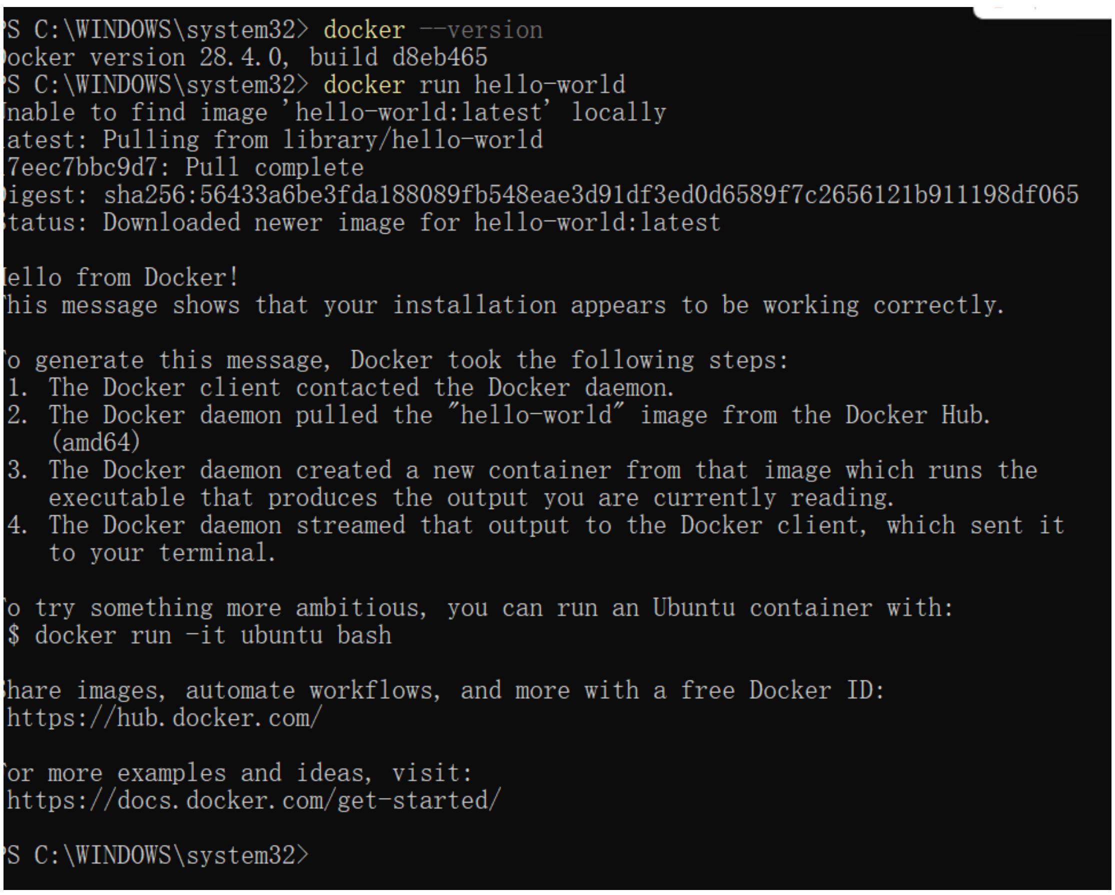
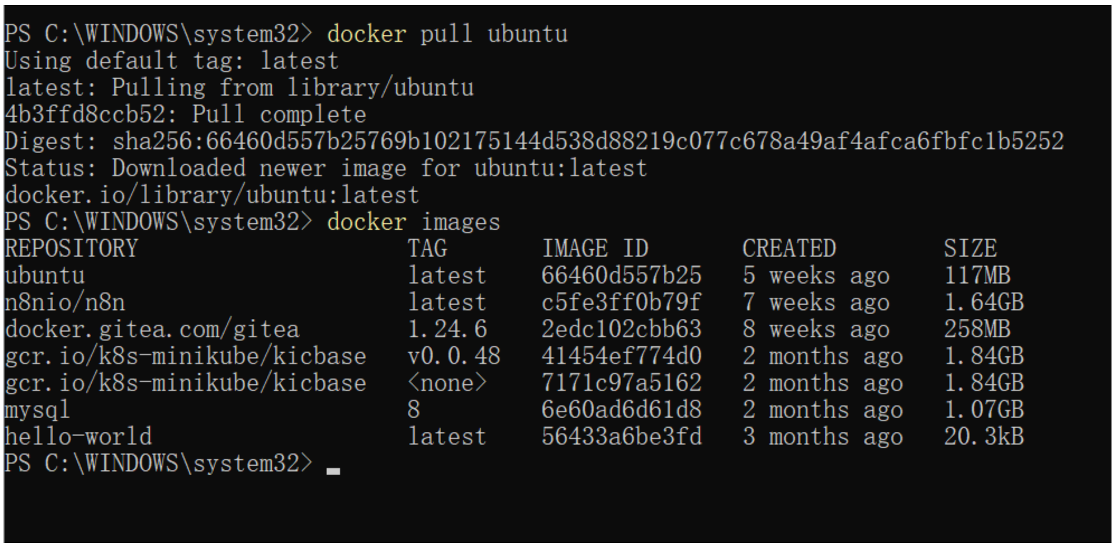
Lab 5 Docker Basics

1. Run Hello World Container



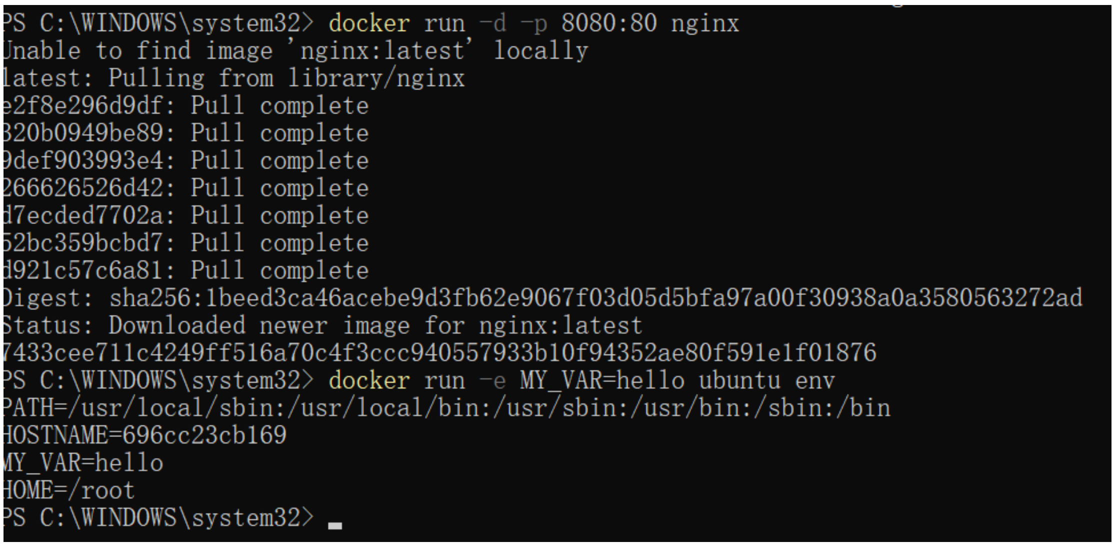
```
'S C:\WINDOWS\system32> docker --version
)ocker version 28.4.0, build d8eb465
'S C:\WINDOWS\system32> docker run hello-world
nable to find image 'hello-world:latest' locally
atest: Pulling from library/hello-world
7eec7bbc9d7: Pull complete
)igest: sha256:56433a6be3fda188089fb548eae3d91df3ed0d6589f7c2656121b911198df065
tatus: Downloaded newer image for hello-world:latest

Iello from Docker!
'his message shows that your installation appears to be working correctly.

'o 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.

'o try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

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

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

'S C:\WINDOWS\system32>
```

2 Pull and List Images

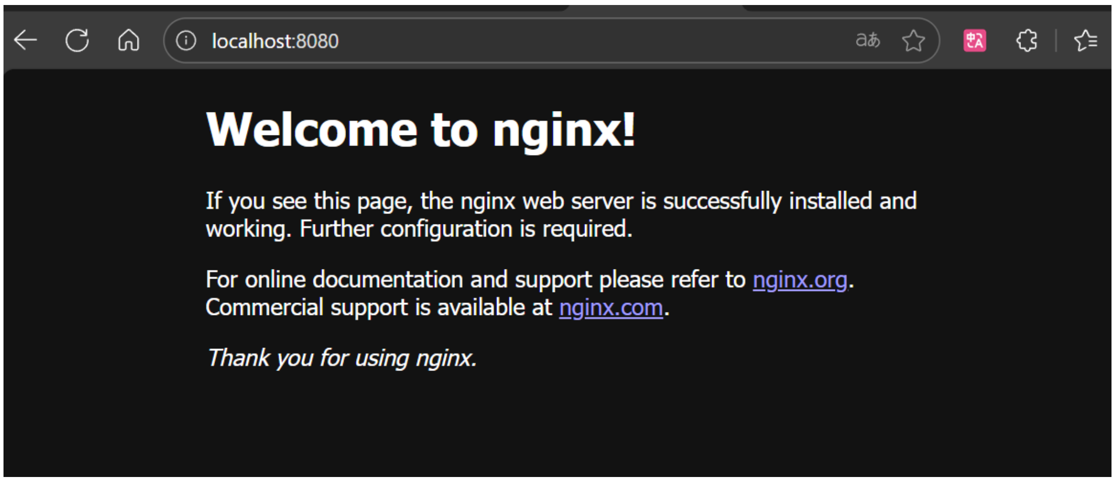


```
PS C:\WINDOWS\system32> docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
4b3ffd8ccb52: Pull complete
Digest: sha256:66460d557b25769b102175144d538d88219c077c678a49af4afca6fbfc1b5252
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
PS C:\WINDOWS\system32> docker images
REPOSITORY                     TAG       IMAGE ID       CREATED        SIZE
ubuntu                         latest    66460d557b25   5 weeks ago    117MB
n8nio/n8n                      latest    c5fe3ff0b79f   7 weeks ago    1.64GB
docker.gitea.com/gitea         1.24.6    2edc102cbb63   8 weeks ago    258MB
gcr.io/k8s-minikube/kicbase    v0.0.48   41454ef774d0   2 months ago   1.84GB
gcr.io/k8s-minikube/kicbase    <none>    7171c97a5162   2 months ago   1.84GB
mysql                          8         6e60ad6d61d8   2 months ago   1.07GB
hello-world                    latest    56433a6be3fd   3 months ago   20.3kB
PS C:\WINDOWS\system32>
```

3. Run a Container and Publish Ports



```
PS C:\WINDOWS\system32> docker run -d -p 8080:80 nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
e2f8e296d9df: Pull complete
320b0949be89: Pull complete
9def903993e4: Pull complete
266626526d42: Pull complete
d7ecded7702a: Pull complete
52bc359bcbd7: Pull complete
d921c57c6a81: Pull complete
Digest: sha256:1beed3ca46acebe9d3fb62e9067f03d05d5bfa97a00f30938a0a3580563272ad
Status: Downloaded newer image for nginx:latest
7433cee711c4249ff516a70c4f3ccc940557933b10f94352ae80f591e1f01876
PS C:\WINDOWS\system32> docker run -e MY_VAR=hello ubuntu env
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOSTNAME=696cc23cb169
MY_VAR=hello
HOME=/root
PS C:\WINDOWS\system32>
```



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Notes and Observations

- Containers are lightweight and isolated environments.
- Images are immutable templates for containers.
- Docker Compose simplifies managing multiple containers.
- Data persistence is essential for stateful applications.
- Sharing local files allows containerized applications to access host resources.

Conclusion

- Successfully installed Docker Desktop.
- Ran containers, published ports, and persisted data.
- Used Docker Compose to run multi-container applications.
- Documented operations with screenshots and notes.
- Gained practical experience with containerization and Docker concepts.