Graded Assignment on Docker

Q1) Pull any image from the docker hub, create its container, and execute it showing the output.

Docker is a software platform to create, test and deploy applications in an isolated environment. Docker uses container to package up an application with all of the parts it needs including, libraries and dependencies. It allows applications to use the kernel and other resources of the host operating system this will boost the performance and reduce the size of the application.Docker Hub is a centralized repository service that allows you to store container images and share them with your team. You can use Pull and Push command to upload and download images to and from the Docker Hub.

*Give the docker version command.

```
C:\Users\Bhanu>docker version
Client:
 Cloud integration: v1.0.29
 Version:           20.10.22
 API version:       1.41
 Go version:        go1.18.9
 Git commit:        3a2c30b
 Built:             Thu Dec 15 22:36:18 2022
 OS/Arch:           windows/amd64
 Context:           default
 Experimental:      true

Server: Docker Desktop 4.16.3 (96739)
 Engine:
  Version:          20.10.22
  API version:      1.41 (minimum version 1.12)
  Go version:       go1.18.9
  Git commit:       42c8b31
  Built:            Thu Dec 15 22:26:14 2022
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          1.6.14
  GitCommit:        9ba4b250366a5ddde94bb7c9d1def331423aa323
 runc:
  Version:          1.1.4
  GitCommit:        v1.1.4-0-g5fd4c4d
 docker-init:
  Version:          0.19.0
  GitCommit:        de40ad0

C:\Users\Bhanu>
```

docker images:

it is the command which is used to list out all the images in the docker hub.

```
C:\Users\Bhanu>docker images
REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
java-app      latest    d0afcd40d190   22 hours ago    526MB
<none>        <none>    db8bfe986dfe   22 hours ago    526MB
<none>        <none>    29d58b2e8846   22 hours ago    526MB
<none>        <none>    84a79db5e8eb   22 hours ago    526MB
<none>        <none>    b43b0a523465   31 hours ago    526MB
nginx         latest    3f8a00f137a0   9 days ago      142MB
ubuntu        latest    58db3edaf2be   3 weeks ago     77.8MB
hello-world   latest    feb5d9fea6a5   17 months ago   13.3kB
```

step1:

we can pull the image from the docker hub using the docker pull imagename .

Let us download the image called hello-world from the docker hub.

Once the hello-world image is downloaded,we get the following output.

```
C:\Users\Bhanu>docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
Digest: sha256:6e8b6f026e0b9c419ea0fd02d3905dd0952ad1feea67543f525c73a0a790fefb
Status: Image is up to date for hello-world:latest
docker.io/library/hello-world:latest
```

Step2:

To create a container,the command is docker create image-name.

→docker create image-name

```
C:\Users\Bhanu>docker create hello-world
aaa188388a437d3a110256369ed4274b286033e8a0b9d40943da4ed2a448e818
```

Step3: To run the container and shows output.

This command is used to tell the docker to run.

```
C:\Users\Bhanu>docker start -a aaa188388a437d3a110256369ed4274b286033e8a0b9d40943da4ed2a448e818

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

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. 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/
```

Step4:

we can also verify the hello-world container with the below command.

docker ps

we will get the following output.

```
C:\Users\Bhanu>docker ps --all
CONTAINER ID   IMAGE         COMMAND               CREATED          STATUS                      PORTS               NAMES
d93031220330   hello-world   "/hello"              5 minutes ago    Exited (0) 5 minutes ago                        sad_wu
```

it will show the container-id.

(or)

Docker run img-name

This command is used to create the image ,pull the image and run the image at the same time.

```
C:\Users\Bhanu>docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:6e8b6f026e0b9c419ea0fd02d3905dd0952ad1feea67543f525c73a0a790fefb
Status: Downloaded newer image for hello-world:latest

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

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. 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/
```

**Creating another container nginx.
The command used to pull the nginx image is
Docker pull nginx

```
C:\Users\Bhanu>docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
bb263680fed1: Pull complete
258f176fd226: Pull complete
a0bc35e70773: Pull complete
077b9569ff86: Pull complete
3082a16f3b61: Pull complete
7e9b29976cce: Pull complete
Digest: sha256:6650513efd1d27c1f8a5351cbd33edf85cc7e0d9d0fcb4ffb23d8fa89b601ba8
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
```

*Once the docker image is pull,now you are ready to run it.
The run command does the work of docker create ,docker start.

```
C:\Users\Bhanu>docker run --name docker-nginx -p 80:80 -d nginx
c0cc4b2156cef0cf695ac6f0d0dcad228d92c6a23f5780754a08f63cfb60bff3
```

*docker ps shows the available containers,the time when it is created,its status and names.

```
C:\Users\Bhanu>docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS         PORTS                NAMES
c0cc4b2156ce   nginx     "/docker-entrypoint.…"   2 minutes ago   Up 2 minutes   0.0.0.0:80->80/tcp   docker-nginx
```

```
C:\Users\Bhanu>docker exec -it docker-nginx /bin/bash
root@c0cc4b2156ce:/# apt update
Get:1 http://deb.debian.org/debian bullseye InRelease [116 kB]
Get:2 http://deb.debian.org/debian-security bullseye-security InRelease [48.4 kB]
Get:3 http://deb.debian.org/debian bullseye-updates InRelease [44.1 kB]
Get:4 http://deb.debian.org/debian bullseye/main amd64 Packages [8183 kB]
Get:5 http://deb.debian.org/debian-security bullseye-security/main amd64 Packages [226 kB]
Get:6 http://deb.debian.org/debian bullseye-updates/main amd64 Packages [14.6 kB]
Fetched 8632 kB in 14min 33s (9884 B/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
1 package can be upgraded. Run 'apt list --upgradable' to see it.
root@c0cc4b2156ce:/#
```

Q2) Create the basic java application, generate its image with necessary files, and execute it with docker.

Creating the basic java application.

Step1:Create a directory,it is used to store the files.

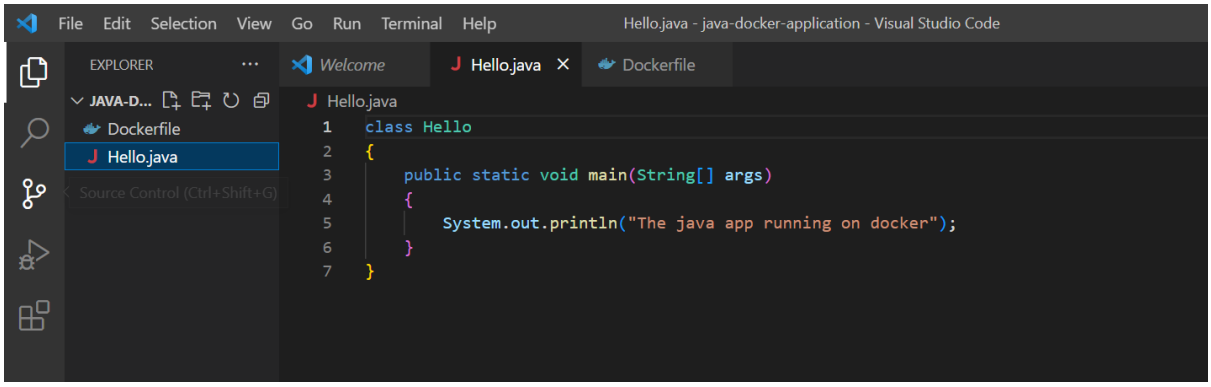
```
C:\Users\Bhanu>mkdir java-docker-application
```

Step2:go to the directory that you have created.

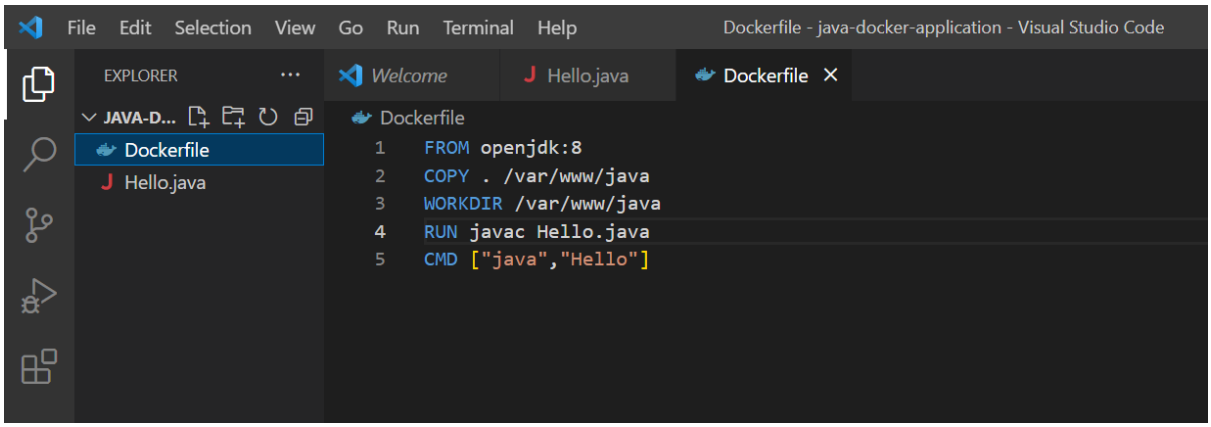
```
C:\Users\Bhanu>cd java-docker-application

C:\Users\Bhanu\java-docker-application>code .
```

Step3:Create a java file,save it as Hello.java



Step4:Create a docker file.



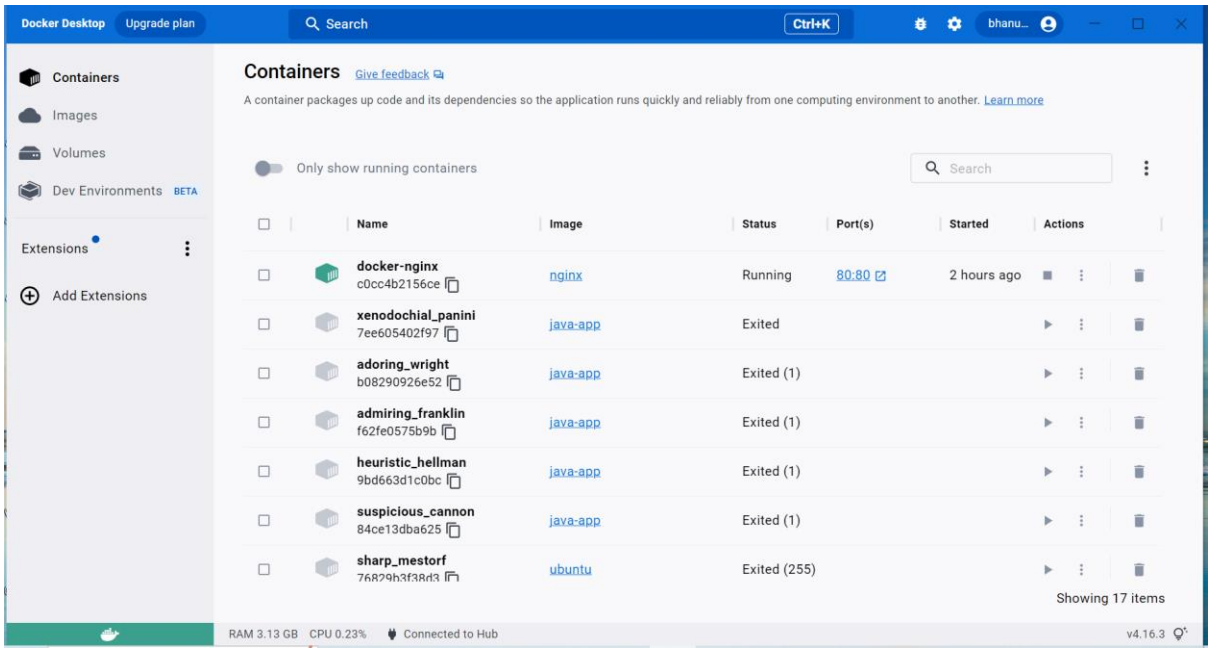
Step5:Now create an image by following below command.we must login as root in order to create a image.In the following command ,java-app is name of the image.We can have any name for our docker image.

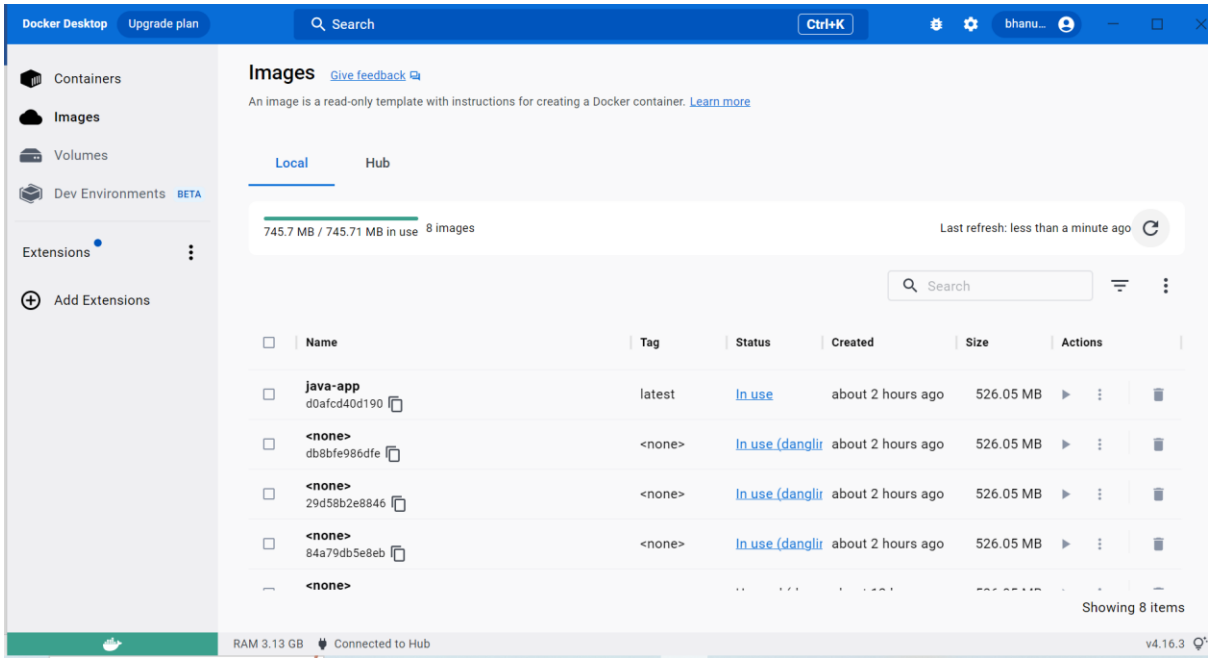
```
C:\Users\Bhanu\java-docker-application>docker build -t java-app .
[+] Building 28.8s (9/9) FINISHED
 => [internal] load build definition from Dockerfile                                                                0.8s
 => => transferring dockerfile: 140B                                                                                0.0s
 => [internal] load .dockerignore                                                                                   1.4s
 => => transferring context: 2B                                                                                     0.0s
 => [internal] load metadata for docker.io/library/openjdk:8                                                       11.5s
 => [internal] load build context                                                                                   0.6s
 => => transferring context: 319B                                                                                   0.0s
 => CACHED [1/4] FROM docker.io/library/openjdk:8@sha256:86e863cc57215cfb181bd319736d0baf625fe8f150577f9eb58bd937   0.0s
 => [2/4] COPY . /var/www/java                                                                                      1.6s
 => [3/4] WORKDIR /var/www/java                                                                                     1.9s
 => [4/4] RUN javac Hello.java                                                                                      4.9s
 => exporting to image                                                                                              4.5s
 => => exporting layers                                                                                             3.5s
 => => writing image sha256:d0afcd40d1900f6065c0ca7cb21459301f7eb3fd4b15755acb1572c58d519dce                        0.1s
 => => naming to docker.io/library/java-app                                                                         0.1s
```

Step6:After successfully building the image,now we can run docker by using run command.

```
C:\Users\Bhanu\java-docker-application>docker run java-app
The java app running on docker
```

*Open docker desktop and you can see that the java application si running.




Docker Desktop
Upgrade plan
Search
Ctrl+K
bhanu...
Containers
Images
Volumes
Dev Environments
BETA
Extensions
Add Extensions
Images
Give feedback
An image is a read-only template with instructions for creating a Docker container. Learn more
Local
Hub
745.7 MB / 745.71 MB in use
8 images
Last refresh: less than a minute ago
Search
Name
Tag
Status
Created
Size
Actions
java-app
d0afcd40d190
latest
In use
about 2 hours ago
526.05 MB
<none>
db8bfe986dfe
<none>
In use (danglir
about 2 hours ago
526.05 MB
<none>
29d58b2e8846
<none>
In use (danglir
about 2 hours ago
526.05 MB
<none>
84a79db5e8eb
<none>
In use (danglir
about 2 hours ago
526.05 MB
<none>
Showing 8 items
RAM 3.13 GB
Connected to Hub
v4.16.3