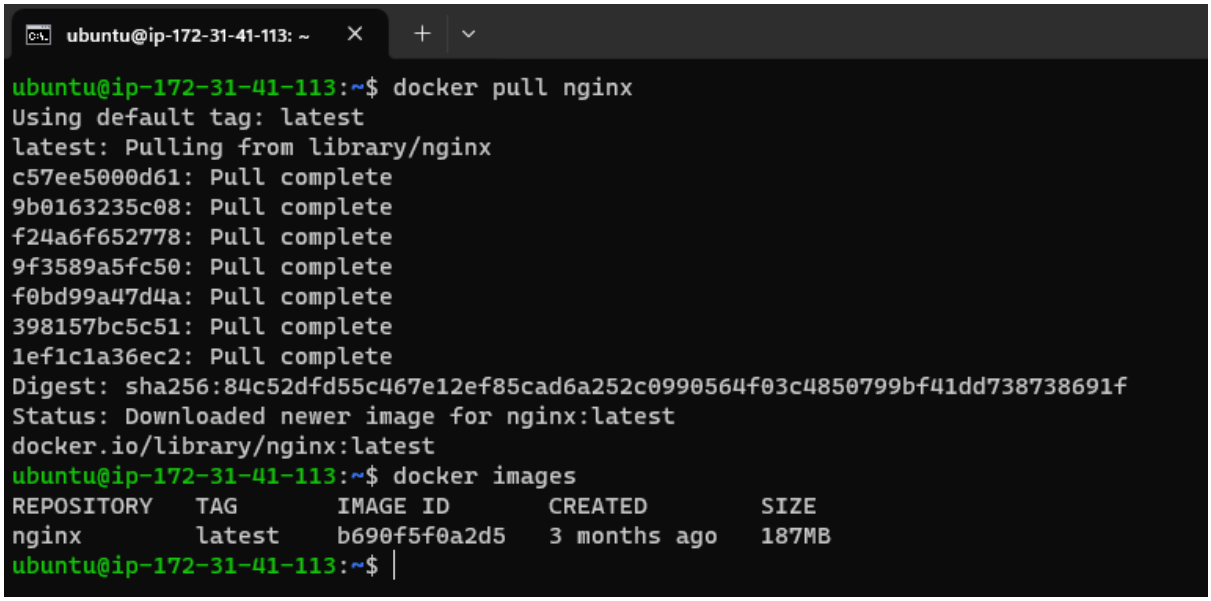
😊 Docker Logs

1. In this lab you are going to learn about Docker logs or say logs of your docker container.
2. Again, come back to Ubuntu session. Login to it.
3. First you need to pull nginx image. So, run a command for that.



4. We have the image Now we can see detailed information about an image by using a command called docker inspect. Now, this is not logs, this is metadata of the image in JSON format.
5. There is a lot of information about this you can see it yourself.

   docker inspect nginx

```
ubuntu@ip-172-31-41-113:~$ docker inspect nginx
[
    {
        "Id": "sha256:b690f5f0a2d535cee5e08631aa508fef339c43bb91d5b1f7d77a1a05cea021a8",
        "RepoTags": [
            "nginx:latest"
        ],
        "RepoDigests": [
            "nginx@sha256:84c52dfd55c467e12ef85cad6a252c0990564f03c4850799bf41dd738738691f"
        ],
        "Parent": "",
        "Comment": "buildkit.dockerfile.v0",
        "Created": "2023-10-24T22:44:45Z",
        "Container": "",
        "ContainerConfig": {
            "Hostname": "",
            "Domainname": "",
            "User": "",
            "AttachStdin": false,
            "AttachStdout": false,
            "AttachStderr": false,
            "Tty": false,
            "OpenStdin": false,
            "StdinOnce": false,
            "Env": null,
            "Cmd": null,
            "Image": "",
            "Volumes": null,
            "WorkingDir": "",
            "Entrypoint": null,
            "OnBuild": null,
            "Labels": null
```

6. Here is, docker run -d that is running in the background, detached mode and capital P is for the port mapping. So automatically it will pick up the host port.
7. You can see that the image is running and its name is hardcore_carson.

   docker run -d -P nginx
   docker ps

```
ubuntu@ip-172-31-41-113: ~
ubuntu@ip-172-31-41-113:~$ docker run -d -P nginx
1f65222ca5cfa790360ee36ef9f7b56527c3c6bd885961fe0f4c9c2212f80d18
ubuntu@ip-172-31-41-113:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS         PORTS                                     NAMES
1f65222ca5cf   nginx     "/docker-entrypoint.…"   4 seconds ago   Up 3 seconds   0.0.0.0:32768->80/tcp, :::32768->80/tcp   hardcore_carson
ubuntu@ip-172-31-41-113:~$
```

8. So, now if you want to see its output you can run the logs command.

   docker logs hardcore_carson

```
ubuntu@ip-172-31-41-113:~$ docker logs hardcore_carson
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2024/02/08 17:20:41 [notice] 1#1: using the "epoll" event method
2024/02/08 17:20:41 [notice] 1#1: nginx/1.25.3
2024/02/08 17:20:41 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2024/02/08 17:20:41 [notice] 1#1: OS: Linux 6.2.0-1017-aws
2024/02/08 17:20:41 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2024/02/08 17:20:41 [notice] 1#1: start worker processes
2024/02/08 17:20:41 [notice] 1#1: start worker process 28
ubuntu@ip-172-31-41-113:~$
```

9. So, now you do not give -d in the command so the output is different.
10. Basically, -d allows image to run in the background. So, if it is not running in the background then obviously it will run in the foreground and because of this it is taking up the shell.
11. So, to get your shell back you need to do ctrl C on your keyboard which will also kill the process.
12. Now if you run docker ps you can see that your image is in exited state.

docker run -P nginx

```
ubuntu@ip-172-31-41-113:~$ docker run -P nginx
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2024/02/08 17:24:33 [notice] 1#1: using the "epoll" event method
2024/02/08 17:24:33 [notice] 1#1: nginx/1.25.3
2024/02/08 17:24:33 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2024/02/08 17:24:33 [notice] 1#1: OS: Linux 6.2.0-1017-aws
2024/02/08 17:24:33 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2024/02/08 17:24:33 [notice] 1#1: start worker processes
2024/02/08 17:24:33 [notice] 1#1: start worker process 29



^C2024/02/08 17:25:02 [notice] 1#1: signal 2 (SIGINT) received, exiting
2024/02/08 17:25:02 [notice] 29#29: exiting
2024/02/08 17:25:02 [notice] 29#29: exit
2024/02/08 17:25:02 [notice] 1#1: signal 17 (SIGCHLD) received from 29
2024/02/08 17:25:02 [notice] 1#1: worker process 29 exited with code 0
2024/02/08 17:25:02 [notice] 1#1: exit
ubuntu@ip-172-31-41-113:~$
```

```
ubuntu@ip-172-31-41-113:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS                     PORTS                                     NAMES
bc98378dcb99   nginx     "/docker-entrypoint.…"   2 minutes ago   Exited (0) 2 minutes ago                                             stupefied_keller
1f65222ca5cf   nginx     "/docker-entrypoint.…"   6 minutes ago   Up 6 minutes               0.0.0.0:32768->80/tcp, :::32768->80/tcp   hardcore_carson
ubuntu@ip-172-31-41-113:~$
```

13. So, the purpose of all this is troubleshooting, when you build your own images, which you will be doing and when you run containers from your own custom-built images, you might make some mistake and your container won't start. How do you figure out what is the problem by looking at the output of the process and that you can do it through docker logs command.
14. Now you are going to run one container in the background and this is going to be MySQL.

    docker run -d -P mysql:5.7

```
ubuntu@ip-172-31-41-113:~$ docker run -d -P mysql:5.7
Unable to find image 'mysql:5.7' locally
5.7: Pulling from library/mysql
20e4dcae4c69: Pull complete
1c56c3d4ce74: Pull complete
e9f03a1c24ce: Pull complete
68c3898c2015: Pull complete
6b95a940e7b6: Pull complete
90986bb8de6e: Pull complete
ae71319cb779: Pull complete
ffc89e9dfd88: Pull complete
43d05e938198: Pull complete
064b2d298fba: Pull complete
df9a4d85569b: Pull complete
Digest: sha256:4bc6bc963e6d8443453676cae56536f4b8156d78bae03c0145cbe47c2aad73bb
Status: Downloaded newer image for mysql:5.7
af989739c6e6ab197ed5674f71f7187bce2933ff08c1cfd12a3e524312deb7c7
```

15. Now if you do a docker ps command you cannot see your container of MySQL. And if you do docker ps -a you can see your container but it is in exited state.

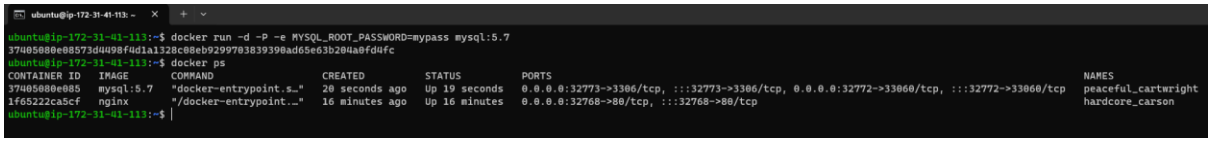
```
ubuntu@ip-172-31-41-113:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS                                     NAMES
1f65222ca5cf   nginx     "/docker-entrypoint.…"   12 minutes ago   Up 12 minutes   0.0.0.0:32768->80/tcp, :::32768->80/tcp   hardcore_carson
ubuntu@ip-172-31-41-113:~$ docker ps -a
CONTAINER ID   IMAGE       COMMAND                  CREATED              STATUS                          PORTS                                     NAMES
af989739c6e6   mysql:5.7   "docker-entrypoint.s…"   About a minute ago   Exited (1) About a minute ago                                             priceless_euler
bc98378dcb99   nginx       "/docker-entrypoint.…"   8 minutes ago        Exited (0) 8 minutes ago                                                  stupefied_keller
1f65222ca5cf   nginx       "/docker-entrypoint.…"   12 minutes ago       Up 12 minutes                   0.0.0.0:32768->80/tcp, :::32768->80/tcp   hardcore_carson
ubuntu@ip-172-31-41-113:~$
```

16. If you do a logs command of this container you can find the error.

```
ubuntu@ip-172-31-41-113:~$ docker logs priceless_euler
2024-02-08 17:32:11+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 5.7.44-1.el7 started.
2024-02-08 17:32:11+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
2024-02-08 17:32:11+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 5.7.44-1.el7 started.
2024-02-08 17:32:12+00:00 [ERROR] [Entrypoint]: Database is uninitialized and password option is not specified
    You need to specify one of the following as an environment variable:
    - MYSQL_ROOT_PASSWORD
    - MYSQL_ALLOW_EMPTY_PASSWORD
    - MYSQL_RANDOM_ROOT_PASSWORD
ubuntu@ip-172-31-41-113:~$
```

17. Now you need to run this command and specify a password to it.

    docker run -d -P -e MYSQL_ROOT_PASSWORD=mypass mysql:5.7



18. After all this you need to remove everything for the next lab.