Learning to pull images from the docker hub

Directly go to the docker hub for example want to download latest version then no need of mention version or else mention.Then start the docker desktop application and in terminal run the command of docker pull.

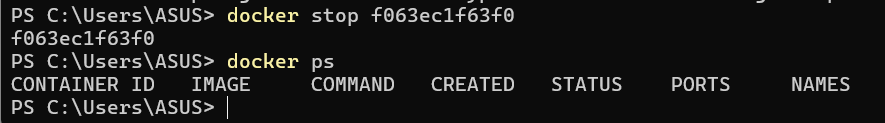


To check which all images are there can use docker images command



### Running a PostgreSQL Container

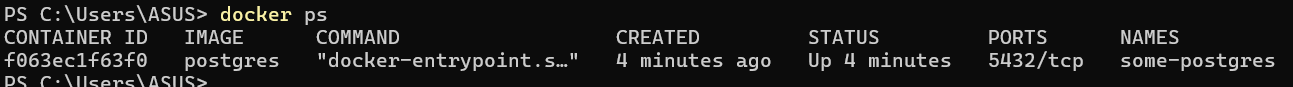
#### 

* docker run: This command is used to create and start a new container from a Docker image.
* --name some-postgres: This option names the container some-postgres. Naming your containers can make it easier to manage and reference them later.
* -e POSTGRES\_PASSWORD=mysecretpassword: This sets an environment variable POSTGRES\_PASSWORD to mysecretpassword. PostgreSQL requires a password to be set for the postgres user when the database is started.
* -d: This option runs the container in detached mode, which means it runs in the background.
* postgres: This is the name of the Docker image to use, which in this case is the official PostgreSQL image from Docker Hub.
* This long string is the container ID, a unique identifier for the running container.
* For stopping it to run we need to write docker stop postgres
* 
* Can be started again by mentioning id

If we do only docker run postgres then it will show its running instead of showing id and can be stop by doing ctrl-c.

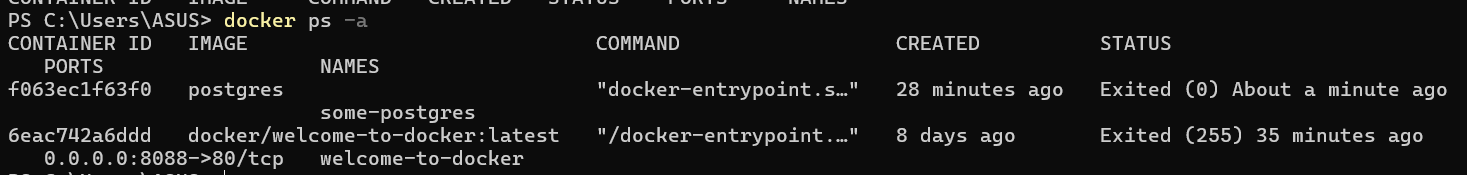
### Listing Running Containers

* docker ps: This command lists all running Docker containers. By default, it shows only the containers that are currently running.

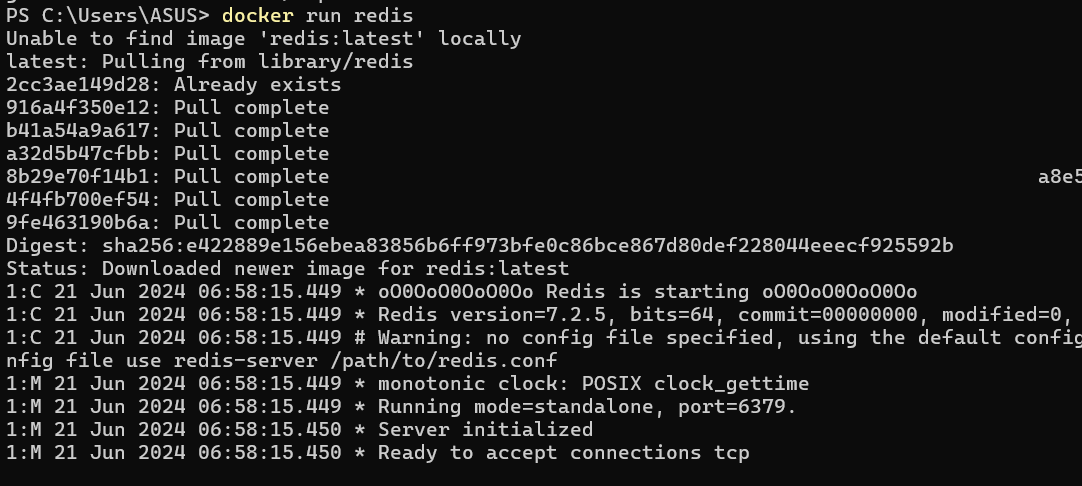


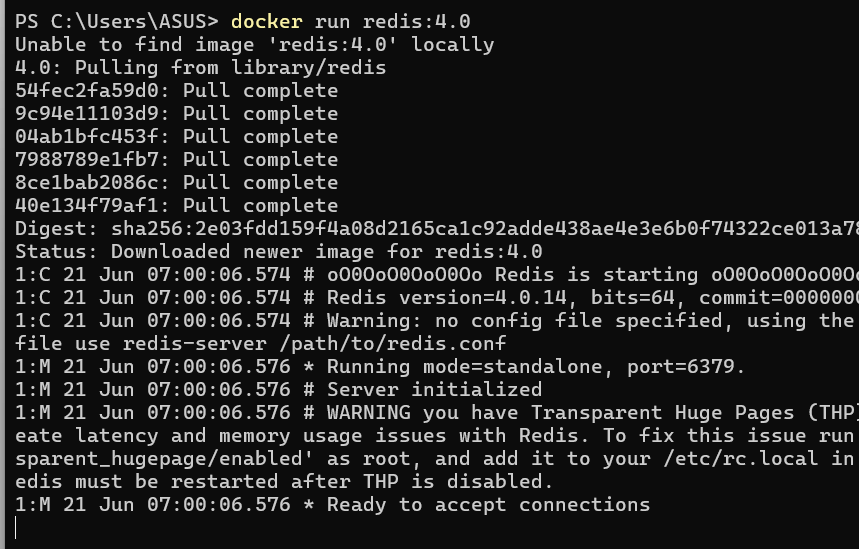
#### Explanation of the Output:

* CONTAINER ID: The unique identifier for the container, in this case, f063ec1f63f0.
* IMAGE: The Docker image used to create the container, which is postgres.
* COMMAND: The command that is being run in the container. Here, it is docker-entrypoint.sh for PostgreSQL.
* CREATED: When the container was created. In this example, it was created 4 minutes ago.
* STATUS: The current status of the container, which is Up 4 minutes indicating that it has been running for 4 minutes.
* PORTS: The ports that are exposed and mapped for the container. Here, 5432/tcp indicates that the container is using port 5432 for TCP connections.
* NAMES: The name of the container, which is some-postgres.

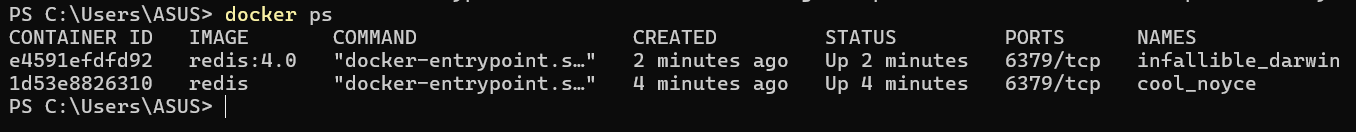
For checking the history and getting list of running and stop containers:  


For running multiple containers:

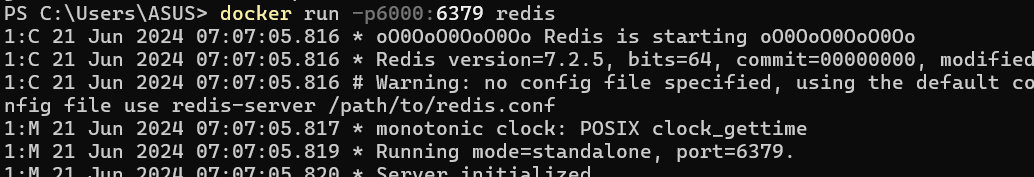


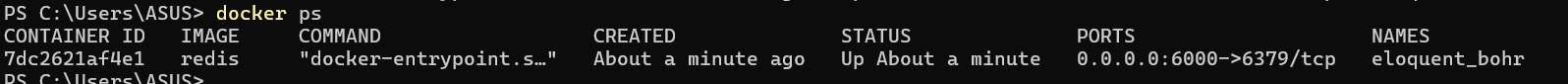


Now we can see both containers running and both are on same port also we can run application on port as be haven’t done binding between container port and host port.

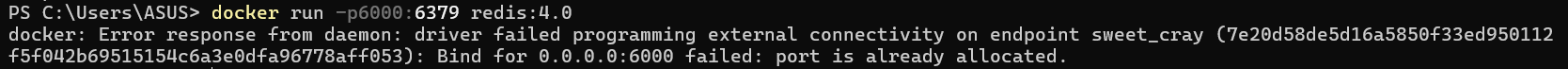


Binding to the host port

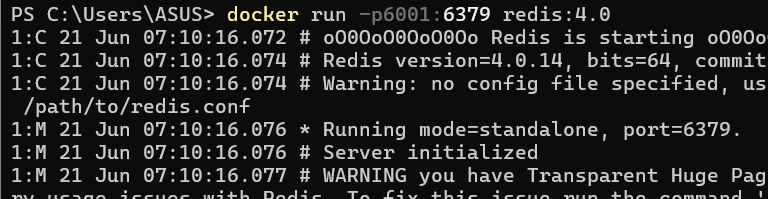


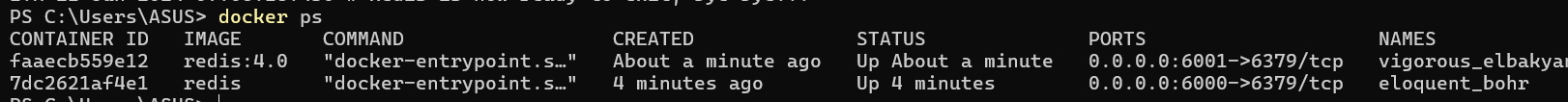


Now cannot allocate same port to other one or else it throws the error



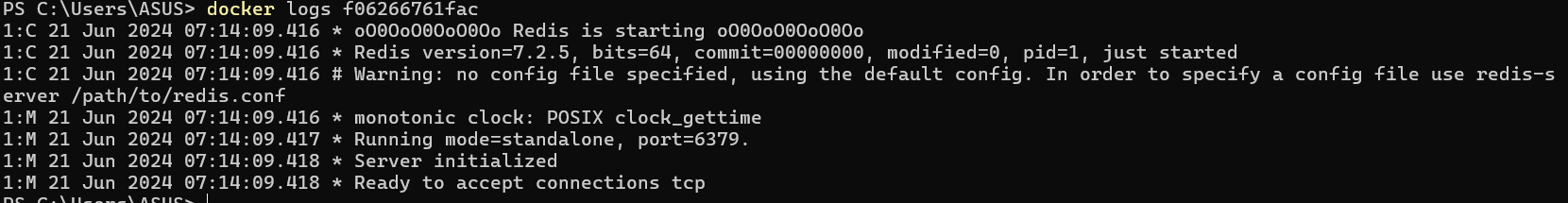
Hence allocating to different port



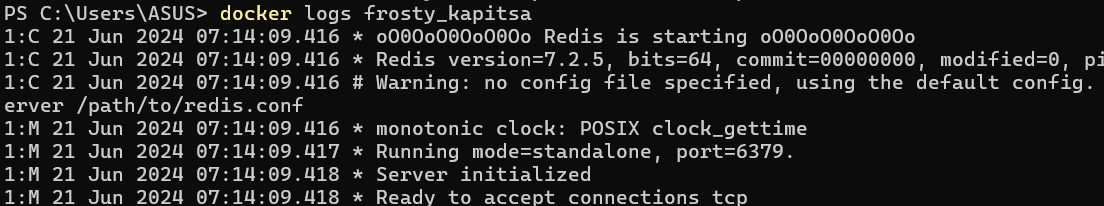


Getting logs:

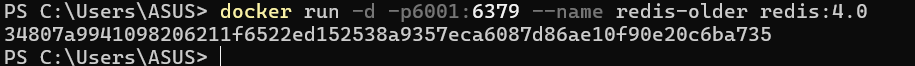
With id

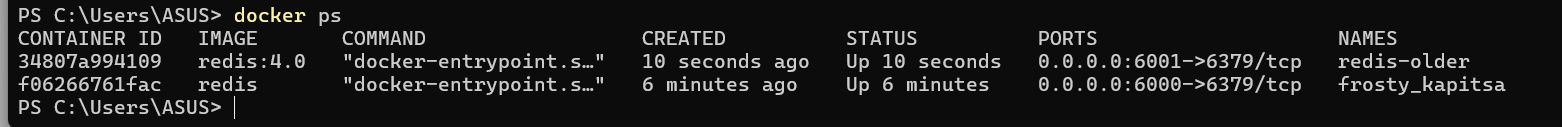


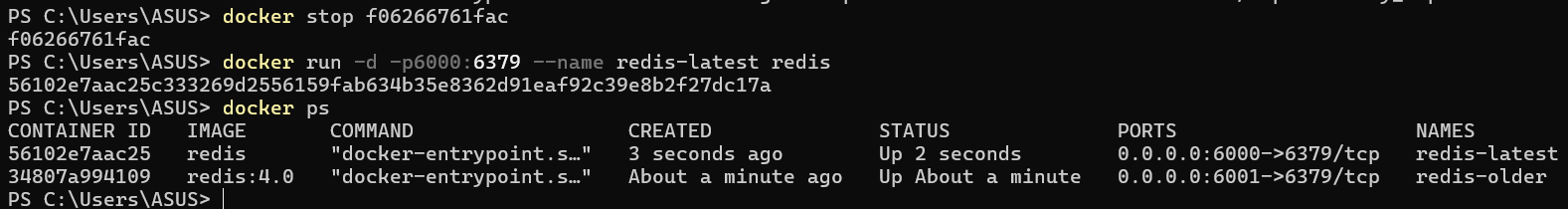
With names



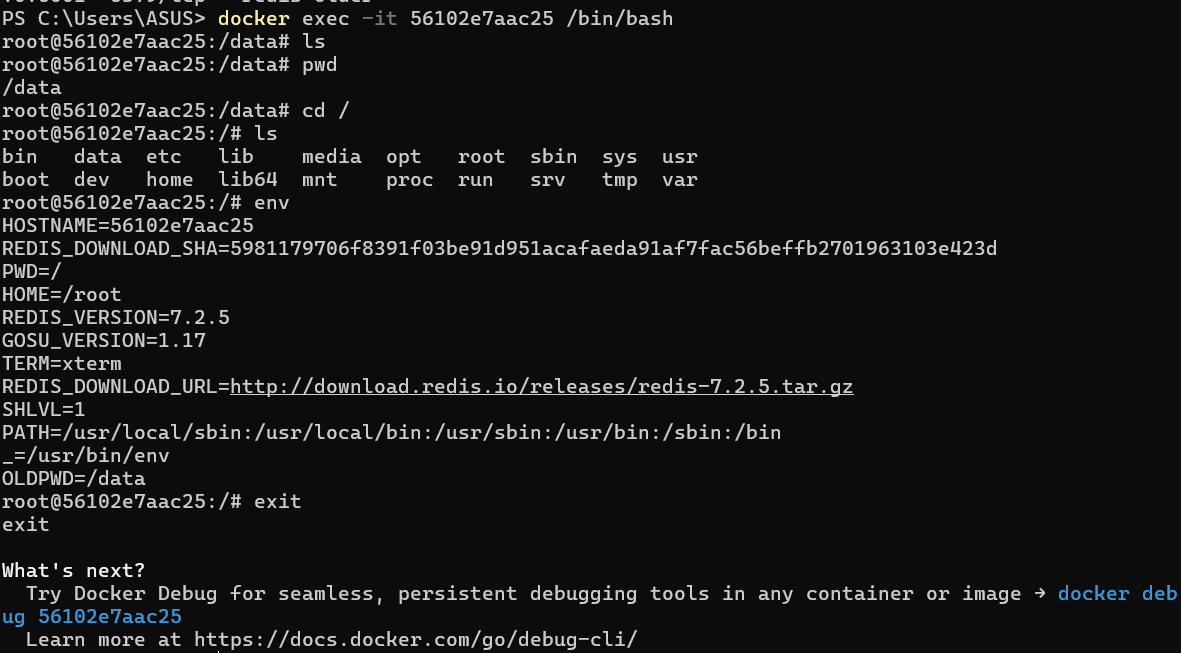
Naming the docker container







The docker exec -it bash command is used to run an interactive shell session within a **running Docker container**.

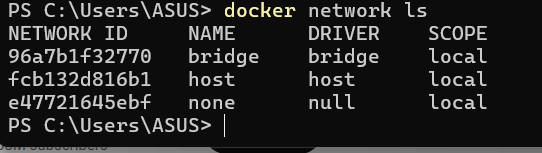


Docker run is used to create a new container from image

Docker start is use to start the container that already exist and is stopped previously

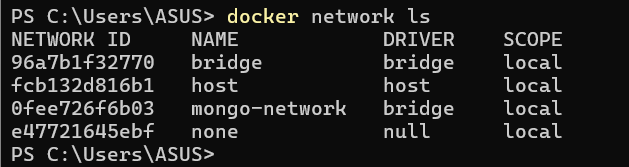
Example project :

Using javscript and nodejs and connecting it with mongodbexpress for backend.

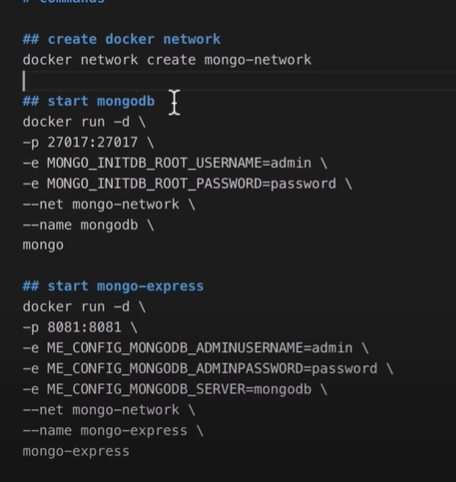


Creating mongo-network





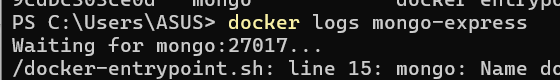
Creating containers that uses this network

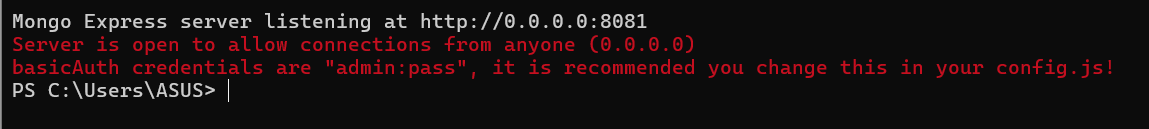


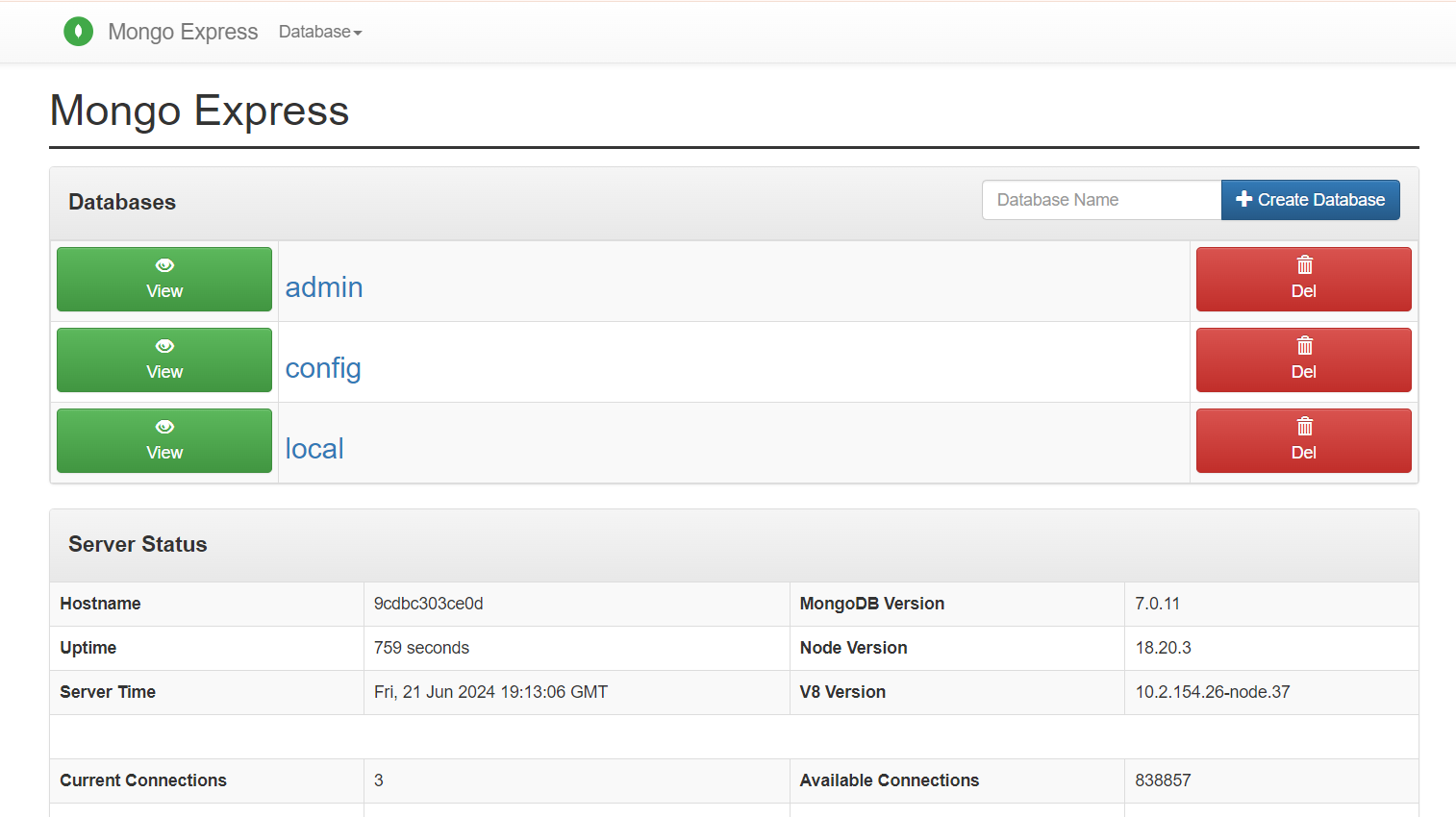
For mongo-express



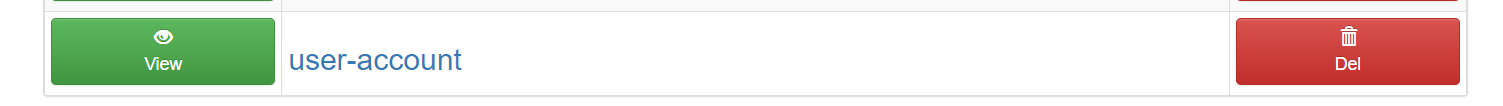
Started mongo-express server





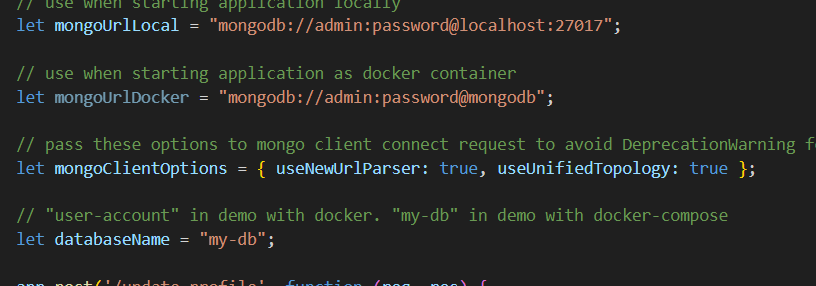


Created user account database:

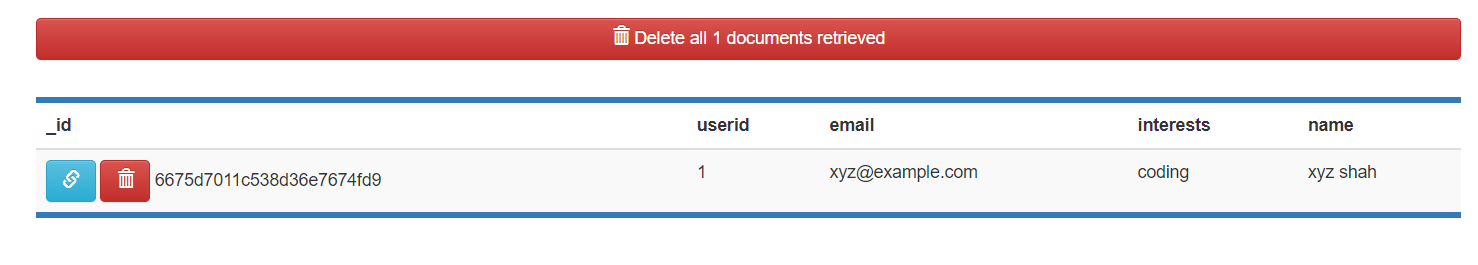


Now connecting with nodejs

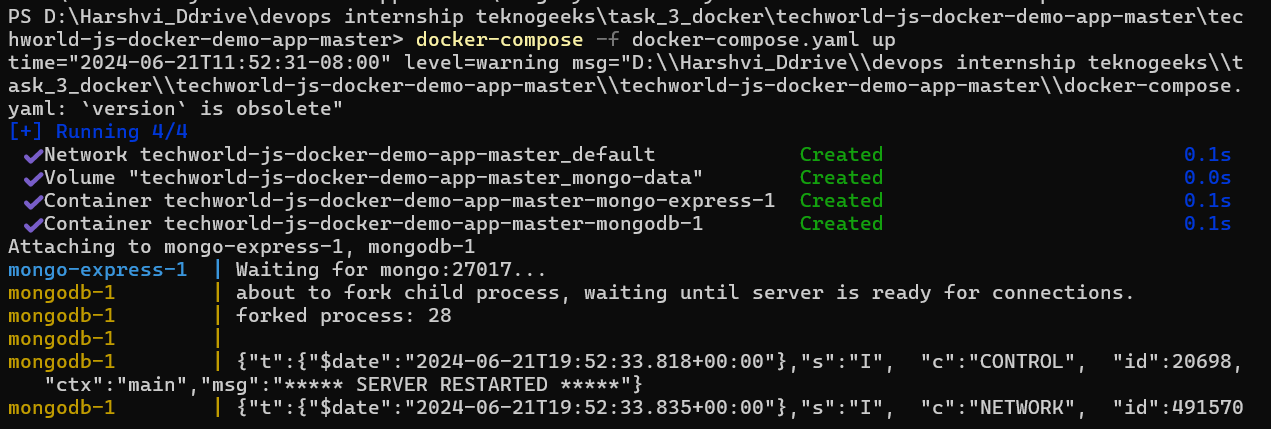
(make changes in code)

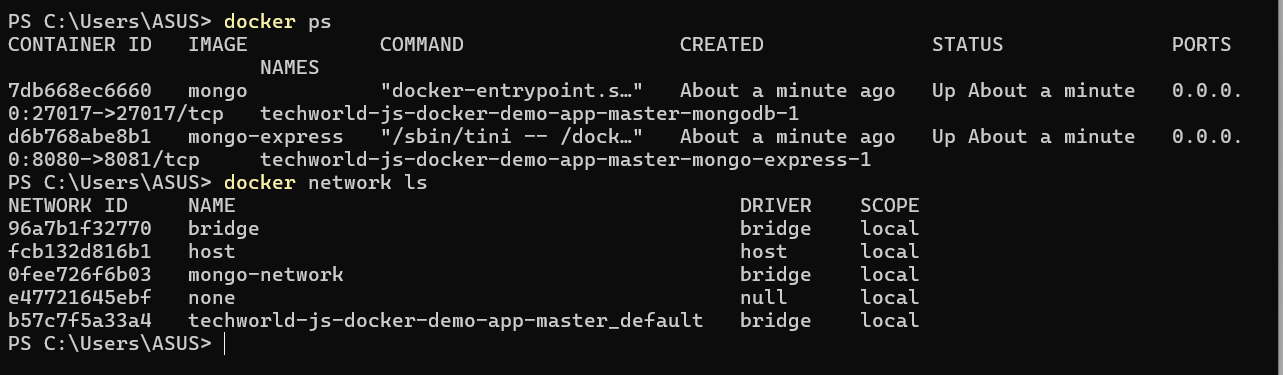


Data inserted

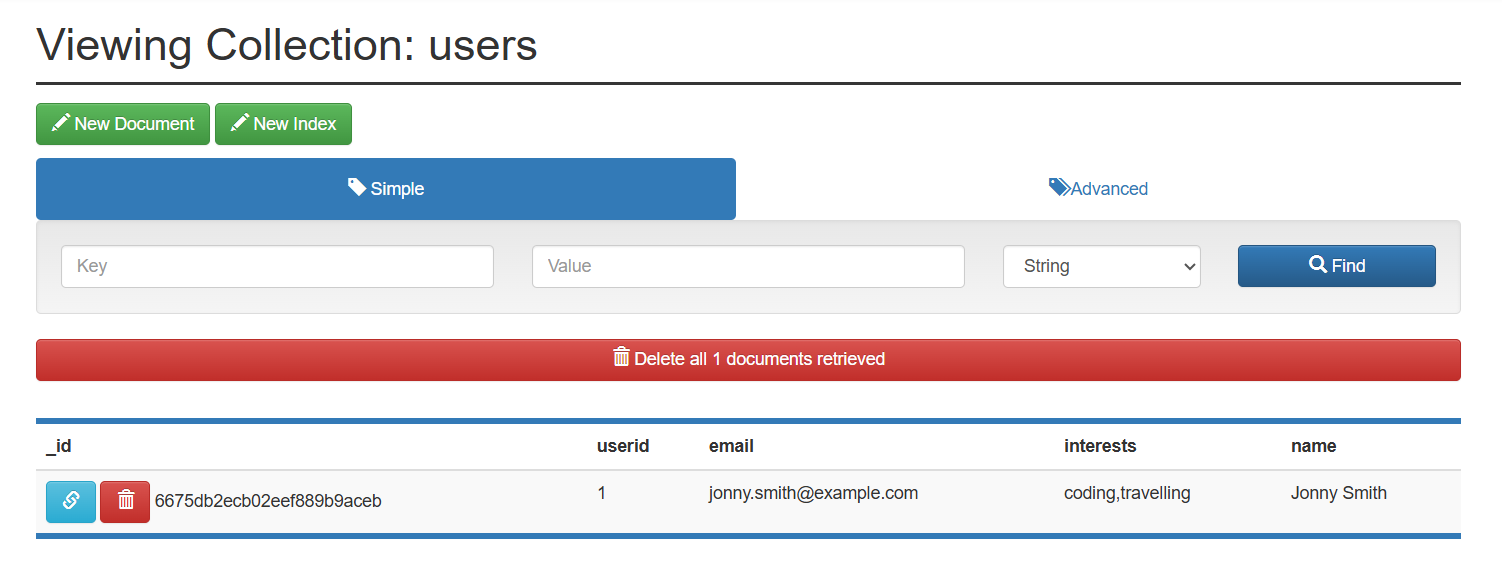


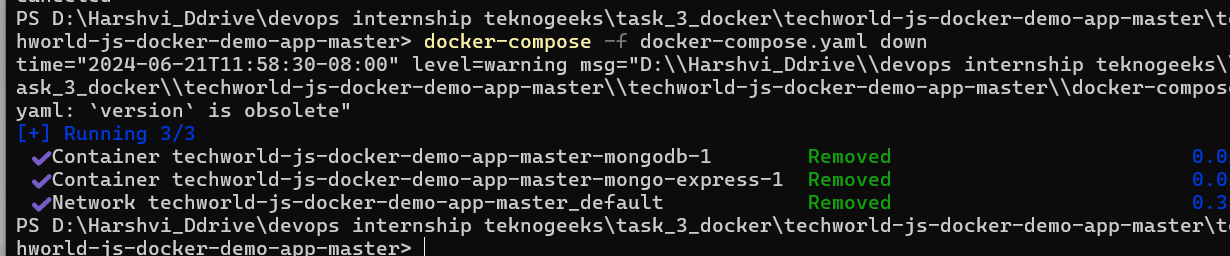
Docker Compose:



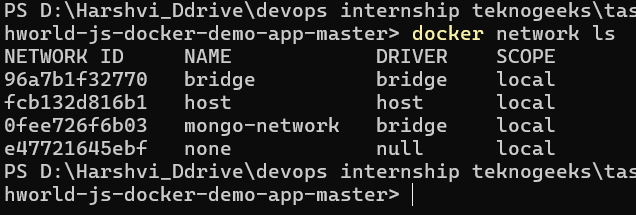


Running on port 8080





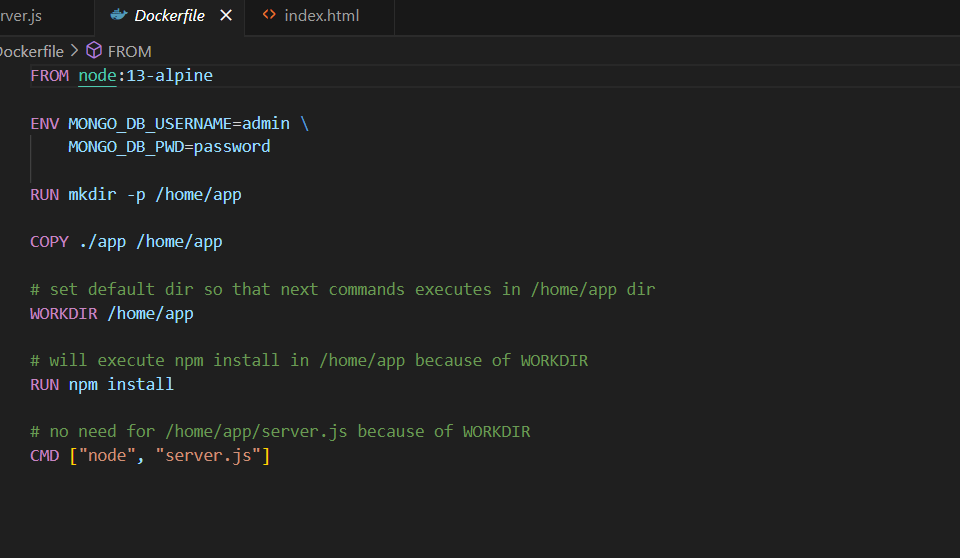
Shut down the containers



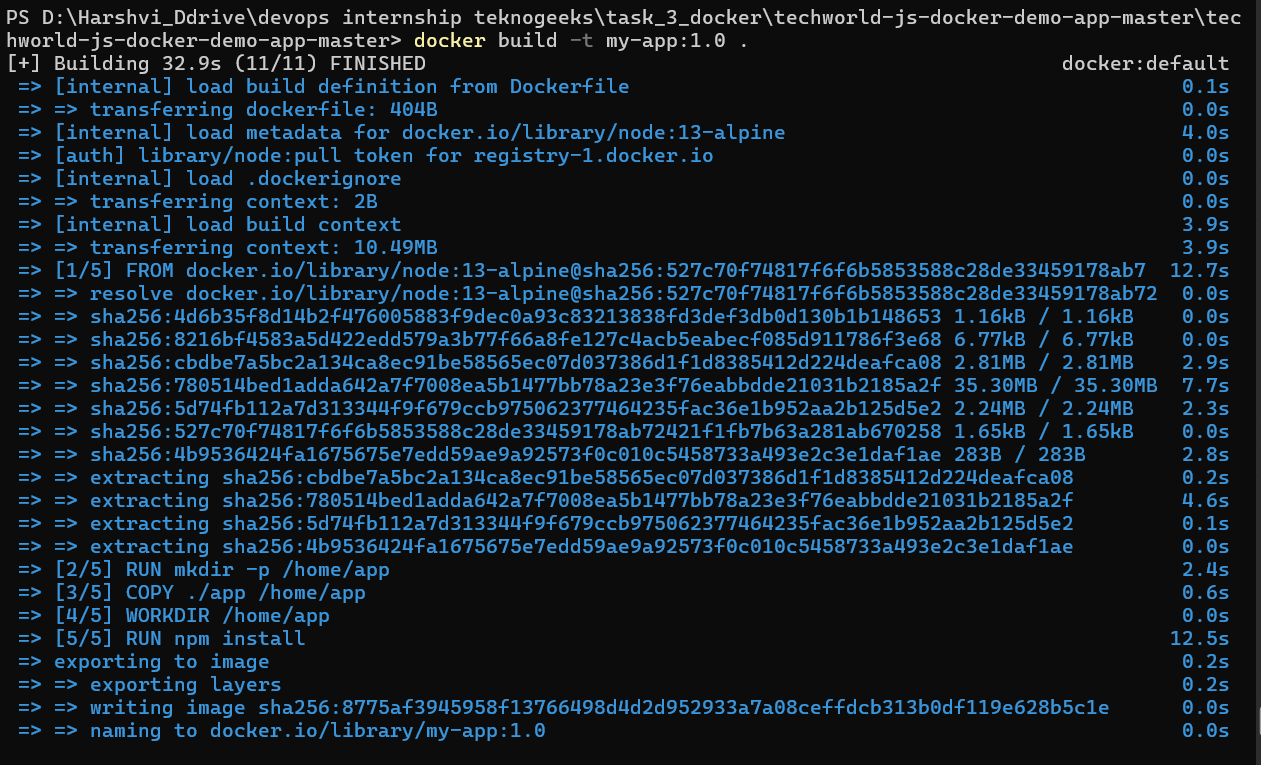
Network also removed

Docker file:Blue print for docker image

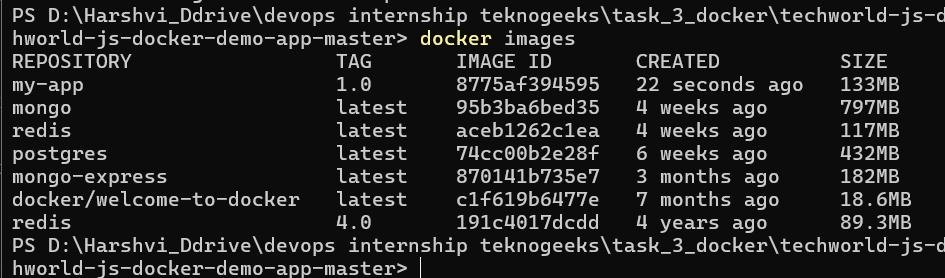
First line for docker file is always “From image\_name”



Now building docker image



My-app image is created



Docker run

