Docker container Lab

Prof. Chia-Yu Lin

National Central University

2024 Fall

Why use docker?

1. Faster Delivery and Deployment Developers can use standard images to build containers. Docker enables quick container creation, iteration, and deployment visibility.
2. More Efficient Virtualization Docker operates at the kernel level, requiring no additional virtualization support. It delivers higher efficiency and better performance.
3. Easier Migration and Scaling Docker containers run on any platform, including physical machines, virtual machines, clouds, and servers, ensuring seamless migration.
4. Simpler Management Incremental updates replace large-scale changes. Docker automates and simplifies management, ensuring efficiency and control.

Container vs Virtual Machines

Feature	Containers	Virtual Machines (Traditional Virtualization)
Startup Time	Seconds	Takes minutes at best
Capacity	MB	GB
Performance	Fast	Slow
Number of Supported Instances	Very many	Only about 10 instances is remarkable
Cloning Identical Environments	Fast	Very slow

Docker concept

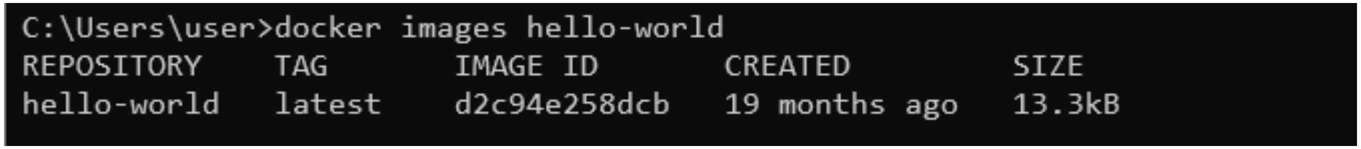
- Image
  - A read-only template containing the OS and applications. Images build containers and are easy to update or share.
- Container
  - A runnable instance of an image. Containers are isolated, secure platforms that can start, stop, and delete easily.
- Registry
  - A centralized repository storing images. Public and private registries like Docker Hub allow easy image sharing and retrieval.

Docker install

- Windows
  - https://www.docker.com/docker-windows
- MAC
  - https://docs.docker.com/desktop/setup/install/mac-install
- Linux
  - https://docs.docker.com/engine/install/ubuntu/

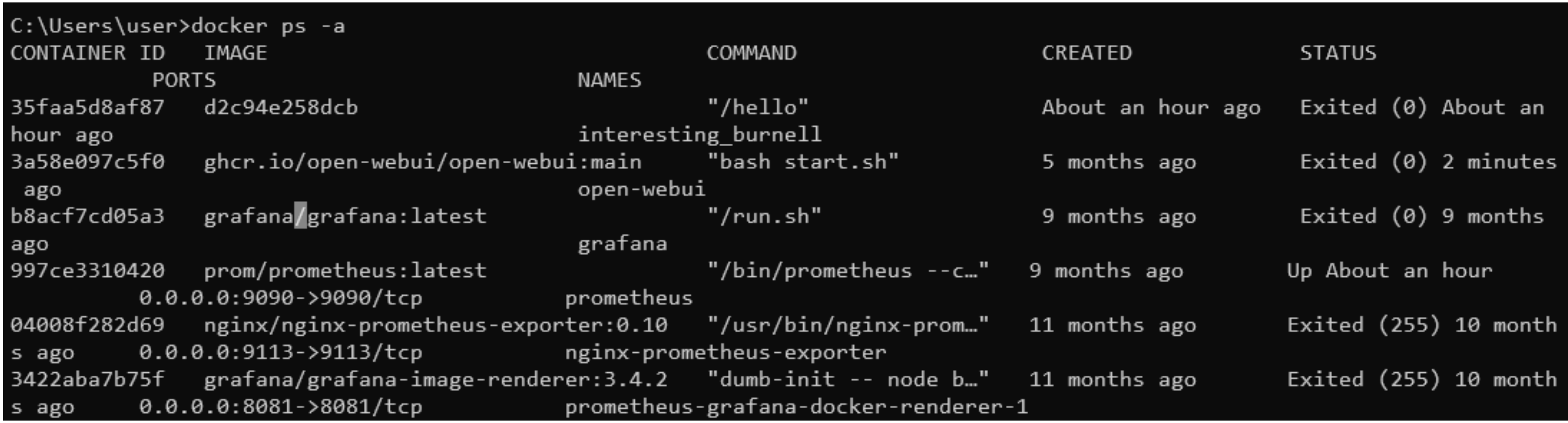
Docker commands - View

- docker images [image name]
  - It will inspect a specific image, while **docker images** will display all the images.



```
C:\Users\user>docker images hello-world
REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
hello-world   latest    d2c94e258dcb   19 months ago   13.3kB
```

- docker ps –a
  - To view all the current containers, including stopped containers



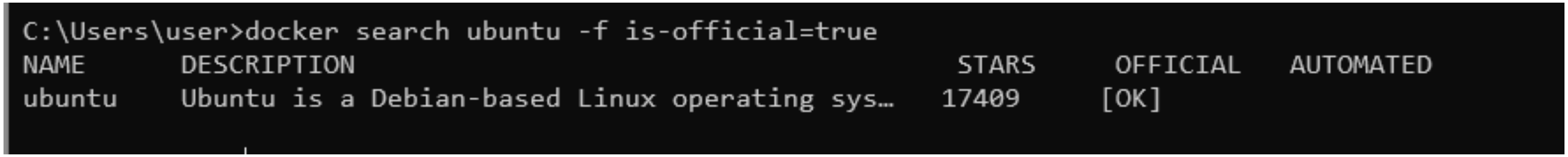
```
C:\Users\user>docker ps -a
CONTAINER ID   IMAGE                                  COMMAND                   CREATED             STATUS
           PORTS                                NAMES
35faa5d8af87   d2c94e258dcb                           "/hello"                  About an hour ago   Exited (0) About an
hour ago                                    interesting_burnell
3a58e097c5f0   ghcr.io/open-webui/open-webui:main     "bash start.sh"           5 months ago        Exited (0) 2 minutes
 ago                                        open-webui
b8acf7cd05a3   grafana/grafana:latest                 "/run.sh"                 9 months ago        Exited (0) 9 months
ago                                         grafana
997ce3310420   prom/prometheus:latest                 "/bin/prometheus --c…"   9 months ago        Up About an hour
          0.0.0.0:9090->9090/tcp           prometheus
04008f282d69   nginx/nginx-prometheus-exporter:0.10   "/usr/bin/nginx-prom…"   11 months ago       Exited (255) 10 month
s ago     0.0.0.0:9113->9113/tcp           nginx-prometheus-exporter
3422aba7b75f   grafana/grafana-image-renderer:3.4.2   "dumb-init -- node b…"   11 months ago       Exited (255) 10 month
s ago     0.0.0.0:8081->8081/tcp           prometheus-grafana-docker-renderer-1
```

Docker Commands - Remove

- docker rmi [image_name_or_image_id]
  - Delete docker image
- docker stop [container_id]
  - Stop docker container
- docker restart [container_id]
  - Restart docker container
- docker rm [container_id]
  - Delete docker container

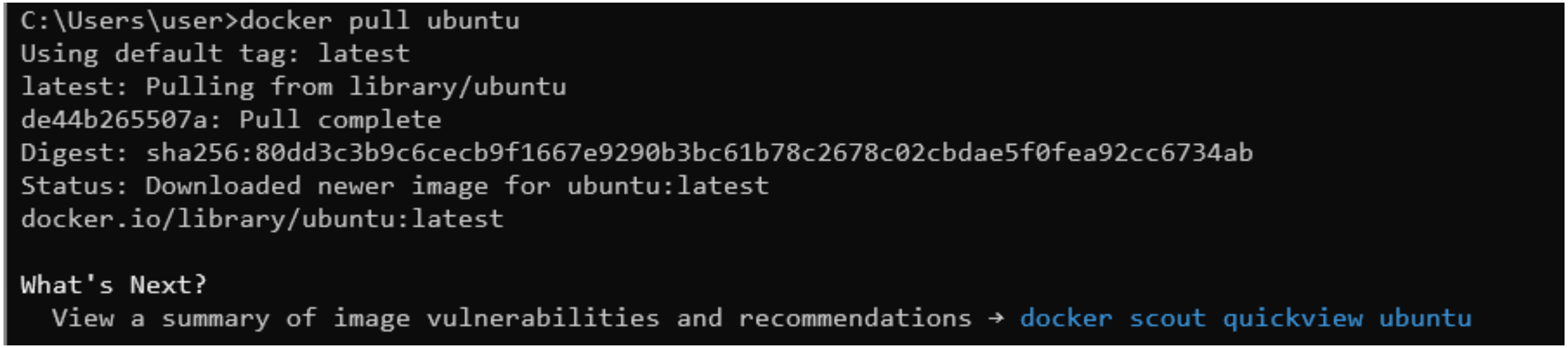
Example of Running Ubuntu in Docker (1/3)

- docker search
  - Search for the image to download
  - **-f is-official=true** means to search for official Docker images.



```
C:\Users\user>docker search ubuntu -f is-official=true
NAME      DESCRIPTION                                     STARS     OFFICIAL   AUTOMATED
ubuntu    Ubuntu is a Debian-based Linux operating sys…   17409     [OK]
```

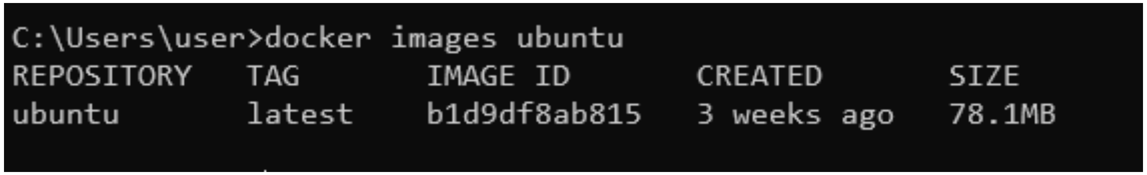
- docker pull
  - Pull this image down.



```
C:\Users\user>docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
de44b265507a: Pull complete
Digest: sha256:80dd3c3b9c6cecb9f1667e9290b3bc61b78c2678c02cbdae5f0fea92cc6734ab
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview ubuntu
```

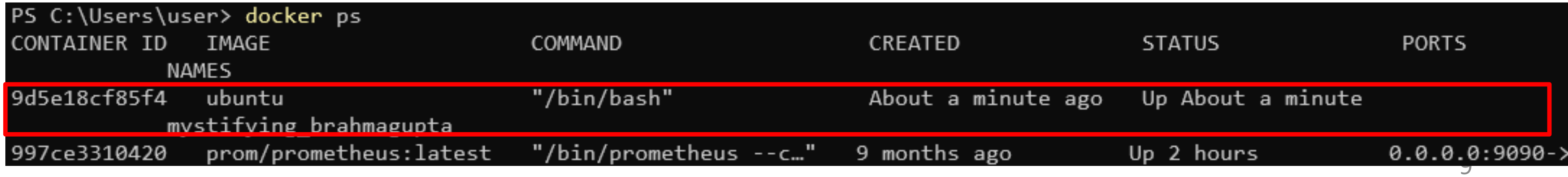
Example of Running Ubuntu in Docker (2/3)

- docker images ubuntu
  - To check if the image has been pulled down



```
C:\Users\user>docker images ubuntu
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
ubuntu        latest    b1d9df8ab815   3 weeks ago    78.1MB
```

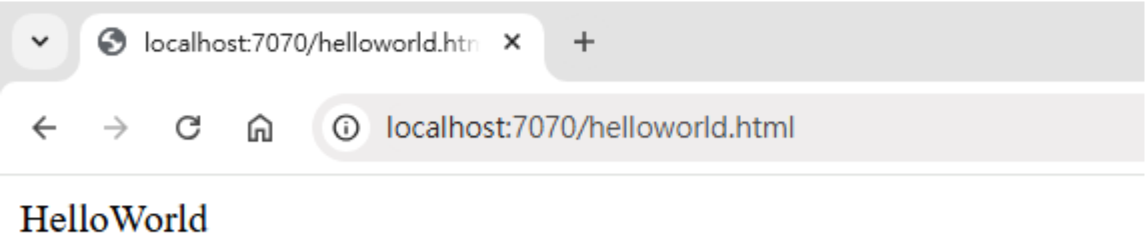
- docker run -it -p 8080:80 ubuntu /bin/bash
  - Once you have the Docker image, you can run the container and enter the Docker container's terminal.
- docker ps
  - Open another terminal window and check if the Docker container is running properly.



```
PS C:\Users\user> docker ps
CONTAINER ID   IMAGE                      COMMAND                  CREATED              STATUS              PORTS
          NAMES
9d5e18cf85f4   ubuntu                     "/bin/bash"              About a minute ago   Up About a minute
          mystifying_brahmagupta
997ce3310420   prom/prometheus:latest     "/bin/prometheus --c…"   9 months ago        Up 2 hours          0.0.0.0:9090->
```

Example of Running Ubuntu in Docker (3/3)

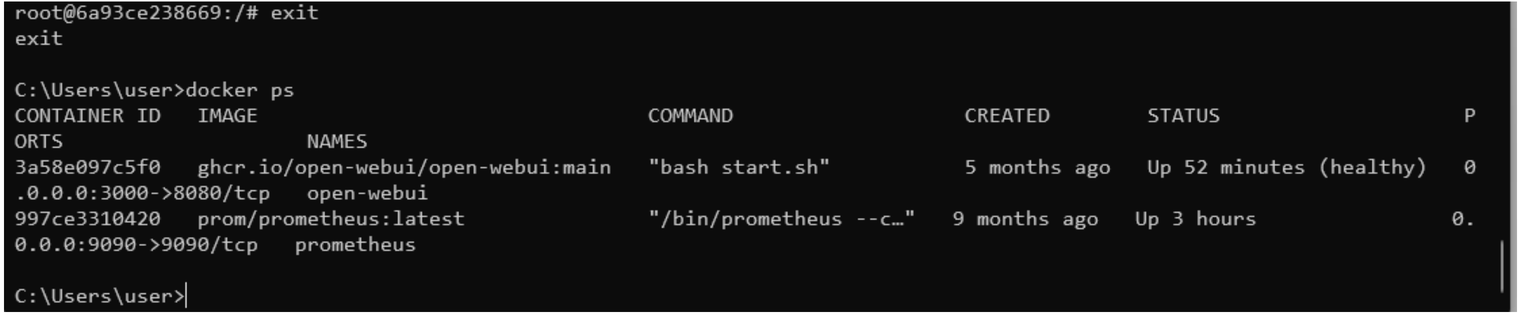
- To create a webpage on Ubuntu running inside a Docker container, follow these steps:
  - Install and Start Apache HTTP Service
    - apt-get update
    - apt-get install -y apache2
    - service apache2 start
  - Create a simple helloworld.html file and place it in the /var/www/html directory
    - echo "HelloWorld" > /var/www/html/helloworld.html
  - Go to http://localhost:7070/helloworld.html



HelloWorld

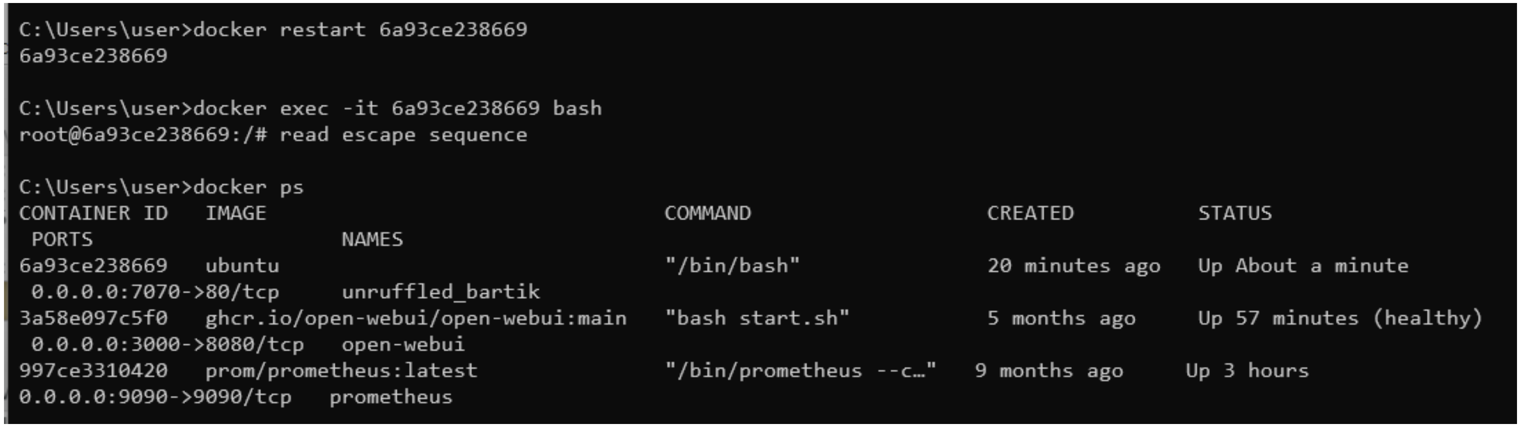
NOTE

- There is a pitfall when leaving the Docker container's terminal. If you enter the exit command, the container will be stopped.



```
root@6a93ce238669:/# exit
exit

C:\Users\user>docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED         STATUS                    P
ORTS                    NAMES
3a58e097c5f0   ghcr.io/open-webui/open-webui:main   "bash start.sh"          5 months ago    Up 52 minutes (healthy)   0
.0.0.0:3000->8080/tcp   open-webui
997ce3310420   prom/prometheus:latest               "/bin/prometheus --c…"   9 months ago   Up 3 hours                0.
0.0.0:9090->9090/tcp   prometheus

C:\Users\user>
```

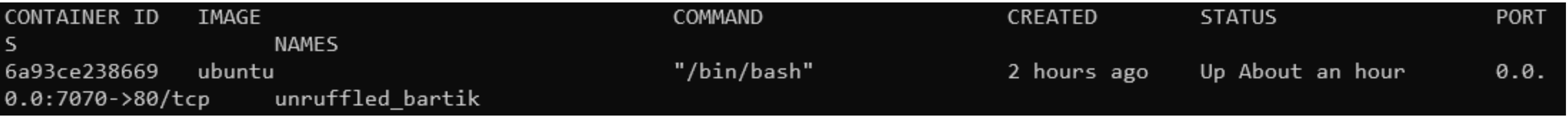
- If you don't want to stop the container and just want to exit the Docker container's terminal, you need to press Ctrl + P followed by Ctrl + Q. This will allow you to exit without stopping the container.



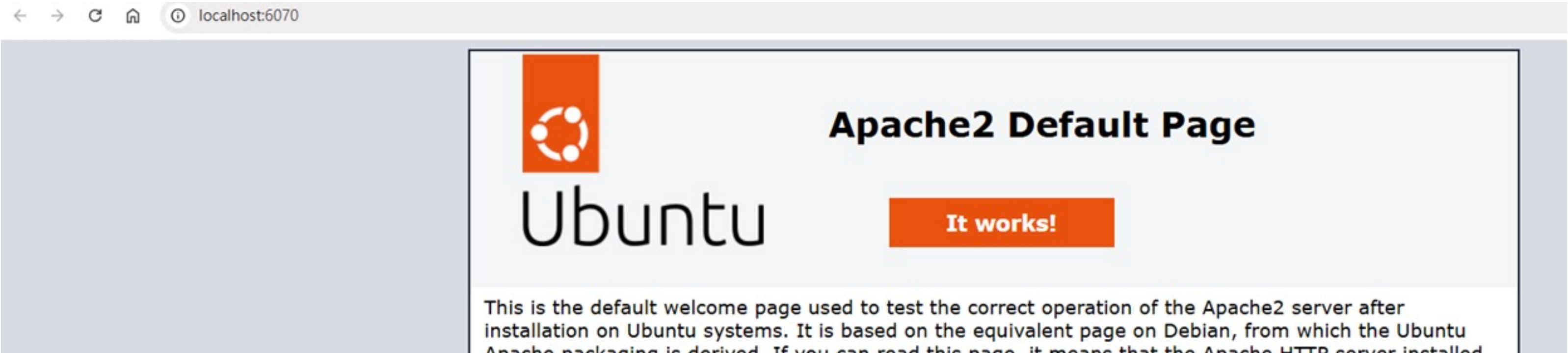
```
C:\Users\user>docker restart 6a93ce238669
6a93ce238669

C:\Users\user>docker exec -it 6a93ce238669 bash
root@6a93ce238669:/# read escape sequence

C:\Users\user>docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED          STATUS
 PORTS                  NAMES
6a93ce238669   ubuntu                               "/bin/bash"              20 minutes ago   Up About a minute
 0.0.0.0:7070->80/tcp      unruffled_bartik
3a58e097c5f0   ghcr.io/open-webui/open-webui:main   "bash start.sh"          5 months ago     Up 57 minutes (healthy)
 0.0.0.0:3000->8080/tcp   open-webui
997ce3310420   prom/prometheus:latest               "/bin/prometheus --c…"   9 months ago    Up 3 hours
0.0.0.0:9090->9090/tcp   prometheus
```

HW: Task 1

- Run an Ubuntu Docker container in the background and map the container's port 80 to a port of your choice on your host machine.
  After running the container, use a web browser to access the default Apache page (if available) on the mapped port.
- You should demonstrate that the container is running, and after installing Apache, display its main page.



```
CONTAINER ID   IMAGE                                   COMMAND             CREATED         STATUS               PORT
S                  NAMES
6a93ce238669   ubuntu                                  "/bin/bash"         2 hours ago     Up About an hour     0.0.
0.0:7070->80/tcp   unruffled_bartik
```



localhost:6070

Apache2 Default Page

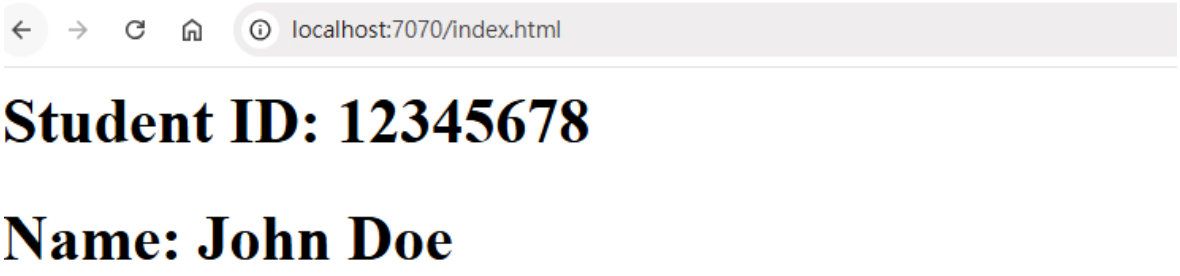
Ubuntu

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed

HW: Task 2

- Start an Ubuntu container and install Apache web server.Create a simple HTML webpage in the container that displays your student ID and name.
- Save the file as index.html in the /var/www/html/ directory.
- Afterward, use the container's IP address or localhost to access the webpage in your browser.
- You should display your name and student ID on the index.html page.



localhost:7070/index.html

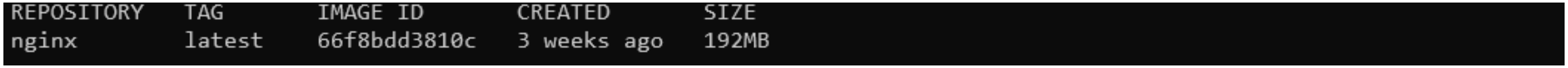
Student ID: 12345678

Name: John Doe

HW: Task 3 (1/2)

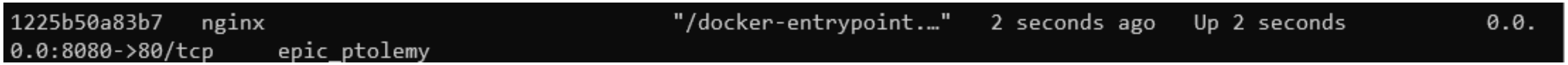
- Create a container from the **nginx** image and run it. Once it is running, stop the container and then restart it.Verify the container’s status before and after restarting.

- Show that you already have the Nginx image



```
REPOSITORY     TAG        IMAGE ID        CREATED        SIZE
nginx          latest     66f8bdd3810c    3 weeks ago    192MB
```

- Show that your container is running Nginx.

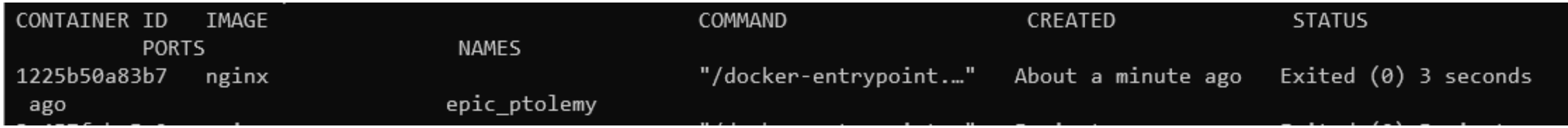


```
1225b50a83b7   nginx                                  "/docker-entrypoint.…"   2 seconds ago   Up 2 seconds             0.0.
0.0:8080->80/tcp      epic_ptolemy
```

HW: Task 3 (2/2)

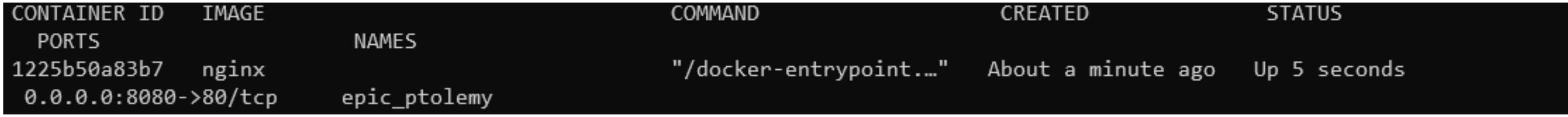
- Create a container from the **nginx** image and run it. Once it is running, stop the container and then restart it.Verify the container’s status before and after restarting.

- Show that your nginx container stops.



```
CONTAINER ID   IMAGE                                      COMMAND                  CREATED              STATUS
          PORTS                    NAMES
1225b50a83b7   nginx                                      "/docker-entrypoint.…"   About a minute ago   Exited (0) 3 seconds
 ago                              epic_ptolemy
```

- Show that your nginx container restarts.

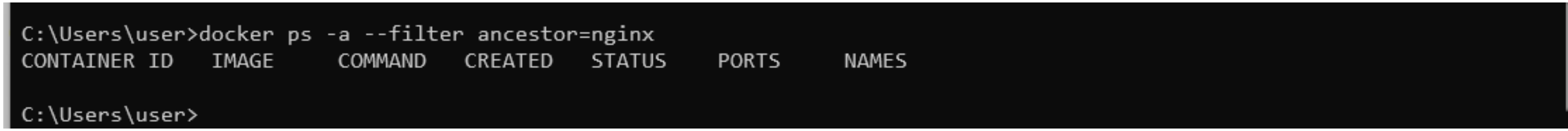


```
CONTAINER ID   IMAGE                                      COMMAND                  CREATED              STATUS
 PORTS                    NAMES
1225b50a83b7   nginx                                      "/docker-entrypoint.…"   About a minute ago   Up 5 seconds
 0.0.0.0:8080->80/tcp     epic_ptolemy
```

HW: Task 4

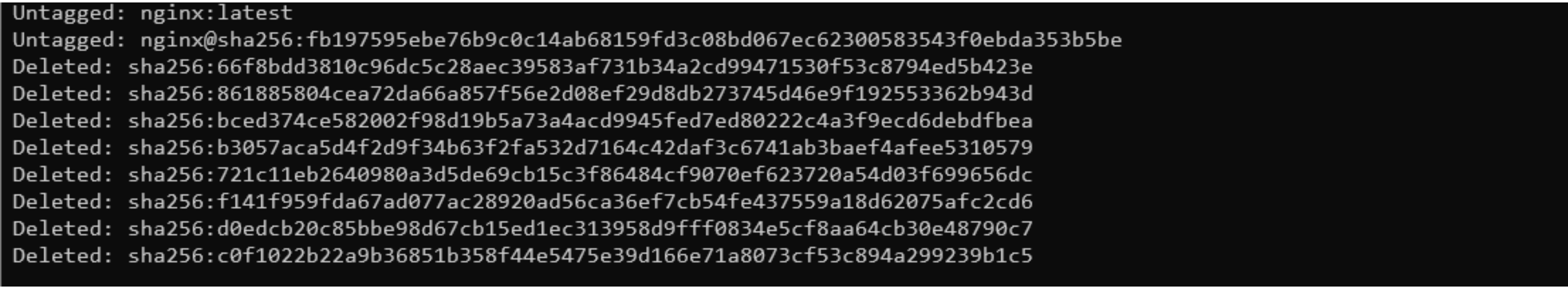
- After running the nginx container from the previous task, remove the container and the image.Verify that the container and the image are no longer present on the system.

- Show that you have deleted your container



```
C:\Users\user>docker ps -a --filter ancestor=nginx
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES

C:\Users\user>
```

- Show that you remove the nginx image



```
Untagged: nginx:latest
Untagged: nginx@sha256:fb197595ebe76b9c0c14ab68159fd3c08bd067ec62300583543f0ebda353b5be
Deleted: sha256:66f8bdd3810c96dc5c28aec39583af731b34a2cd99471530f53c8794ed5b423e
Deleted: sha256:861885804cea72da66a857f56e2d08ef29d8db273745d46e9f192553362b943d
Deleted: sha256:bced374ce582002f98d19b5a73a4acd9945fed7ed80222c4a3f9ecd6debdfbea
Deleted: sha256:b3057aca5d4f2d9f34b63f2fa532d7164c42daf3c6741ab3baef4afee5310579
Deleted: sha256:721c11eb2640980a3d5de69cb15c3f86484cf9070ef623720a54d03f699656dc
Deleted: sha256:f141f959fda67ad077ac28920ad56ca36ef7cb54fe437559a18d62075afc2cd6
Deleted: sha256:d0edcb20c85bbe98d67cb15ed1ec313958d9fff0834e5cf8aa64cb30e48790c7
Deleted: sha256:c0f1022b22a9b36851b358f44e5475e39d166e71a8073cf53c894a299239b1c5
```