Day 16 Task: Docker for DevOps Engineers:

Docker:

Docker is a platform and technology for building, shipping, and running distributed applications in containers. Containers are lightweight, portable, and self-sufficient executable packages that include all the necessary dependencies to run a piece of software. Docker allows developers to package their applications and dependencies into a single container, which can then be easily deployed and run on any host with Docker installed. This allows for consistent and predictable behavior across different environments and reduces the "works on my machine" problem.

Tasks:

1. **Use the docker run command to start a new container and interact with it through the command line. [Hint: docker run hello-world]**
   - **docker run** command is used to create a new container.

```
ubuntu@ip-172-31-80-127:~$ sudo docker run -d nginx
3efb9ef55f33b5abc44703066c9f3e530e128c89273510a42be960e2367d4a5a
ubuntu@ip-172-31-80-127:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                  NAMES
3efb9ef55f33   nginx     "/docker-entrypoint.…"   21 seconds ago   Up 21 seconds   80/tcp                 cool_bhabha
124570dcc75f   mysql     "docker-entrypoint.s…"   11 minutes ago   Up 11 minutes   3306/tcp, 33060/tcp    some-mysql
ubuntu@ip-172-31-80-127:~$
```

   - To create a container with container name:

```
ubuntu@ip-172-31-80-127:~$ sudo docker run -d --name nginx_server nginx
0cc8706de526a9070fbf7e2853723cb439cf5eb7c24b1283f93eccf4b10ded35
ubuntu@ip-172-31-80-127:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                                   NAMES
0cc8706de526   nginx     "/docker-entrypoint.…"   9 seconds ago    Up 8 seconds    80/tcp                                  nginx_server
616294c5a9f1   nginx     "/docker-entrypoint.…"   10 minutes ago   Up 10 minutes   0.0.0.0:8000->80/tcp, :::8000->80/tcp   musing_galileo
3efb9ef55f33   nginx     "/docker-entrypoint.…"   16 minutes ago   Up 16 minutes   80/tcp                                  cool_bhabha
124570dcc75f   mysql     "docker-entrypoint.s…"   27 minutes ago   Up 27 minutes   3306/tcp, 33060/tcp                     some-mysql
ubuntu@ip-172-31-80-127:~$
```

   - To create a container with port number:

```
ubuntu@ip-172-31-80-127:~$ sudo docker run -d -p 8000:80 nginx
616294c5a9f10c5f06714a477a9f60b2986d852fb2c61efbced85dc258bd4006
ubuntu@ip-172-31-80-127:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                                   NAMES
616294c5a9f1   nginx     "/docker-entrypoint.…"   22 seconds ago   Up 21 seconds   0.0.0.0:8000->80/tcp, :::8000->80/tcp   musing_galileo
3efb9ef55f33   nginx     "/docker-entrypoint.…"   6 minutes ago    Up 6 minutes    80/tcp                                  cool_bhabha
124570dcc75f   mysql     "docker-entrypoint.s…"   17 minutes ago   Up 17 minutes   3306/tcp, 33060/tcp                     some-mysql
ubuntu@ip-172-31-80-127:~$
```

2. **Use the docker inspect command to view detailed information about a container or image.**

The “docker inspect” command allows you to view detailed information about a container. This information includes the container’s configuration, network setting, and current state. To use the command, you need to specify the container ID or name as an argument.

```
ubuntu@ip-172-31-80-127:~$ sudo docker inspect nginx
[
    {
        "Id": "sha256:a99a39d070bfd1cb60fe65c45dea3a33764dc00a9546bf8dc46cb5a11b1b50e9",
        "RepoTags": [
            "nginx:latest"
        ],
        "RepoDigests": [
            "nginx@sha256:b8f2383a95879e1ae064940d9a200f67a6c79e710ed82ac42263397367e7cc4e"
        ],
        "Parent": "",
        "Comment": "",
        "Created": "2023-01-11T06:31:09.511615468Z",
        "Container": "e9ec36a6bb8cd0d7927c7e91b304ab1948fe4dd4c2ef22fb2fbc0ac3bed8f2e6",
        "ContainerConfig": {
            "Hostname": "e9ec36a6bb8c",
            "Domainname": "",
            "User": "",
            "AttachStdin": false,
            "AttachStdout": false,
            "AttachStderr": false,
            "ExposedPorts": {
                "80/tcp": {}
            },
```

3. Use the docker port command to list the port mappings for a container.

The “docker port” command allows you to list the port mappings for a container. To use the command, you need to specify the ID or name as an argument.

```
ubuntu@ip-172-31-80-127:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                                   NAMES
0cc8706de526   nginx     "/docker-entrypoint.…"   8 minutes ago    Up 8 minutes    80/tcp                                  nginx_server
616294c5a9f1   nginx     "/docker-entrypoint.…"   18 minutes ago   Up 18 minutes   0.0.0.0:8000->80/tcp, :::8000->80/tcp   musing_galileo
3efb9ef55f33   nginx     "/docker-entrypoint.…"   24 minutes ago   Up 24 minutes   80/tcp                                  cool_bhabha
124570dcc75f   mysql     "docker-entrypoint.s…"   35 minutes ago   Up 35 minutes   3306/tcp, 33060/tcp                     some-mysql
ubuntu@ip-172-31-80-127:~$ sudo docker port musing_galileo
80/tcp -> 0.0.0.0:8000
80/tcp -> :::8000
ubuntu@ip-172-31-80-127:~$
```

4. Use the docker stats command to view resource usage statistics for one or more containers.

The “docker stats” command allows you to view resources usage statistics for one or more running containers. The command shows a live stream of resources usage statistics for the specified containers, including CPU, memory, network, and storage usage. By default, “docker stats” show statistics for all running containers.

```
ubuntu@ip-172-31-80-127: ~
CONTAINER ID   NAME             CPU %     MEM USAGE / LIMIT     MEM %     NET I/O       BLOCK I/O         PIDS
0cc8706de526   nginx_server     0.00%     2.727MiB / 966.2MiB   0.28%     1.01kB / 0B   1.89MB / 12.3kB   2
616294c5a9f1   musing_galileo   0.00%     2.098MiB / 966.2MiB   0.22%     1.08kB / 0B   217kB / 12.3kB    2
3efb9ef55f33   cool_bhabha      0.00%     2.582MiB / 966.2MiB   0.27%     1.08kB / 0B   1.79MB / 12.3kB   2
124570dcc75f   some-mysql       0.43%     352.2MiB / 966.2MiB   36.46%    1.37kB / 0B   121MB / 275MB     38
```

5. Use the docker top command to view the processes running inside a container.

The “docker top” command allows you to view the processes running inside a container. It is similar to the “top” command on linux and show the same information, such as process ID, user, CPU usage, and memory usage. To use the command, you need to specify the container ID or name as an argument.

```
ubuntu@ip-172-31-80-127:~$ sudo docker top nginx_server
UID                 PID                 PPID                C                   STIME               TTY                 TIME                CMD
root                4310                4288                0                   09:24               ?                   00:00:00            ngin
x: master process nginx -g daemon off;
systemd+            4361                4310                0                   09:24               ?                   00:00:00            ngin
x: worker process
ubuntu@ip-172-31-80-127:~$
```

6. Use the docker save command to save an image to a tar archive.

The “docker save” command allows you to save an image to a tar archive. This is useful for creating backups of images or for transferring images between different system. The command takes the image name or ID as an argument and save the image to a tar archive.

```
ubuntu@ip-172-31-80-127:~$ sudo docker save nginx > nginx_image.tar
ubuntu@ip-172-31-80-127:~$ ls -sh
total 140M
140M nginx_image.tar  4.0K snap
ubuntu@ip-172-31-80-127:~$
```

7. Use the docker load command to load an image from a tar archive.

The “docker load” command allows you to load an image from tar archive. This is useful for restoring backups of images or for transferring images between different system. The command takes the file name of the tar archive as an argument and loads the image(S) into the local image store.

```
ubuntu@ip-172-31-80-127:~$
ubuntu@ip-172-31-80-127:~$ sudo docker load < nginx_image.tar
Loaded image: nginx:latest
ubuntu@ip-172-31-80-127:~$
```

Thank you for reading! I hope you find this article helpful.

Happy Learning 😊