

# Task\_1

- Run the container hello-world

```
docker run hello-world
```

- Check the container status

```
docker ps -a
```

- Start the stopped container

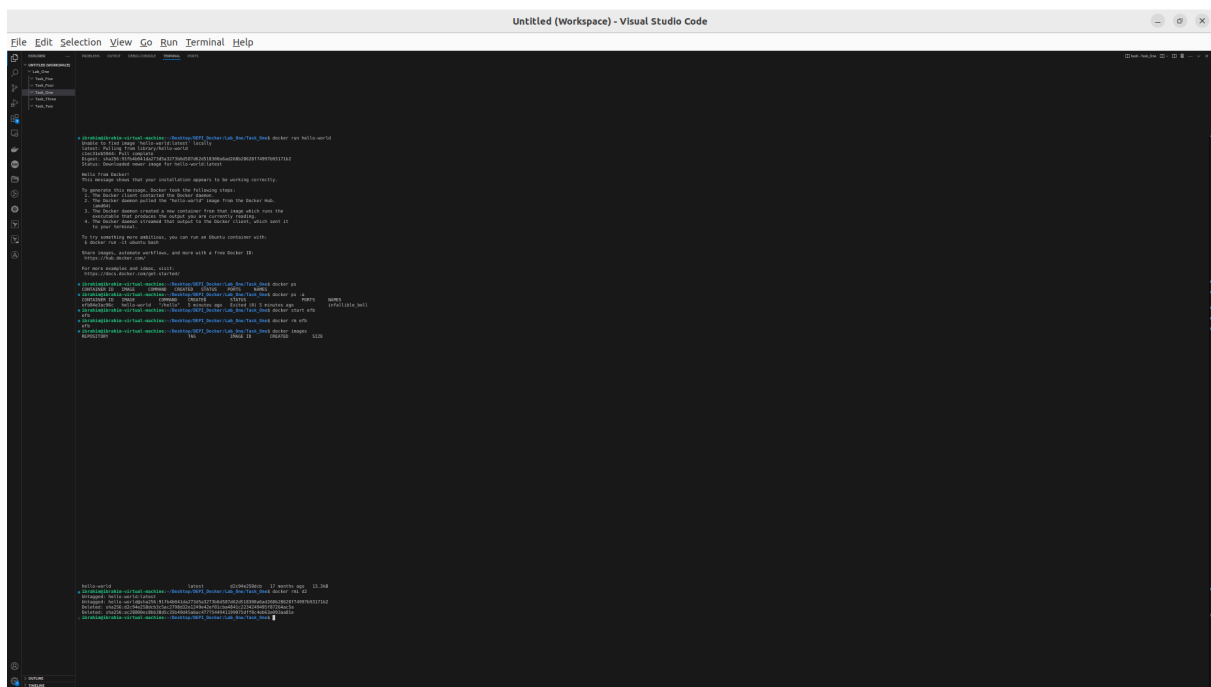
```
docker start <container_id>
```

- Remove the container

```
docker rm <container_id>
```

- Remove the image

```
docker rmi <image_id>
```



The screenshot shows a Visual Studio Code window with a terminal open. The terminal output displays the following commands and their results:

```
root@kali:~# docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
024f20fae722: Pull complete
Digest: sha256:325b46964486bd68b4fa172953fbd945437680c12541674c0be1f5e7923711721
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.
3. The daemon created a new container from that image which runs the
   command that displays the output of the command that was run.
4. The 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://cloud.docker.com

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

root@kali:~# docker ps -a
CONTAINER ID   IMAGE                                COMMAND                  STATUS    PORTS       NAMES
d9a0c2b2c1e1   hello-world:latest                  "/bin/sh -c 'echo hello world'"  exited 0  0.0.0.0:2375->127.0.0.1:2375   hello-world

root@kali:~# docker start d9a0c2b2c1e1
d9a0c2b2c1e1

root@kali:~# docker rm d9a0c2b2c1e1
d9a0c2b2c1e1

root@kali:~# docker rmi hello-world:latest
hello-world:latest: Removing image...
```