Assignment-1

Demonstrate minimum 15 basic docker command with explanation and screenshot.

- 1. To build an image from the dockerfile in the current directory
 - Docker build -t <image_name> .

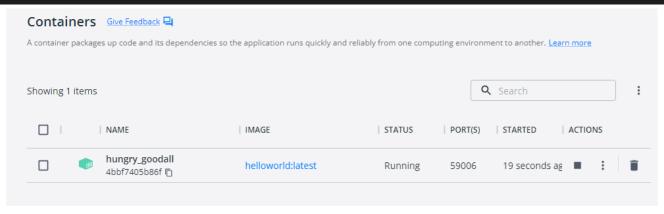
2. To check the list docker images

Docker images

```
(base) E:\Industru Project\Docker\Day04-Docker\Hello World>docker images
REPOSITORY
                                    IMAGE ID
                          TAG
                                                   CREATED
                                                                    SIZE
helloworld
                          latest
                                    fa3af6094ff1
                                                   2 hours ago
                                                                    1.06GB
ubuntu
                          latest
                                    216c552ea5ba
                                                                    77.8MB
                                                   8 days ago
krishnaik06/welcome-app
                          latest
                                    ab38a6cb29ad
                                                   2 months ago
                                                                    58.2MB
docker/getting-started
                          latest
                                    cb90f98fd791
                                                                    28.8MB
                                                   6 months ago
hello-world
                          latest
                                    feb5d9fea6a5
                                                   12 months ago
                                                                    13.3kB
```

- To run the container
 - Docker run –d –p <host port>-<container port> <image_name>
 - → detached mode
 - > -p→ port

(base) E:\Industru Project\Docker\Day04-Docker\Hello World>docker run -d -p 5000-5000 helloworld 4bbf7405b86f65c14ca1d9c89186254f88127efc3c74cccaa7419b5eb2397251



4. To pull an image or a repository from a registry

docker pull <image name: tag[latest]>

```
(base) E:\Industru Project\Docker\Day04-Docker\Hello World>docker pull python:3.7.15-alpine3.16
3.7.15-alpine3.16: Pulling from library/python
213ec9aee27d: Pull complete
47858aee13bf: Pull complete
21d1eb95d787: Pull complete
17ac4101fac6: Pull complete
131ff38582db: Pull complete
Digest: sha256:45438a907ad35872739d31a5ae28a72a79c3f6abb43ec0b32688216e1ea1759c
Status: Downloaded newer image for python:3.7.15-alpine3.16
docker.io/library/python:3.7.15-alpine3.16
```

5. To check the docker container active process running

- docker ps
- docker ps –a
 - -a all container (stopped and active)

```
(base) E:\Industru Project\Docker\Day04-Docker\Hello World>docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

2c6e60b81ac0 python:3.7.15-alpine3.16 "python3" 9 seconds ago Up 3 seconds elegant_robinson
```

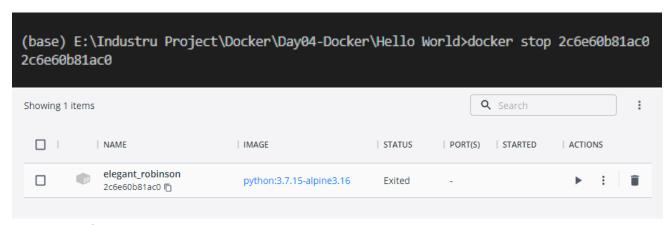
To start the container

Docker start <id/name>

(base) E:\Industru Project\Docker\Day04-Docker\Hello World>docker start 36a43f7ae09f46522fac1589ad704b62df09cb2dfad852f9df9a04318bfb2be6 36a43f7ae09f46522fac1589ad704b62df09cb2dfad852f9df9a04318bfb2be6

7. To stop the container

Docker stop <id/name>



8. To remove the container

docker rm /<ID/container_name>

(base) E:\Industru Project\Docker\Day04-Docker\Hello World>docker rm /2c6e60b81ac0 /2c6e60b81ac0

9. To remove the image

- docker rmi /<ID/image name>

```
(base) E:\Industru Project\Docker\Day04-Docker\Hello World>docker rmi /helloworld Untagged: helloworld:latest Deleted: sha256:fa3af6094ff1b07762e02d4224743c03bd4d77bd09d54feb064fe7fe51976fd5
```

10. To push the docker image

Docker push <username/imagename>

```
$ docker push maheshkumarvb/hello-world
Using default tag: latest
The push refers to repository [docker.io/maheshkumarvb/hello-world]
84024b1fbbe4: Pushed
ffba3295bab2: Pushed
08f7737fec66: Mounted from library/python
dee03037c4fc: Mounted from library/python
17517a754285: Mounted from library/python
0c7daf9a72c8: Mounted from library/python
75ba02937496: Mounted from library/python
288cf3a46e32: Mounted from library/python
```

11. To provide detailed information on constructs controlled by Docker.

docker inspect <id/name>

```
(base) E:\Industru Project\Docker\Day04-Docker\Hello World>docker inspect python:3.7.15-alpine3.16
        "Id": "sha256:dedec9fb414999ece43a424380467f77c07c6a9fb83f2c85d3131519ffc9c4d3",
        "RepoTags": [
            "python:3.7.15-alpine3.16"
        "RepoDigests": [
            "python@sha256:45438a907ad35872739d31a5ae28a72a79c3f6abb43ec0b32688216e1ea1759c"
        "Parent": "",
"Comment": "",
        "Created": "2022-10-11T21:11:10.728094317Z",
        "Container": "55b690704721eff15ae4989c29c8d6cc0ddff7e6b75803de5c83f9796c13ec30",
        "ContainerConfig": {
"Hostname": "55b690704721",
            "Hostname": "55b6
"Domainname": "",
            "User": "",
            "AttachStdin": false,
            "AttachStdout": false,
            "AttachStderr": false,
            "Tty": false,
            "OpenStdin": false,
            "StdinOnce": false,
            "Env": [
                 "PATH=/usr/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",
                "LANG=C.UTF-8",
                "GPG_KEY=0D96DF4D4110E5C43FBFB17F2D347EA6AA65421D",
                "PYTHON_VERSION=3.7.15",
```

12. To Fetch the logs of a container

docker logs <id/name>

```
(base) E:\Industru Project\Docker\Docker\Docker\Hello World>docker logs 36a43f7ae09f
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2022/10/13 12:56:49 [notice] 1#1: using the "epoll" event method
2022/10/13 12:56:49 [notice] 1#1: pginx/1.21.6
2022/10/13 12:56:49 [notice] 1#1: built by gcc 10.3.1 20211027 (Alpine 10.3.1_git20211027)
2022/10/13 12:56:49 [notice] 1#1: St. Linux 5.10.102.1-microsoft-standard-WSL2
2022/10/13 12:56:49 [notice] 1#1: start worker processes
2022/10/13 12:56:49 [notice] 1#1: start worker processes
2022/10/13 12:56:49 [notice] 1#1: start worker process 32
2022/10/13 12:56:49 [notice] 1#1: start worker process 34
2022/10/13 12:56:49 [notice] 1#1: start worker process 35
```

13. To kill the container

Docker kill <sha/name>

(base) E:\Industru Project\Docker\Day04-Docker\Hello World>docker kill ff4dea55eeb7f363d938302a23ac26bf4080e2c64f3a7118dd1ef5a503a9c03a ff4dea55eeb7f363d938302a23ac26bf4080e2c64f3a7118dd1ef5a503a9c03a

(base) E:\Industru Project\Docker\Day04-Docker\Hello World>docker kill bold_hellman bold hellman

14. To Start a container in the background.

docker wait <containername>

15. To check the docker version

- docker -version

```
docker version
Client:
Version:
                    20.10.17
API version:
                    1.41
Go version:
                    qo1.17.11
Git commit:
                    100c701
Built:
                    Mon Jun 6 22:56:42 2022
OS/Arch:
                    linux/amd64
Context:
                    default
Experimental:
                    true
```

Assignment 2

Run Hello World Docker Image Locally

(base) E:\Industru Project\Docker\Day04-Docker\Hello World>docker pull hello-world

Using default tag: latest

latest: Pulling from library/hello-world

Digest: sha256:18a657d0cc1c7d0678a3fbea8b7eb4918bba25968d3e1b0adebfa71caddbc346

Status: Image is up to date for hello-world:latest

docker.io/library/hello-world:latest

(base) E:\Industru Project\Docker\Day04-Docker\Hello World>docker run hello-world

Hello from Docker!

This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

- The Docker client contacted the Docker daemon.
- The Docker daemon pulled the "hello-world" image from the Docker Hub. (amd64)
- 3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
- 4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

To try something more ambitious, you can run an Ubuntu container with:

\$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID: https://hub.docker.com/

For more examples and ideas, visit: https://docs.docker.com/get-started/