeks ago     77.8MB
alpine        latest    9c6f07244728   7 weeks ago     5.54MB
hello-world   latest    feb5d9fea6a5   12 months ago   13.3kB
[root@ip-172-31-27-180 ec2-user]# docker ps -a
CONTAINER ID   IMAGE         COMMAND    CREATED          STATUS                      PORTS     NAMES
2ad226301c50   hello-world   "/hello"   4 minutes ago    Exited (0) 4 minutes ago              great_kepler
23dacf9e5052   ubuntu        "bash"     18 minutes ago   Up 18 minutes                         beautiful_northcutt
[root@ip-172-31-27-180 ec2-user]# docker run -it -d alpine
bfb7cfca22b5777bbbb13880e31ce460dae08d4e227f07fb006ec084df9e362f
[root@ip-172-31-27-180 ec2-user]# docker ps -a
CONTAINER ID   IMAGE         COMMAND     CREATED          STATUS                      PORTS     NAMES
```

```
[root@ip-172-31-27-180 ec2-user]# docker run -it -d alpine
bfb7cfca22b5777bbbb13880e31ce460dae08d4e227f07fb006ec084df9e362f
[root@ip-172-31-27-180 ec2-user]# docker ps -a
CONTAINER ID   IMAGE         COMMAND     CREATED          STATUS                      PORTS     NAMES
bfb7cfca22b5   alpine        "/bin/sh"   4 seconds ago    Up 3 seconds                          gracious_cannon
2ad226301c50   hello-world   "/hello"    6 minutes ago    Exited (0) 6 minutes ago              great_kepler
23dacf9e5052   ubuntu        "bash"      19 minutes ago   Up 19 minutes                         beautiful_northcutt
[root@ip-172-31-27-180 ec2-user]# docker container ls
CONTAINER ID   IMAGE     COMMAND     CREATED              STATUS              PORTS     NAMES
bfb7cfca22b5   alpine    "/bin/sh"   About a minute ago   Up About a minute             gracious_cannon
23dacf9e5052   ubuntu    "bash"      20 minutes ago       Up 20 minutes                 beautiful_northcutt
[root@ip-172-31-27-180 ec2-user]# docker build --help

Usage:  docker build [OPTIONS] PATH | URL | -

Build an image from a Dockerfile

Options:
      --add-host list           Add a custom host-to-IP mapping (host:ip)
      --build-arg list          Set build-time variables
      --cache-from strings      Images to consider as cache sources
      --cgroup-parent string    Optional parent cgroup for the container
      --compress                Compress the build context using gzip
      --cpu-period int          Limit the CPU CFS (Completely Fair Scheduler) period
      --cpu-quota int           Limit the CPU CFS (Completely Fair Scheduler) quota
  -c, --cpu-shares int          CPU shares (relative weight)
      --cpuset-cpus string      CPUs in which to allow execution (0-3, 0,1)
      --cpuset-mems string      MEMs in which to allow execution (0-3, 0,1)
      --disable-content-trust   Skip image verification (default true)
  -f, --file string             Name of the Dockerfile (Default is 'PATH/Dockerfile')
      --force-rm                Always remove intermediate containers
```



```
[root@ip-172-31-27-180 ec2-user]# docker search ubuntu
NAME                              DESCRIPTION                                     STARS     OFFICIAL   AUTOMATED
ubuntu                            Ubuntu is a Debian-based Linux operating sys…   15005     [OK]
websphere-liberty                 WebSphere Liberty multi-architecture images …   289       [OK]
ubuntu-upstart                    DEPRECATED, as is Upstart (find other proces…   112       [OK]
neurodebian                       NeuroDebian provides neuroscience research s…   93        [OK]
ubuntu/nginx                      Nginx, a high-performance reverse proxy & we…   61
open-liberty                      Open Liberty multi-architecture images based…   55        [OK]
ubuntu-debootstrap                DEPRECATED; use "ubuntu" instead                46        [OK]
ubuntu/apache2                    Apache, a secure & extensible open-source HT…   41
ubuntu/mysql                      MySQL open source fast, stable, multi-thread…   36
ubuntu/squid                      Squid is a caching proxy for the Web. Long-t…   34
kasmweb/ubuntu-bionic-desktop     Ubuntu productivity desktop for Kasm Workspa…   31
ubuntu/prometheus                 Prometheus is a systems and service monitori…   31
ubuntu/bind9                      BIND 9 is a very flexible, full-featured DNS…   27
ubuntu/postgres                   PostgreSQL is an open source object-relation…   19
ubuntu/kafka                      Apache Kafka, a distributed event streaming …   12
ubuntu/redis                      Redis, an open source key-value store. Long-…   11
ubuntu/prometheus-alertmanager    Alertmanager handles client alerts from Prom…   8
ubuntu/grafana                    Grafana, a feature rich metrics dashboard & …   6
ubuntu/memcached                  Memcached, in-memory keyvalue store for smal…   5
ubuntu/zookeeper                  ZooKeeper maintains configuration informatio…   5
ubuntu/telegraf                   Telegraf collects, processes, aggregates & w…   4
ubuntu/dotnet-deps                Chiselled Ubuntu for self-contained .NET & A…   3
ubuntu/cortex                     Cortex provides storage for Prometheus. Long…   3
ubuntu/cassandra                  Cassandra, an open source NoSQL distributed …   2
ubuntu/loki                       Grafana Loki, a log aggregation system like …   0
[root@ip-172-31-27-180 ec2-user]# docker search mysql
NAME                              DESCRIPTION                                     STARS     OFFICIAL   AUTOMATED
mysql                             MySQL is a widely used, open-source relation…   13234     [OK]
mariadb                           MariaDB Server is a high performing open sou…   5062      [OK]
phpmyadmin                        phpMyAdmin - A web interface for MySQL and M…   640       [OK]
percona                           Percona Server is a fork of the MySQL relati…   588       [OK]
bitnami/mysql                     Bitnami MySQL Docker Image                      77                   [OK]
databack/mysql-backup             Back up mysql databases to... anywhere!         70
linuxserver/mysql-workbench                                                       44
linuxserver/mysql                 A Mysql container, brought to you by LinuxSe…   37
ubuntu/mysql                      MySQL open source fast, stable, multi-thread…   36
circleci/mysql                    MySQL is a widely used, open-source relation…   27
google/mysql                      MySQL server for Google Compute Engine          21                   [OK]
rapidfort/mysql                   RapidFort optimized, hardened image for MySQL   13
bitnami/mysqld-exporter                                                           3
ibmcom/mysql-s390x                Docker image for mysql-s390x                    2
vitess/mysqlctld                  vitess/mysqlctld                                1                    [OK]
newrelic/mysql-plugin             New Relic Plugin for monitoring MySQL databa…   1                    [OK]
hashicorp/mysql-portworx-demo                                                     0
mirantis/mysql                                                                    0
docksal/mysql                     MySQL service images for Docksal - https://d…   0
cimg/mysql                                                                        0
drud/mysql                                                                        0
```

Conclusion:

In this experiment, docker file instructions were performed and a sample image for a web application was used and instructions like pull, start and stop were implemented.

We have achieved LO5 from this experiment.

We have also achieved Program Outcomes PO1, PO2, PO3, PO4, PO5, PO12.