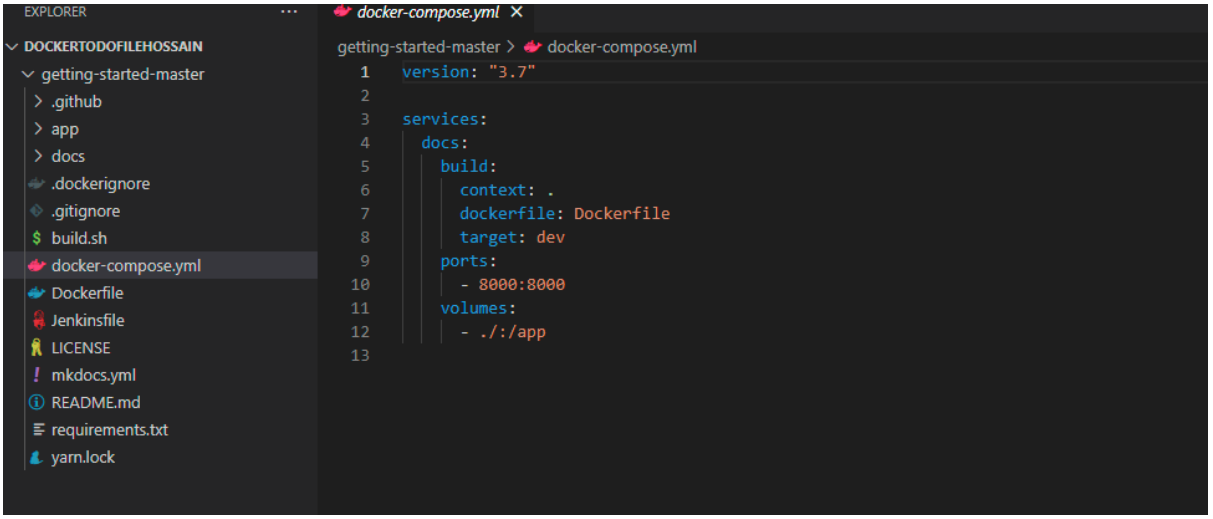
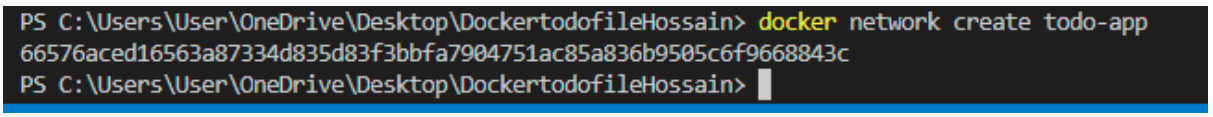
Docker Homework 01
Name : Hossain Md Al Amin
Group :212

Part 1

Start MySQL



Create the network.



Start a MySQL container and attach it to the network. We're also going to define a few environment variables that the database will use to initialize the database.

```
PS C:\Users\User\OneDrive\Desktop\DockertodofileHossain> docker run -d `
>>      --network todo-app --network-alias mysql `
>>      -v todo-mysql-data:/var/lib/mysql `
c19d5474d2cf: Pull complete
734cfb67bc8c: Pull complete
729529a0299e: Pull complete
63d3c603a591: Pull complete
305e0fbc2dcb: Pull complete
```

```
PS C:\Users\User\OneDrive\Desktop\DockertodofileHossain> docker run -d `
>>      --network todo-app --network-alias mysql `
>>      -v todo-mysql-data:/var/lib/mysql `
>>      -e MYSQL_ROOT_PASSWORD=secret `
>>      -e MYSQL_DATABASE=todos `
>>      mysql:5.7
19279a0f450268cbefd98f09a4174147cc98226b4e88a09eeffe83abee08a76f
PS C:\Users\User\OneDrive\Desktop\DockertodofileHossain>
```

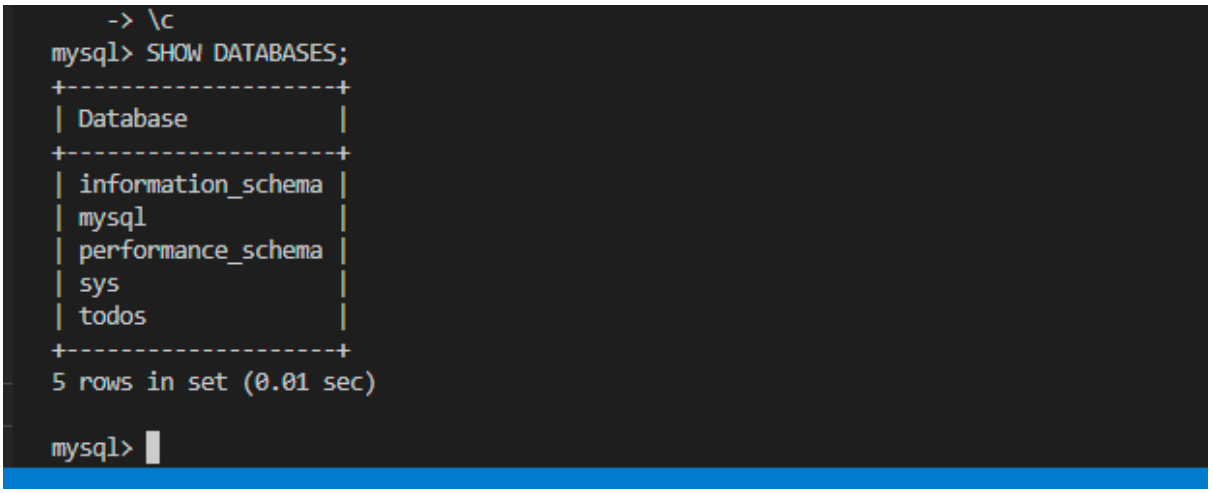
```
PS C:\Users\User\OneDrive\Desktop\DockertodofileHossain> docker ps
CONTAINER ID   IMAGE       COMMAND                  CREATED          STATUS          PORTS                 NAMES
da415e298ac4   mysql:5.7   "docker-entrypoint.s…"   12 minutes ago   Up 12 minutes   3306/tcp, 33060/tcp   heuristic_heyrovsky
PS C:\Users\User\OneDrive\Desktop\DockertodofileHossain> docker exec -it da415e298ac4 mysql -u root -p
Enter password:
```

Ln 1, Col 1 Spaces: 2 UTF-8 LF

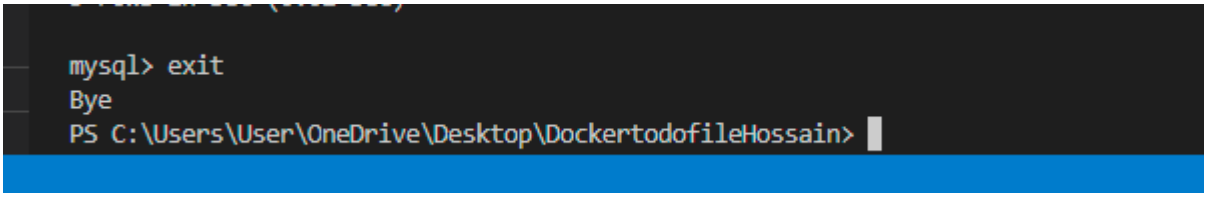
 # To confirm we have the database up and running, connect to the database and verify it connects.

```
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.7.40 MySQL Community Server (GPL)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

#Output looks like this

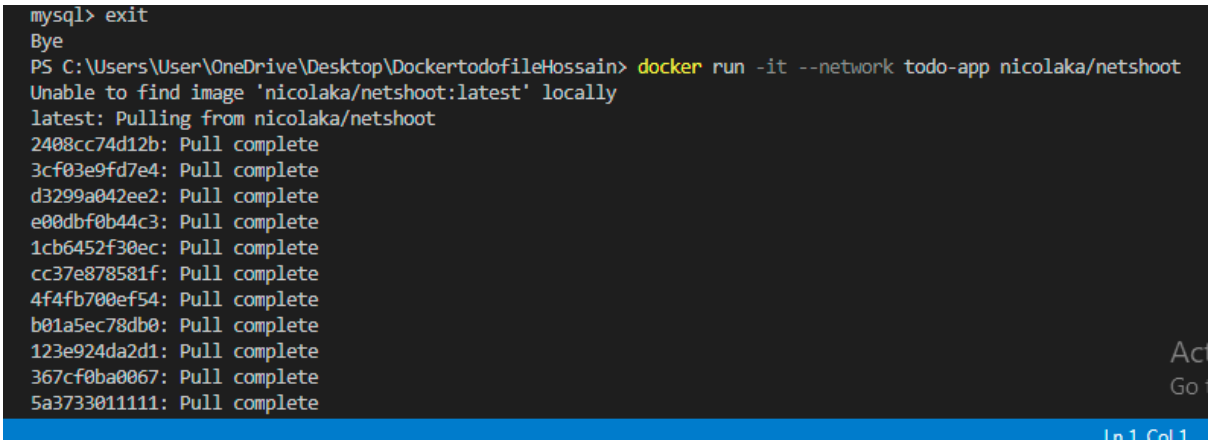


 # Exit the MySQL shell to return to the shell on our machine.

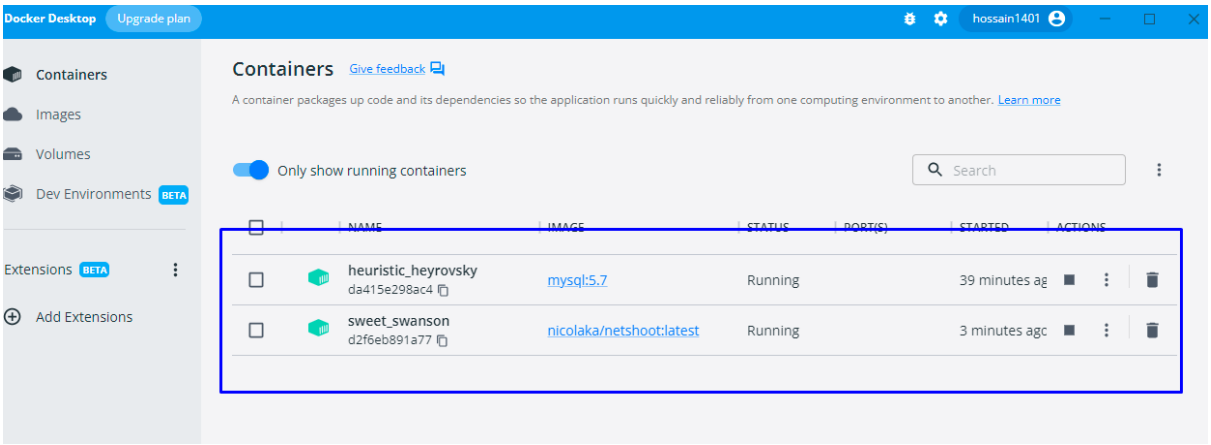


 # Connect to MySQL

Start a new container using the nicolaka/netshoot image. Make sure to connect it to the same network.



```
6adf3d98c1c3: Pull complete
77c79836780d: Pull complete
4aafb6d72b2b: Pull complete
Digest: sha256:aeafd567d7f7f1edb5127ec311599bb2b8a9c0fb31d7a53e9cff26af6d29fd4e
Status: Downloaded newer image for nicolaka/netshoot:latest
                    dP            dP                               dP
                    88            88                               88
88d888b. .d8888b. d8888P .d8888b. 88d888b. .d8888b. .d8888b. d8888P
88'  `88 88ooood8   88   Y8ooooo. 88'  `88 88'  `88 88'  `88   88
88    88 88.  ...   88         88 88    88 88.  .88 88.  .88   88
dP    dP `88888P'   dP   `88888P' dP    dP `88888P' `88888P'   dP

Welcome to Netshoot! (github.com/nicolaka/netshoot)

d2f6eb891a77 ~
```



 # Inside the container, we're going to use the dig command, which is a useful DNS tool. We're going to look up the IP address for the hostname mysql.

```
d2f6eb891a77 ~ dig mysql

; <<>> DiG 9.18.3 <<>> mysql
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 63416
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;mysql.                         IN      A

;; ANSWER SECTION:
mysql.                  600     IN      A       172.18.0.2

;; Query time: 180 msec
```

```
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 63416
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;mysql.                         IN      A

;; ANSWER SECTION:
mysql.                  600     IN      A       172.18.0.2

;; Query time: 180 msec
;; SERVER: 127.0.0.11#53(127.0.0.11) (UDP)
;; WHEN: Sat Oct 29 11:56:30 UTC 2022
;; MSG SIZE  rcvd: 44

d2f6eb891a77 ~
```

 # Run app with MySQL

```
mysql> ALTER USER root INDENTIFIED WITH mysql_native_password BY secret
    -> flush privileges
    ->
```

 # We’ll specify each of the environment variables above, as well as connect the container to our app network.

```
PS C:\Users\User\OneDrive\Desktop\DockertodofileHossain>  docker run -dp 4000:4000 `
>>    -w /app -v "$(pwd):/app" `
>>    --network todo-app `
>>    -e MYSQL_HOST=mysql `
>>    -e MYSQL_USER=root `
>>    -e MYSQL_PASSWORD=secret `
>>    -e MYSQL_DB=todos `
>>    node:12-alpine `
>>    sh -c "yarn install && yarn run dev"
690f4cd5c78358163edf055f9920d38bbd4f9f9cbd07d3c0894da05dd4aa8f5c
PS C:\Users\User\OneDrive\Desktop\DockertodofileHossain>
```

Built docker file:

```
PS C:\Users\User\OneDrive\Desktop\DockertodofileHossain> docker build -t getting-started .
[+] Building 1.7s (2/2) FINISHED
 => [internal] load build definition from Dockerfile                                    1.1s
 => => transferring dockerfile: 2B                                                      0.2s
 => [internal] load .dockerignore                                                       1.4s
 => => transferring context: 2B                                                         0.3s
failed to solve with frontend dockerfile.v0: failed to read dockerfile: open /var/lib/docker/tmp/buildkit-mount985958070/Dockerf
ile: no such file or directory
PS C:\Users\User\OneDrive\Desktop\DockertodofileHossain>
```

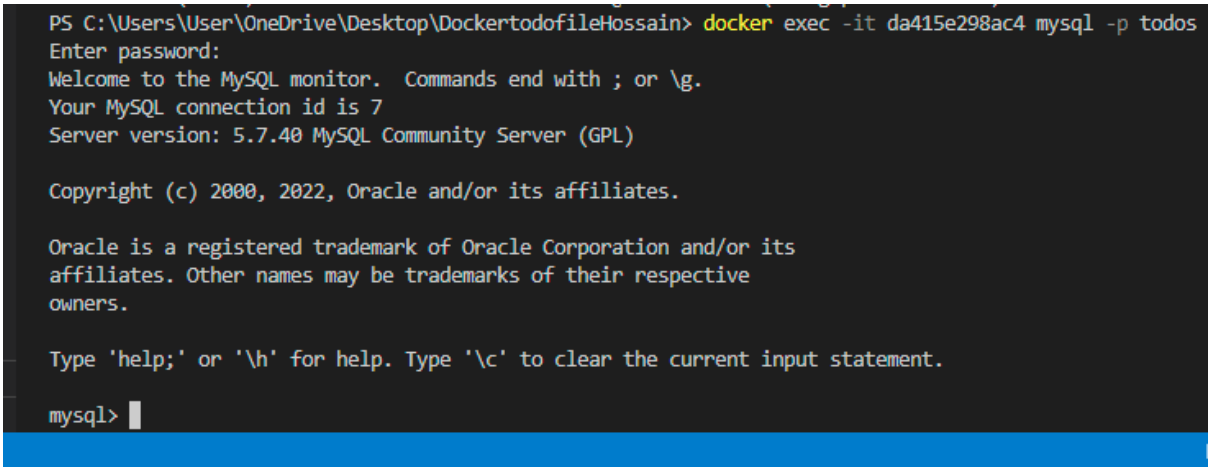
Activate Windows
Go to Settings to activate

Ln 1, Col 1 Spaces: 2 UTF-8 LF

Docker Images exist:

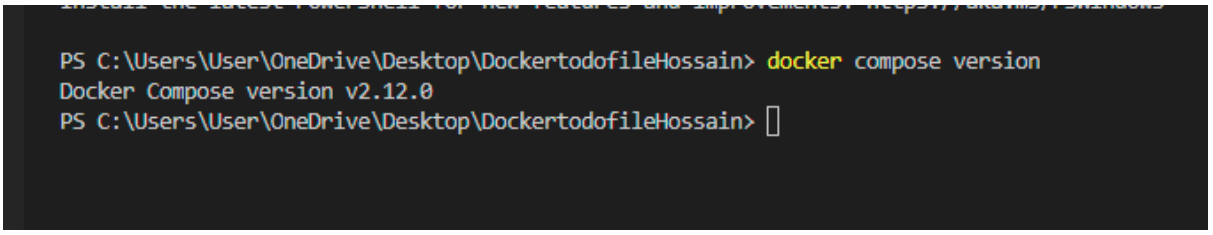
```
PS C:\Users\User\OneDrive\Desktop\DockertodofileHossain> ls


    Directory: C:\Users\User\OneDrive\Desktop\DockertodofileHossain


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        29-Oct-22   2:07 PM                getting-started-master
d-----        29-Oct-22   4:36 PM                node_modules
-a----        29-Oct-22   4:36 PM             86 yarn.lock


PS C:\Users\User\OneDrive\Desktop\DockertodofileHossain>
```

 # Connect to the mysql database and prove that the items are being written to the database. Remember, the password is secret.

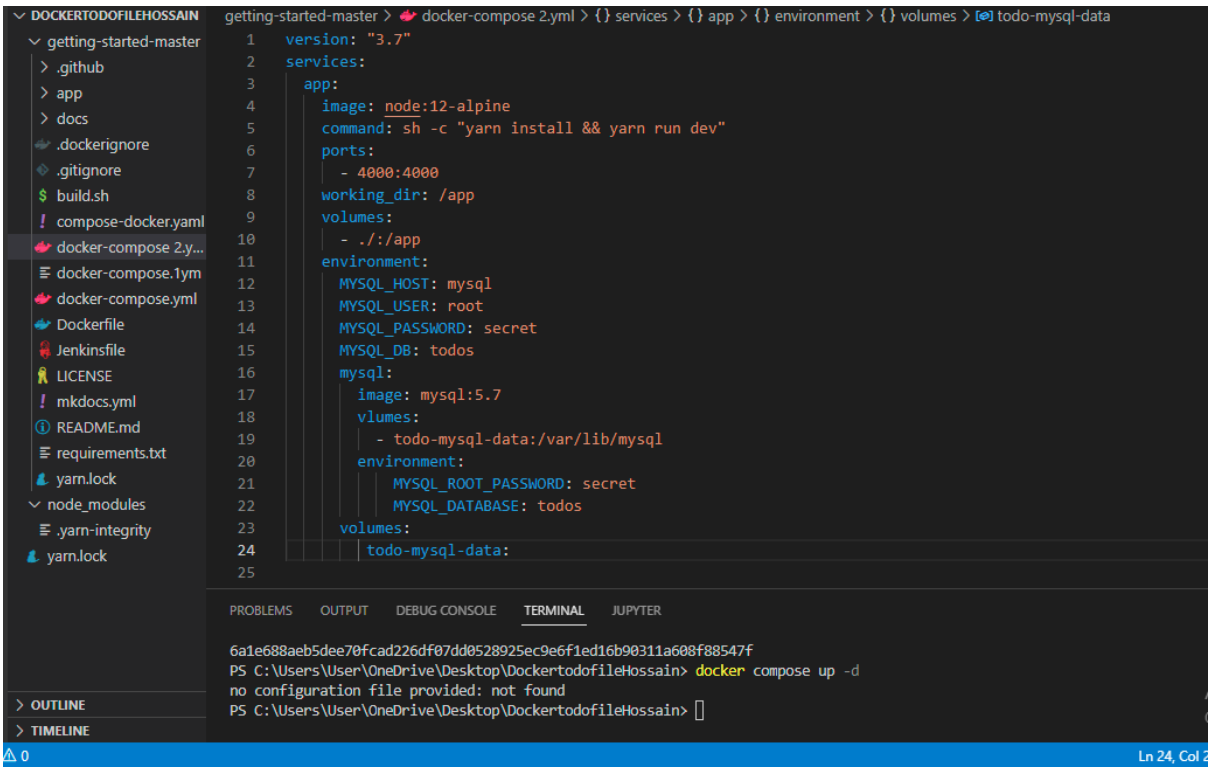


Part 2

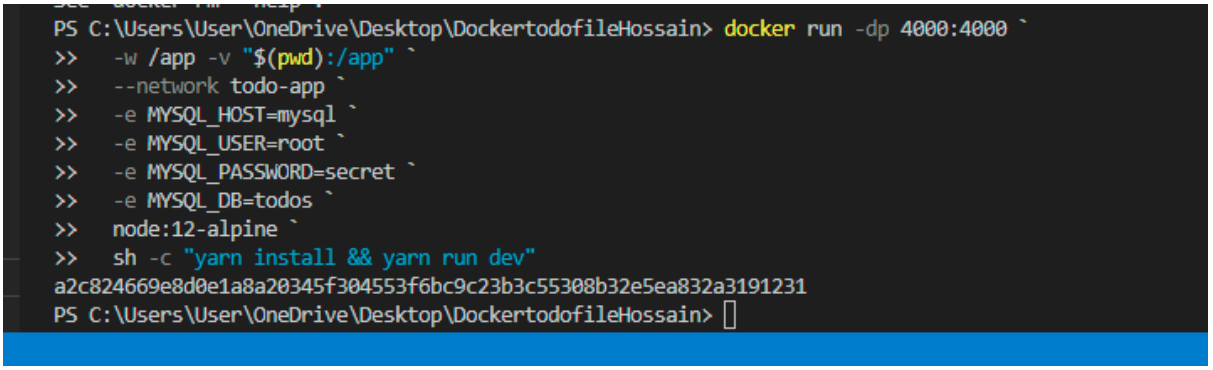
Install Docker Compose



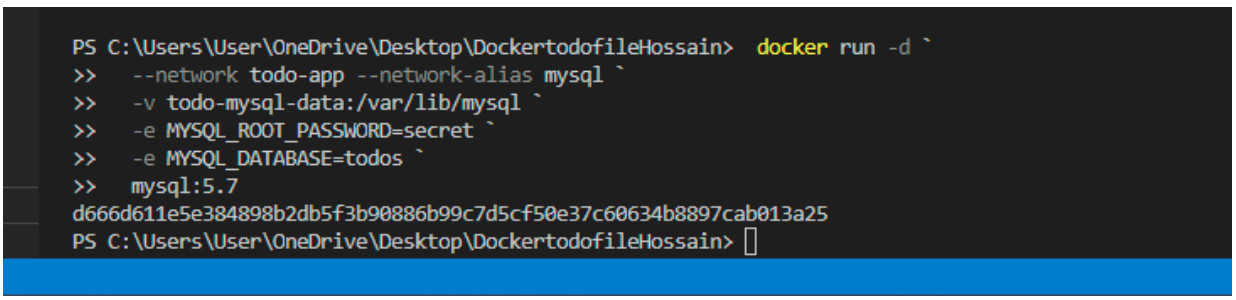
Create the Compose file



Define the app service

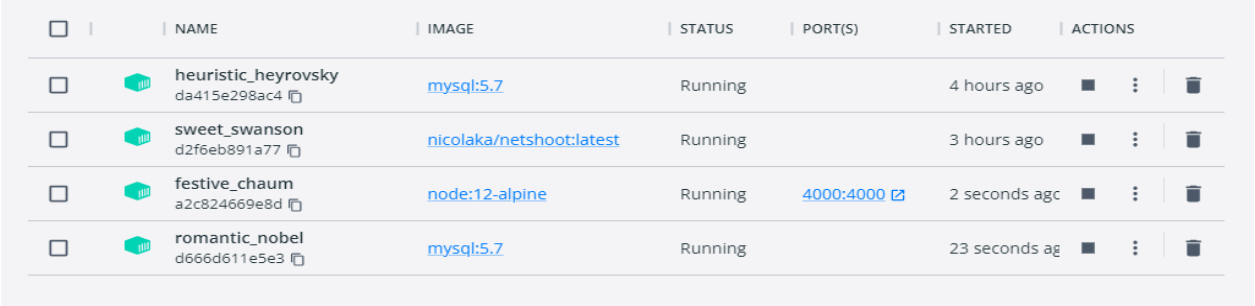


Define the MySQL service



Run the application stack

```
mysql> docker compose up -d
    -> Creating networking "app_default"with the default driver
    -> Creating volume "app_todo-mysql-data" with the default driver
    -> Creating app_app_1  ...done
    -> Creating app_mysql_1 ..done
```



NAME	IMAGE	STATUS	PORT(S)	STARTED	ACTIONS
heuristic_heyrovsky da415e298ac4	mysql:5.7	Running		4 hours ago	
sweet_swanson d2f6eb891a77	nicolaka/netshoot:latest	Running		3 hours ago	
festive_chaum a2c824669e8d	node:12-alpine	Running	4000:4000	2 seconds agc	
romantic_nobel d666d611e5e3	mysql:5.7	Running		23 seconds ag	

Github_link

https://github.com/Alamin2022/Docker-HW-1-Cloud-systems.