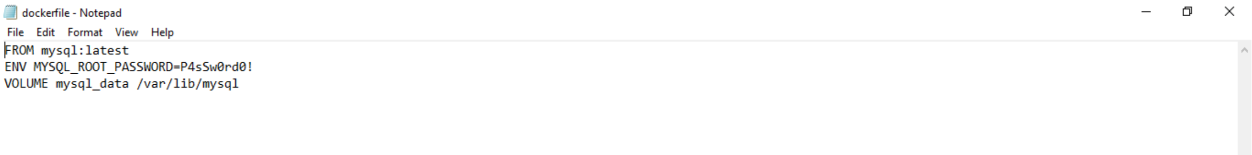
Name : Reem Ahmed mansour

Lab 2

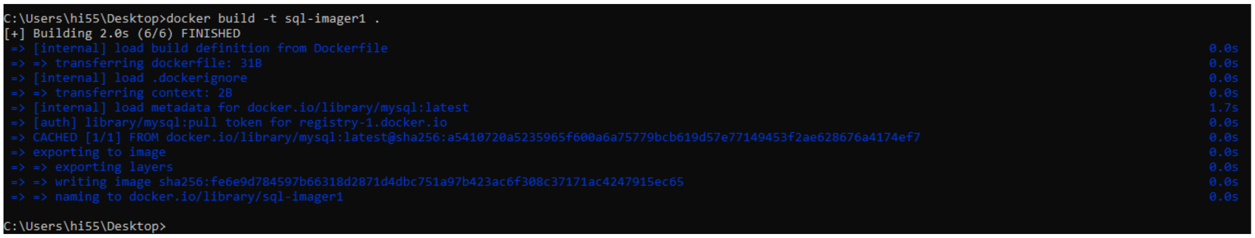
- P2: Create MySQL database docker image with:

volume called mysql_data to /var/lib/mysql

Set ENV variable MYSQL_ROOT_PASSWORD to P4sSw0rd0!.



```
FROM mysql:latest
ENV MYSQL_ROOT_PASSWORD=P4sSw0rd0!
VOLUME mysql_data /var/lib/mysql
```



```
C:\Users\hi55\Desktop>docker build -t sql-imager1 .
[+] Building 2.0s (6/6) FINISHED
 => [internal] load build definition from Dockerfile                                                                  0.0s
 => => transferring dockerfile: 31B                                                                                   0.0s
 => [internal] load .dockerignore                                                                                     0.0s
 => => transferring context: 2B                                                                                       0.0s
 => [internal] load metadata for docker.io/library/mysql:latest                                                       1.7s
 => [auth] library/mysql:pull token for registry-1.docker.io                                                          0.0s
 => CACHED [1/1] FROM docker.io/library/mysql:latest@sha256:a5410720a5235965f600a6a75779bcb619d57e77149453f2ae628676a4174ef7  0.0s
 => exporting to image                                                                                                0.0s
 => => exporting layers                                                                                               0.0s
 => => writing image sha256:fe6e9d784597b66318d2871d4dbc751a97b423ac6f308c37171ac4247915ec65                          0.0s
 => => naming to docker.io/library/sql-imager1                                                                        0.0s

C:\Users\hi55\Desktop>
```

P3: What is the rest of Docker Networks ? "Name and Definition"

Network drivers - Docker supports different network drivers, such as bridge, host, overlay, and macvlan, which provide different networking options for containers.

1-Bridge network - This is the default network driver in Docker. It creates a virtual network bridge that allows containers to communicate with each other and with the host machine.

2-Host network - This network driver allows a container to use the host machine's network stack directly, without any network isolation.

3-Overlay network - This network driver enables multi-host networking for Docker Swarm services. It creates an overlay network that spans multiple Docker hosts, allowing containers to communicate with each other seamlessly.

4-Macvlan network - This network driver allows a container to be directly connected to a physical network interface on the host machine, providing better performance and network isolation.

5-Network namespaces - Docker uses network namespaces to provide network isolation for containers. Each container has its own network namespace, which allows it to have its own network interfaces, IP addresses, routing tables, and firewall rules.

P4: What is the different between docker images tags ?

1-Version: Docker image tags often include a version number, such as "1.0" or "2.3.4". This allows you to track changes and updates to the image over time.

2-Variation: Image tags can also be used to differentiate between variations of an image. For example, you might have a "nginx" image that has different tags for different versions or configurations, such as "nginx:1.18-alpine" or "nginx:1.18-ubuntu".

3-Stability: Docker image tags can indicate the stability of an image, such as "latest" or "stable". However, it's important to note that the "latest" tag doesn't necessarily mean the most stable version of an image, as it always refers to the most recently built image.

- alpha: An alpha tag typically represents an early stage of development, where the image is still in the testing phase and may contain bugs or incomplete features. Images with alpha tags should not be used in production environments.

- beta: A beta tag typically represents a later stage of development, where the image is more stable but may still contain some bugs or incomplete features. Images with beta tags may be suitable for testing and development environments, but should still be used with caution.

- stable: A stable tag typically represents a fully tested and production-ready image. Images with stable tags have been thoroughly tested and are considered to be reliable and bug-free. These images are suitable for production environments

4-Repository: Docker image tags can also be used to indicate the repository where the image is stored, such as "docker.io/nginx:1.18". This can be useful if you're using images from multiple repositories.