

ITI Docker Lab2

Mazen AbdelTawab Saad

Problem 1:

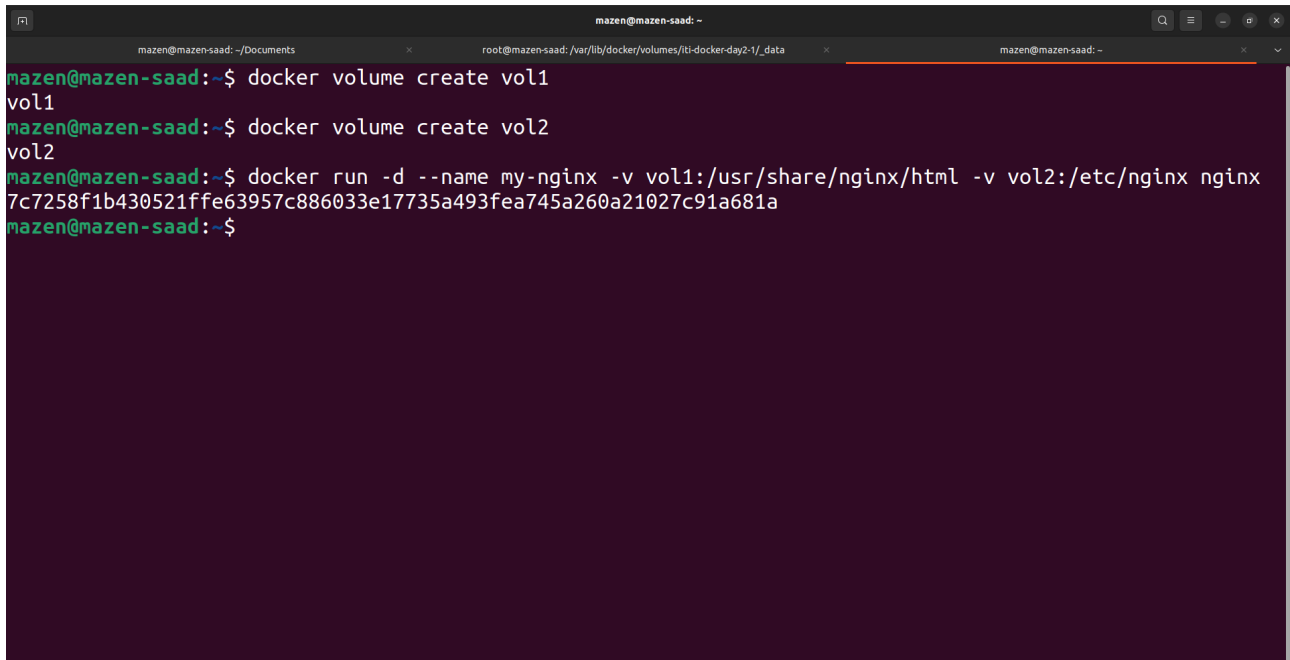
1- Run a container nginx with name my-nginx and attach 2 volumes to the container using volume mount

|

docker volume create vol1

docker volume create vol2

docker run -d --name my-nginx -v vol1:/usr/share/nginx/html -v vol2:/etc/nginx nginx

A terminal window with a dark purple background and green text. The window title is 'mazen@mazen-saad: ~'. There are three tabs open: 'mazen@mazen-saad: ~/Documents', 'root@mazen-saad: /var/lib/docker/volumes/iti-docker-day2-1/_data', and 'mazen@mazen-saad: ~'. The terminal shows the following commands and output:

```
mazen@mazen-saad:~$ docker volume create vol1
vol1
mazen@mazen-saad:~$ docker volume create vol2
vol2
mazen@mazen-saad:~$ docker run -d --name my-nginx -v vol1:/usr/share/nginx/html -v vol2:/etc/nginx nginx
7c7258f1b430521ffe63957c886033e17735a493fea745a260a21027c91a681a
mazen@mazen-saad:~$
```

- 2- Volume1 for containing static html file
- 3- Volume2 for containing nginx configuration

4- Edit the html content

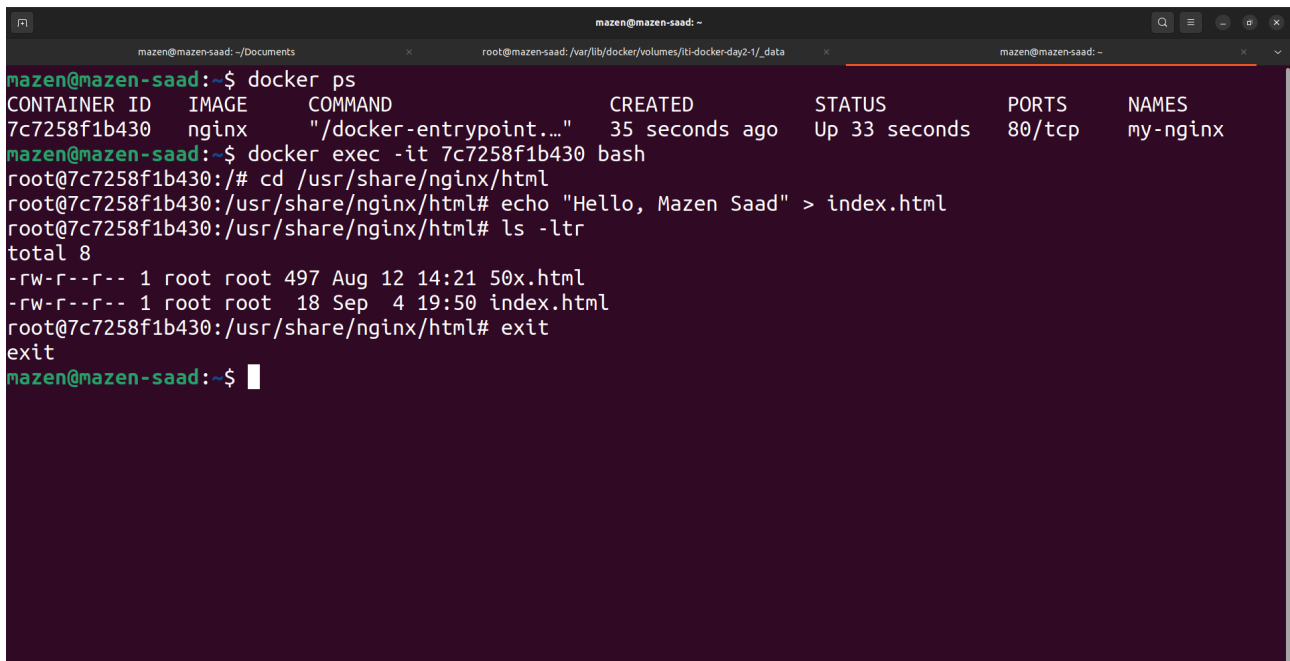
|

docker ps

docker exec -it my-nginx bash

cd /usr/share/nginx/html

echo "Hello, Mazen Saad" > index.html



```
mazen@mazen-saad: ~  
mazen@mazen-saad: ~/Documents  
root@mazen-saad: /var/lib/docker/volumes/lti-docker-day2-1/_data  
mazen@mazen-saad: ~  
mazen@mazen-saad:~$ docker ps  
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS        NAMES  
7c7258f1b430   nginx    "/docker-entrypoint...." 35 seconds ago Up 33 seconds 80/tcp       my-nginx  
mazen@mazen-saad:~$ docker exec -it 7c7258f1b430 bash  
root@7c7258f1b430:/# cd /usr/share/nginx/html  
root@7c7258f1b430:/usr/share/nginx/html# echo "Hello, Mazen Saad" > index.html  
root@7c7258f1b430:/usr/share/nginx/html# ls -ltr  
total 8  
-rw-r--r-- 1 root root 497 Aug 12 14:21 50x.html  
-rw-r--r-- 1 root root 18 Sep  4 19:50 index.html  
root@7c7258f1b430:/usr/share/nginx/html# exit  
exit  
mazen@mazen-saad:~$
```

5- Remove the container

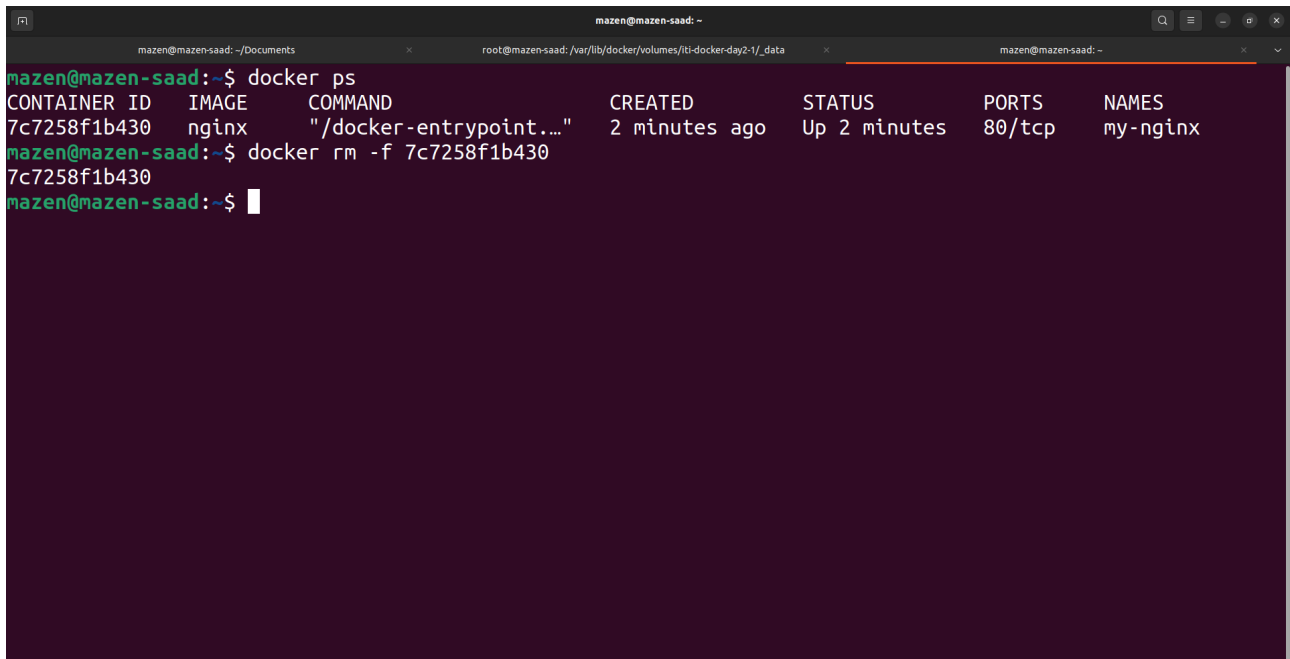
|

`docker ps`

`docker rm -f my-nginx`

or

`docker rm -f md88j8d3`

A terminal window with a dark background and light green text. The window title is 'mazen@mazen-saad: ~'. The terminal shows the command 'docker ps' being executed, followed by a table of running containers. The table has columns: CONTAINER ID, IMAGE, COMMAND, CREATED, STATUS, PORTS, and NAMES. One container is listed: 7c7258f1b430, nginx, "/docker-entrypoint....", 2 minutes ago, Up 2 minutes, 80/tcp, my-nginx. Then, the command 'docker rm -f 7c7258f1b430' is entered. The prompt returns to 'mazen@mazen-saad: ~\$' with a cursor.

```
mazen@mazen-saad: ~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS        NAMES
7c7258f1b430   nginx    "/docker-entrypoint...." 2 minutes ago  Up 2 minutes  80/tcp      my-nginx
mazen@mazen-saad: ~$ docker rm -f 7c7258f1b430
7c7258f1b430
mazen@mazen-saad: ~$
```

6- Run a new 2 containers with the following:

6-1 Attach the two volumes that were attached to the previous container using volume mount

6-2 Map port 80 to port 8080 on you host machine

|
docker run -d --name nginx1 -v vol1:/usr/share/nginx/html -v vol2:/etc/nginx -p 8080:80 nginx

docker run -d --name nginx2 -v vol1:/usr/share/nginx/html -v vol2:/etc/nginx -p 8081:80 nginx

```
mazen@mazen-saad: ~  
mazen@mazen-saad: ~/Documents  
root@mazen-saad: /var/lib/docker/volumes/iti-docker-day2-1/_data  
mazen@mazen-saad: ~  
mazen@mazen-saad: $ docker run -d --name nginx1 -v vol1:/usr/share/nginx/html -v vol2:/etc/nginx -p 8080:80 nginx  
417575b00350f5032e7e83fc55996bfec3dcf6e46009758a3a749e42f43a66ca  
mazen@mazen-saad: $ docker run -d --name nginx2 -v vol1:/usr/share/nginx/html -v vol2:/etc/nginx -p 8081:80 nginx  
f5bd45b612ab784084a88cf9e96442ffeff480748cd4824ab35bec82e539311e  
mazen@mazen-saad: $ docker ps  
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES  
f5bd45b612ab   nginx     "/docker-entrypoint..." 10 seconds ago Up 8 seconds   0.0.0.0:8081->80/tcp, [::]:8081->80/tcp  nginx2  
417575b00350   nginx     "/docker-entrypoint..." 33 seconds ago Up 31 seconds   0.0.0.0:8080->80/tcp, [::]:8080->80/tcp  nginx1  
mazen@mazen-saad: $
```

6-3 Access the html files from your browser

|

Open a web browser and navigate to

<http://localhost:8080>

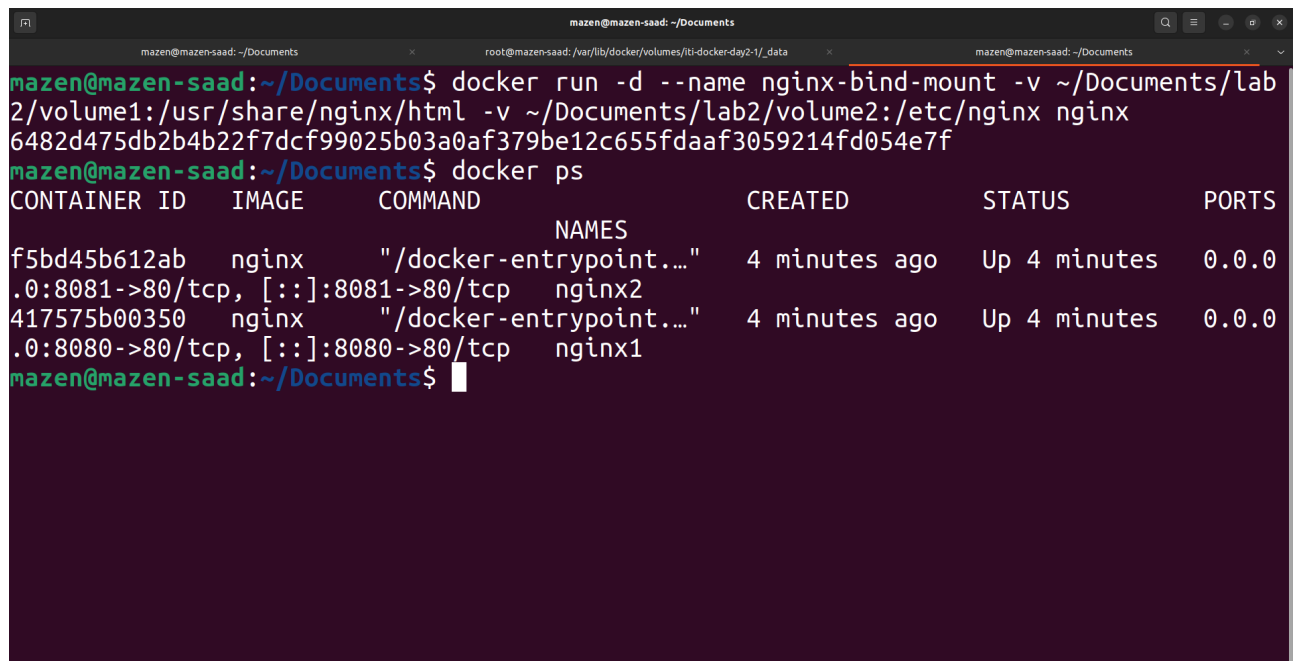
<http://localhost:8081>



Problem 2:

1- Run a container Nginx with name nginx-bind-mount and attach 2 volumes using bind mount under any paths

```
|
docker run -d --name nginx-bind-mount -v
~/Documents/lab2/volume1:/usr/share/nginx/html -v
~/Documents/lab2/volume2:/etc/nginx nginx
```



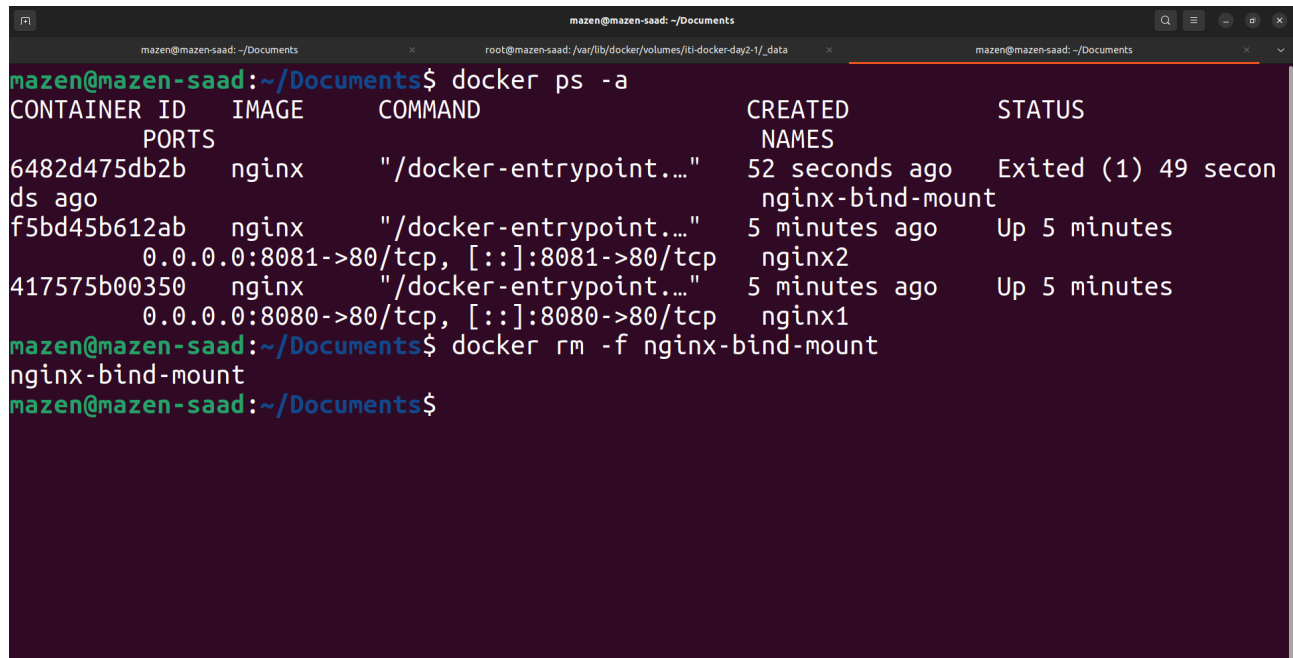
```
mazen@mazen-saad: ~/Documents
mazen@mazen-saad: ~/Documents$ docker run -d --name nginx-bind-mount -v ~/Documents/lab
2/volume1:/usr/share/nginx/html -v ~/Documents/lab2/volume2:/etc/nginx nginx
6482d475db2b4b22f7dcf99025b03a0af379be12c655fdaaf3059214fd054e7f
mazen@mazen-saad: ~/Documents$ docker ps
CONTAINER ID   IMAGE     COMMAND                  NAMES
f5bd45b612ab   nginx    "/docker-entrypoint...."  nginx2
.0:8081->80/tcp, [::]:8081->80/tcp
417575b00350   nginx    "/docker-entrypoint...."  nginx1
.0:8080->80/tcp, [::]:8080->80/tcp
mazen@mazen-saad: ~/Documents$
```

2- Remove the container

|

```
docker ps -a
```

```
docker rm -f nginx-bind-mount
```



A terminal window with a dark background and light green text. The window title is 'mazen@mazen-saad: ~/Documents'. The terminal shows the following commands and output:

```
mazen@mazen-saad:~/Documents$ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
6482d475db2b	nginx	"/docker-entryptoint..."	52 seconds ago	Exited (1) 49 seconds ago
f5bd45b612ab	nginx	"/docker-entryptoint..."	5 minutes ago	Up 5 minutes
417575b00350	nginx	"/docker-entryptoint..."	5 minutes ago	Up 5 minutes

The output also includes port mappings for the running containers:

- 0.0.0.0:8081->80/tcp, [::]:8081->80/tcp (nginx2)
- 0.0.0.0:8080->80/tcp, [::]:8080->80/tcp (nginx1)

```
mazen@mazen-saad:~/Documents$ docker rm -f nginx-bind-mount
nginx-bind-mount
mazen@mazen-saad:~/Documents$
```


3- Run a new container with the following:

3-1 Attach the two volumes that were attached to the previous container

3-2 Check the old data in the new containers

|

docker run -it --name nginx-bind-mount-2 -v

~/Documents/lab2/volume1:/usr/share/nginx/html -v

~/Documents/lab2/volume2:/etc/nginx nginx bash

ls -ltr

exit

```
mazen@mazen-saad: ~/Documents
mazen@mazen-saad: ~/Documents
root@mazen-saad: /var/lib/docker/volumes/iti-docker-day2-1/_data
mazen@mazen-saad: ~/Documents

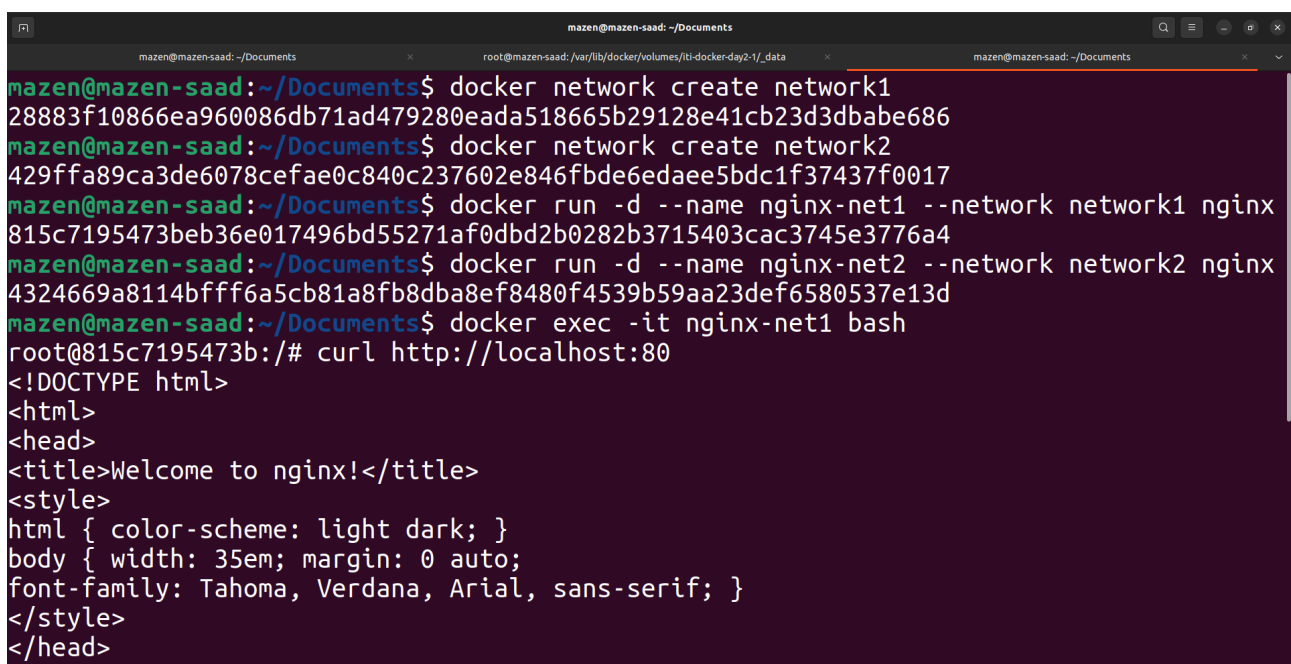
mazen@mazen-saad:~/Documents$ docker run -it --name nginx-bind-mount-2 -v ~/Documents/
lab2/volume1:/usr/share/nginx/html -v ~/Documents/lab2/volume2:/etc/nginx nginx bash
root@6b42bca19646:/# ls -ltr
total 56
drwxr-xr-x  2 root root 4096 Mar 29 17:20 home
drwxr-xr-x  2 root root 4096 Mar 29 17:20 boot
drwxr-xr-x  1 root root 4096 Aug 12 00:00 var
drwxr-xr-x  1 root root 4096 Aug 12 00:00 usr
drwxrwxrwt  2 root root 4096 Aug 12 00:00 tmp
drwxr-xr-x  2 root root 4096 Aug 12 00:00 srv
lrwxrwxrwx  1 root root   8 Aug 12 00:00 sbin -> usr/sbin
drwxr-xr-x  3 root root 4096 Aug 12 00:00 run
drwx----- 2 root root 4096 Aug 12 00:00 root
drwxr-xr-x  2 root root 4096 Aug 12 00:00 opt
drwxr-xr-x  2 root root 4096 Aug 12 00:00 mnt
drwxr-xr-x  2 root root 4096 Aug 12 00:00 media
lrwxrwxrwx  1 root root   9 Aug 12 00:00 lib64 -> usr/lib64
lrwxrwxrwx  1 root root   7 Aug 12 00:00 lib -> usr/lib
lrwxrwxrwx  1 root root   7 Aug 12 00:00 bin -> usr/bin
-rwxr-xr-x  1 root root 1620 Aug 15 17:58 docker-entrypoint.sh
```

Problem 3:

1- Create 2 nginx containers with 2 different network of type bridge, enter to one of them and use curl command to view the content of the other container.

```
|
docker network create network1
docker network create network2
docker run -d --name nginx-net1 --network network1 nginx
docker run -d --name nginx-net2 --network network2 nginx

docker exec -it nginx-net1 bash
curl http://localhost:80
```

A terminal window with a dark background and light-colored text. The window title is 'mazen@mazen-saad: ~/Documents'. The terminal shows the following commands and output:
mazen@mazen-saad:~/Documents\$ docker network create network1
28883f10866ea960086db71ad479280eada518665b29128e41cb23d3dbabe686
mazen@mazen-saad:~/Documents\$ docker network create network2
429ffa89ca3de6078cefae0c840c237602e846fbde6edae5bdc1f37437f0017
mazen@mazen-saad:~/Documents\$ docker run -d --name nginx-net1 --network network1 nginx
815c7195473beb36e017496bd55271af0dbd2b0282b3715403cac3745e3776a4
mazen@mazen-saad:~/Documents\$ docker run -d --name nginx-net2 --network network2 nginx
4324669a8114bfff6a5cb81a8fb8dba8ef8480f4539b59aa23def6580537e13d
mazen@mazen-saad:~/Documents\$ docker exec -it nginx-net1 bash
root@815c7195473b:/# curl http://localhost:80
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>

Problem 4:

1- Create docker compose with: Two services nginx and mysql Add needed ports and environments for both services nginx service is depending on mysql service

docker-compose.yaml

services:

mysql:

image: mysql:latest

environment:

MYSQL_ROOT_PASSWORD: 123123

MYSQL_DATABASE: iti_docker_lab2

MYSQL_USER: u_docker

MYSQL_PASSWORD: 123123

ports:

- "3306:3306"

nginx:

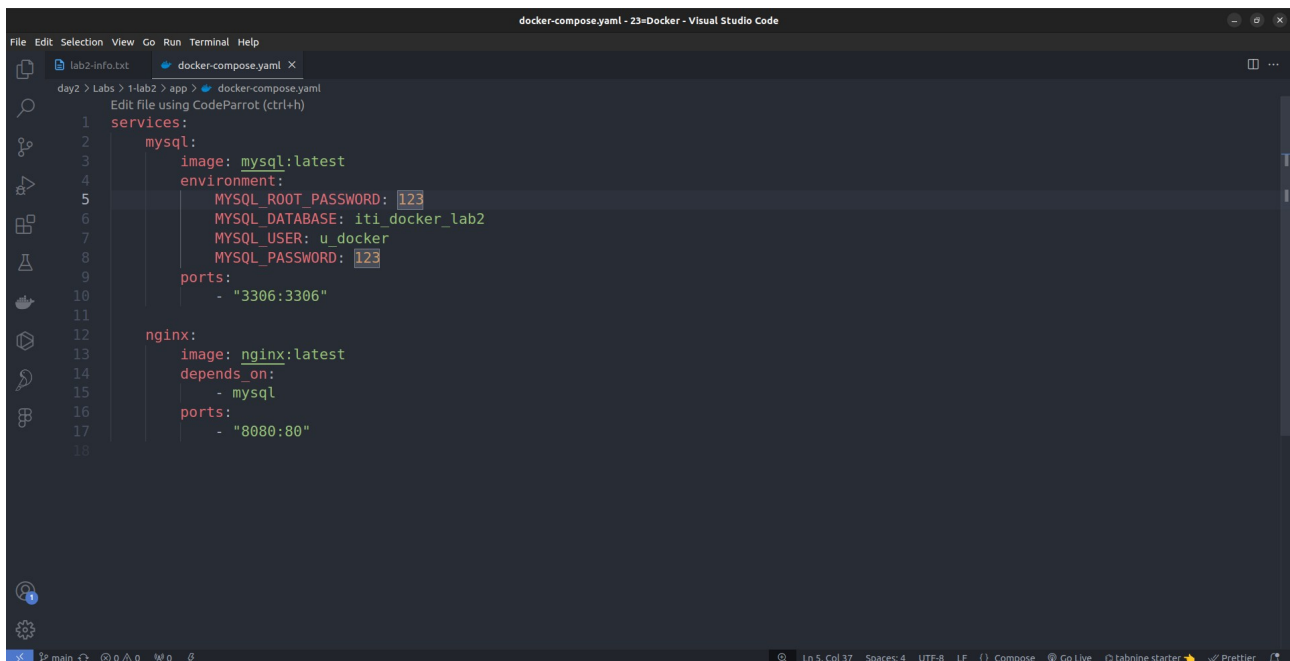
image: nginx:latest

depends_on:

- mysql

ports:

- "8080:80"

A screenshot of the Visual Studio Code editor interface. The title bar at the top reads "docker-compose.yaml - 23-Docker - Visual Studio Code". The editor window shows a file named "docker-compose.yaml" with the following content:

```
1 services:
2   mysql:
3     image: mysql:latest
4     environment:
5       MYSQL_ROOT_PASSWORD: 123
6       MYSQL_DATABASE: iti_docker_lab2
7       MYSQL_USER: u_docker
8       MYSQL_PASSWORD: 123
9     ports:
10      - "3306:3306"
11
12   nginx:
13     image: nginx:latest
14     depends_on:
15       - mysql
16     ports:
17       - "8080:80"
```

The interface includes a sidebar on the left with icons for Explorer, Search, Run and Debug, and Docker. The bottom status bar shows "Ln 5, Col 37", "Spaces: 4", "UTF-8", "LF", and several extension icons like Compose, Go Live, tabnine starter, and Prettier.