

```
C:\ Command Prompt
Microsoft Windows [Version 10.0.22000.376]
(c) Microsoft Corporation. All rights reserved.

C:\Users\amalu>docker -v
Docker version 20.10.11, build dea9396

C:\Users\amalu>docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
getting-started-master  latest   9f6dd5bd7a85  23 hours ago  454MB
getting-started      latest   66be9fb723b6  23 hours ago  28.8MB
<none>              <none>   52e8e7b2f668  44 hours ago  454MB
<none>              <none>   b6d52e3c4d35  44 hours ago  28.8MB
9656562186/new      latest   7ac7352abb28  44 hours ago  454MB
9656562186/nework123  latest   a1af3eb26a49  8 days ago   454MB
9656562186/nework123  latest   a1af3eb26a49  8 days ago   454MB
9656562186/nework123  latest   07cf9896679  9 days ago   454MB
9656562186/nework123  latest   d00d8b860e77  10 days ago  403MB
9656562186/nework123  latest   4e831b2291d2  11 days ago  403MB
9656562186/getting-started  latest   b613aea483e3  11 days ago  403MB
9656562186/getting-started  latest   bd7f7fcfa0bb  2 weeks ago  403MB
9656562186/getting-started  latest   6fd104ec5f5  2 weeks ago  403MB
mysql               5.7     c20987f18b13  2 weeks ago  448MB
mysql               latest   3218b58a490ce  2 weeks ago  516MB
node                12-alpine  106bb59759ad  3 weeks ago  89.5MB
9656562186/docker101tutorial  latest   6c72795c3de8  3 weeks ago  28.5MB
docker101tutorial    latest   6c72795c3de8  3 weeks ago  28.5MB
redis               latest   aea9b698d7d1  5 weeks ago  113MB
docker/getting-started  latest   26d80cd96d69  5 weeks ago  28.5MB
alpine              latest   c059bfaa849c  6 weeks ago  5.59MB
alpine/git           latest   c6b70534b534  7 weeks ago  27.4MB
nicolaka/netshoot    latest   f4c8dceca780  7 weeks ago  432MB
docker/desktop-kubernetes  kubernetes-v1.22.4-cni-v0.8.5-critools-v1.17.0-debian  v1.22.4  493a106d3678  7 weeks ago  294MB
k8s.gcr.io/kube-apiserver  v1.22.4  8a5cc299272d  7 weeks ago  128MB
k8s.gcr.io/kube-controller-manager  v1.22.4  0ce02f92d3ed  7 weeks ago  122MB
k8s.gcr.io/kube-scheduler  v1.22.4  721ba97f54a6  7 weeks ago  52.7MB
k8s.gcr.io/kube-proxy    v1.22.4  edeff37e4a382  7 weeks ago  104MB
ubuntu              latest   ba6acc4cd29  2 months ago  72.8MB
k8s.gcr.io/etcd        3.5.0-0  004811815584  6 months ago  295MB
k8s.gcr.io/coredns     coredns  8d147537fb7d  7 months ago  47.6MB
alpine              v1.8.4  8d147537fb7d  7 months ago  47.6MB
docker/desktop-vpnkit-controller  v2.0     8c2c38aa676e  8 months ago  21MB
```

```
C:\ Command Prompt
ubuntu              latest   ba6acc4cd29  2 months ago  72.8MB
k8s.gcr.io/etcd        3.5.0-0  004811815584  6 months ago  295MB
k8s.gcr.io/coredns     coredns  8d147537fb7d  7 months ago  47.6MB
docker/desktop-vpnkit-controller  v2.0     8c2c38aa676e  8 months ago  21MB
dockerd/desktop-storage-provisioner  v2.0     99f89471f470  8 months ago  41.9MB
k8s.gcr.io/pause       3.5     ed210e3e4a5b  9 months ago  683kB
alpine              3.5     f80194ae2e0c  2 years ago  4MB

C:\Users\amalu>docker run -dp 80:80 docker/getting-started
561fe79d84041ff6b00bc4b7d6eeb1cfae015882f44849534084fb2cea3699f6
docker: Error response from daemon: Ports are not available: listen tcp 0.0.0.0:80: bind: An attempt was made to access a socket in a way forbidden by its access permission
s.

C:\Users\amalu>docker image ls
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
getting-started-master  latest   9f6dd5bd7a85  23 hours ago  454MB
getting-started      latest   66be9fb723b6  23 hours ago  28.8MB
<none>              <none>   52e8e7b2f668  44 hours ago  454MB
<none>              <none>   b6d52e3c4d35  44 hours ago  28.8MB
9656562186/new      latest   7ac7352abb28  44 hours ago  454MB
9656562186/nework123  latest   a1af3eb26a49  8 days ago   454MB
9656562186/nework123  latest   a1af3eb26a49  8 days ago   454MB
9656562186/nework123  latest   07cf9896679  9 days ago   454MB
9656562186/nework123  latest   d00d8b860e77  10 days ago  403MB
9656562186/nework123  latest   4e831b2291d2  11 days ago  403MB
9656562186/getting-started  latest   b613aea483e3  11 days ago  403MB
9656562186/getting-started  latest   bd7f7fcfa0bb  2 weeks ago  403MB
9656562186/getting-started  latest   6fd104ec5f5  2 weeks ago  403MB
mysql               5.7     c20987f18b13  2 weeks ago  448MB
mysql               latest   3218b58a490ce  2 weeks ago  516MB
node                12-alpine  106bb59759ad  3 weeks ago  89.5MB
9656562186/docker101tutorial  latest   6c72795c3de8  3 weeks ago  28.5MB
docker101tutorial    latest   6c72795c3de8  3 weeks ago  28.5MB
redis               latest   aea9b698d7d1  5 weeks ago  113MB
docker/getting-started  latest   26d80cd96d69  5 weeks ago  28.5MB
alpine              latest   c059bfaa849c  6 weeks ago  5.59MB
alpine/git           latest   c6b70534b534  7 weeks ago  27.4MB
nicolaka/netshoot    latest   f4c8dceca780  7 weeks ago  432MB
docker/desktop-kubernetes  kubernetes-v1.22.4-cni-v0.8.5-critools-v1.17.0-debian  v1.22.4  493a106d3678  7 weeks ago  294MB
k8s.gcr.io/kube-apiserver  v1.22.4  8a5cc299272d  7 weeks ago  128MB
```

```
Ex Command Prompt
E:\Users\amalu>docker image history redis
IMAGE          CREATED        CREATED BY
sea9b698d7d1  5 weeks ago   /bin/sh -c #(nop)  CMD ["redis-server"]
       5 weeks ago   /bin/sh -c #(nop) EXPOSE 6379
<missing>      5 weeks ago   /bin/sh -c #(nop) ENTRYPOINT ["docker-entry...
<missing>      5 weeks ago   /bin/sh -c #(nop) COPY file:df205a0ef5e6df89...
<missing>      5 weeks ago   /bin/sh -c #(nop) WORKDIR /data
<missing>      5 weeks ago   /bin/sh -c #(nop) VOLUME [/data]
<missing>      5 weeks ago   /bin/sh -c mkdir -p /data && chown redis:redis ...
<missing>      5 weeks ago   /bin/sh -c set -eu; savedAptMark="$(apt-m...
<missing>      5 weeks ago   /bin/sh -c #(nop) ENV REDIS_DOWNLOAD_SHA5b...
<missing>      5 weeks ago   /bin/sh -c #(nop) ENV REDIS_DOWNLOAD_URL=ht...
<missing>      5 weeks ago   /bin/sh -c #(nop) ENV REDIS_VERSION=6.2.6
<missing>      5 weeks ago   /bin/sh -c set -eu; savedAptMark="$apt-m...
<missing>      5 weeks ago   /bin/sh -c #(nop) ENV GOSU_VERSION=1.12
<missing>      5 weeks ago   /bin/sh -c groupadd -r -g 999 redis && usera...
<missing>      5 weeks ago   /bin/sh -c #(nop) CMD ["bash"]
<missing>      5 weeks ago   /bin/sh -c #(nop) ADD file:ece5ff85ca549f0b1...
6c72705c3de8  3 weeks ago
6c72705c3de8  3 weeks ago
aea9b698d7d1  5 weeks ago
26d80c9d6d69  5 weeks ago
c059bfaa849c  6 weeks ago
c6b76534b534  7 weeks ago
f4c8dceca788  7 weeks ago
493a106d3678  7 weeks ago
8a5cc299272d  7 weeks ago
0ce02f92d3e4  7 weeks ago
721ba97f54a6  7 weeks ago
edeff87e4802  7 weeks ago
ba6accedd29  2 months ago
004811815584  6 months ago
8d147537fb7d  7 months ago
8c2c38aa676e  8 months ago
99f89471f479  8 months ago
ed210e3e4a5b  9 months ago
f80194ae2e0c  2 years ago

3 weeks ago    28.5MB
3 weeks ago    28.5MB
5 weeks ago   113MB
28.5MB
5.59MB
27.4MB
432MB
294MB
128MB
122MB
52.7MB
184MB
72.8MB
295MB
47.6MB
21MB
41.9MB
683kB
4MB

C:\Users\amalu>
```

Docker for Beginners - Linux

training.play-with-docker.com/beginner-linux/#Task\_0

**Play with Docker classroom**

If the commandline doesn't appear in the terminal, make sure popups are enabled or try resizing the browser window.

```
# is HIGHLY! discouraged. Any consequences of doing so are
# completely the user's responsibilites.
#
# The PWD team.
#####
[node1] (local) root@192.168.0.13 ~
$ git clone https://github.com/docker-samples/linux_tweet_app
Cloning into 'linux_tweet_app'...
remote: Enumerating objects: 14, done.
remote: Total 14 (delta 0), reused 0 (delta 0), pack-reused 14
Receiving objects: 100% (14/14), 10.76 KiB | 10.76 MiB/s, done.
Resolving deltas: 100% (5/5), done.
[node1] (local) root@192.168.0.13 ~
$ docker container run alpine hostname
Unable to find image 'alpine:latest' locally
latest: Pulling from library/alpine
59bf1c3509f3: Pull complete
Digest: sha256:2la3deaa0d32a8057914f36584b5280d2e5ecc984380bc0118285c70fa8c9300
Status: Downloaded newer image for alpine:latest
7359782513c7
```

1. Run the following command in your Linux console.

```
docker container run alpine hostname
```

The output below shows that the `alpine:latest` image could not be found locally. When this happens, Docker automatically *pulls* it from Docker Hub.

After the image is pulled, the container's hostname is displayed (`888e89a3b36b` in the example below).

Unable to find image 'alpine:latest' locally

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

f t

10:50 08-01-2022

Docker for Beginners - Linux

training.play-with-docker.com/beginner-linux/#Task\_0

**Play with Docker classroom**

If the commandline doesn't appear in the terminal, make sure popups are enabled or try resizing the browser window.

```
9300
Status: Downloaded newer image for alpine:latest
7359782513c7
[node1] (local) root@192.168.0.13 ~
$ docker container ls --all
CONTAINER ID IMAGE COMMAND CREATED STATUS
PORTS NAMES
7359782513c7 alpine "hostname" 8 seconds ago Exited (0) 7 seconds ago
silly_joliot
[node1] (local) root@192.168.0.13 ~
$ docker container run --interactive --tty --rm ubuntu bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
ea362ff368469: Pull complete
Digest: sha256:b5a61709a9a44284d88fb12e5c48db0409cfad5b69d4ff8224077c57302df9cf
Status: Downloaded newer image for ubuntu:latest
root@d428025088594:/# ls /
bin dev home lib32 libx32 mnt proc run srv var
boot etc lib lib64 media opt root sbin sys usr
root@d428025088594:/#
```

1. Run a Docker container and access its shell.

```
docker container run --interactive --tty --rm ubuntu bash
```

In this example, we're giving Docker three parameters:

- o `--interactive` says you want an interactive session.
- o `--tty` allocates a pseudo-tty.
- o `--rm` tells Docker to go ahead and remove the container when it's done executing.

The first two parameters allow you to interact with the Docker container.

We're also telling the container to run `bash` as its main process (PID 1).

When the container starts you'll drop into the bash shell with the default prompt `root@<container id>:#`. Docker has attached to the shell in the container, relaying input and output between your local session and the shell session in the container.

2. Run the following commands in the container.

```
ls /
ps aux
```

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

f t

10:50 08-01-2022

Docker for Beginners - Linux

[https://training.play-with-docker.com/beginner-linux/#Task\\_0](https://training.play-with-docker.com/beginner-linux/#Task_0)

Play with Docker classroom

2. Run the following commands in the container.

`ls /`

`ps aux`

`cat /etc/issue`

3. Type `exit` to leave the shell session. This will terminate the `bash` process, causing the container to exit.

`exit`

**Note:** As we used the `--rm` flag when we started the container, Docker removed the container when it stopped. This means if you run another `docker container ls --all` you won't see the Ubuntu container.

4. For fun, let's check the version of our host VM.

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

Ubuntu 20.04.3 LTS | n1

```
$ docker container run --interactive --tty --rm ubuntu bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
ea362f368469: Pull complete
Digest: sha256:b5a61709a9a44284d8fb12e5c48db0409cfad5b69d4ff8224077c57302d
f9cf
Status: Downloaded newer image for ubuntu:latest
root@d42802588594:/# ls /
bin dev home lib32 libx32 mnt proc run srv var
boot etc lib lib64 media opt root sbin sys usr
root@d42802588594:/# ps aux
USER PID %CPU %MEM VSZ RSS TTY STAT START TIME COMMAND
root 1 0.3 0.0 4108 3592 pts/0 Ss 05:20 0:00 bash
root 10 0.0 0.0 5896 2784 pts/0 R+ 05:20 0:00 ps aux
root@d42802588594:/# cat /etc/issue
Ubuntu 20.04.3 LTS ~
root@d42802588594:/# exit
exit
[node1] (local) root@192.168.0.13 ~
$ 
```

Docker for Beginners - Linux

[https://training.play-with-docker.com/beginner-linux/#Task\\_0](https://training.play-with-docker.com/beginner-linux/#Task_0)

Play with Docker classroom

You can commit a container to make an image from it - but you should avoid that wherever possible. It's much better to use a repeatable [Dockerfile](#) to build your image. You'll see that shortly.

### Run a background MySQL container

Background containers are how you'll run most applications. Here's a simple example using MySQL.

1. Run a new MySQL container with the following command.

```
docker container run \
--detach \
--name mydb \
-e MYSQL_ROOT_PASSWORD=my-secret-pw \
mysql:latest
```

- o `--detach` will run the container in the background.
- o `--name` will name it **mydb**.
- o `-e` will use an environment variable to specify the root password (NOTE: This should never be done in production).

As the MySQL image was not available locally, Docker automatically pulled it from Docker Hub.

```
Unable to find image 'mysql:latest' locally
latest: Pulling from library/mysql
aa18ad1a0d33: Pull complete
fd88d82d0ca2: Pull complete
```

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

Alpine Linux 3.12

```
root@d42802588594:/# exit
exit
[node1] (local) root@192.168.0.13 ~
$ cat /etc/issue
Welcome to Alpine Linux 3.12
Kernel r3 on an lm (l1)

[node1] (local) root@192.168.0.13 ~
$ docker container run \
> --detach \
> --name mydb \
> -e MYSQL_ROOT_PASSWORD=my-secret-pw \
> mysql:latest
Unable to find image 'mysql:latest' locally
latest: Pulling from library/mysql
72a69066d2fe: Pull complete
93619dbc5b36: Pull complete
99da31dd6142: Pull complete
626033c43d70: Pull complete
37d5d7efb64e: Pull complete
ac563150bd721: Pull complete
```

10:51 08-01-2022

```
75b6cce7b50d3: Pull complete  
ed1d0a3a64e4: Pull complete  
8eb36a82c85b: Pull complete  
41be6f1a1c40: Pull complete  
0e1b414eac71: Pull complete  
914c28654a91: Pull complete  
587693eb988c1: Pull complete  
b183c3585729: Pull complete  
315e21657a44: Pull complete  
Digest: sha256:0dc3dabc751ef46a6647234abdec2d47400f0dfbe77ab49b0b2bffdæ57846d  
Status: Downloaded newer image for mysql:latest  
41d6157c9f7d1529a6c922acb8167ca66f167119df0fe3d86964db6c0d7ba4e0
```

As long as the MySQL process is running, Docker will keep the container running in the background.

2. List the running containers.

```
docker container ls
```

Notice your container is running.

CONTAINER ID	IMAGE	COMMAND	CREATED
STATUS	PORTS	NAMES	
3f4e8da0caf7	mysql:latest	"docker-entrypoint..."	52 seconds ago
Up 51 seconds	3306/tcp	mydb	

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

f t

10:51 08-01-2022

```
2. List the running containers.  
docker container ls
```

Notice your container is running.

CONTAINER ID	IMAGE	COMMAND	CREATED
STATUS	PORTS	NAMES	
3f4e8da0caf7	mysql:latest	"docker-entrypoint..."	52 seconds ago
Up 51 seconds	3306/tcp	mydb	

3. You can check what's happening in your containers by using a couple of built-in Docker commands:  
`docker container logs` and `docker container top`.

```
docker container logs mydb
```

This shows the logs from the MySQL Docker container.

```
<output truncated>  
2017-09-29T16:02:58.605004Z 0 [Note] Executing 'SELECT * FROM INFORMATION_SCHEMA.TABLES' to get a list of tables using the deprecated partition engine. You may use the startup option '--disable-partition-engine-check' to skip this check.  
2017-09-29T16:02:58.605026Z 0 [Note] Beginning of list of non-natively partitioned tables  
2017-09-29T16:02:58.616575Z 0 [Note] End of list of non-natively partitioned table
```

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

f t

10:51 08-01-2022

Docker for Beginners - Linux

training.play-with-docker.com/beginner-linux/#Task\_0

Play with Docker classroom

This shows the logs from the MySQL Docker container.

```
<output truncated>
2017-09-29T16:02:58.605004Z 0 [Note] Executing 'SELECT * FROM INFORMATION_SCHEMA.TABLES;' to get a list of tables using the deprecated partition engine. You may use the startup option '--disable-partition-engine-check' to skip this check.
2017-09-29T16:02:58.605026Z 0 [Note] Beginning of list of non-natively partitioned tables
2017-09-29T16:02:58.616575Z 0 [Note] End of list of non-natively partitioned tables
```

Let's look at the processes running inside the container.

```
docker container top mydb
```

You should see the MySQL daemon (`mysqld`) is running in the container.

PID	USER	TIME	COMMAND
2876	999	0:00	mysqld

Although MySQL is running, it is isolated within the container because no network ports have been published to the host. Network traffic cannot reach containers from the host unless ports are explicitly published.

4. List the MySQL version using `docker container exec`.

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

If the commandline doesn't appear in the terminal, make sure popups are enabled or try resizing the browser window.

```
2022-01-08 05:21:18+00:00 [Note] [Entrypoint]: Temporary server started.
Warning: Unable to load '/usr/share/zoneinfo/iso3166.tab' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/leap-seconds.list' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/zone.tab' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/zone1970.tab' as time zone. Skipping it.
```

```
2022-01-08 05:21:21+00:00 [Note] [Entrypoint]: Stopping temporary server
2022-01-08T05:21:21.594986Z 10 [System] [MY-013172] [Server] Received SHUTDOWN from user root. Shutting down mysqld (Version: 8.0.27).
2022-01-08T05:21:22.916119Z 0 [System] [MY-010910] [Server] /usr/sbin/mysqld: Shutdown complete (mysqld 8.0.27) MySQL Community Server - GPL.
[node1] (local) root@192.168.0.13 ~
$ docker container top mydb
PID USER TIME COMMAND
1133 999 0:00 mysqld
[node1] (local) root@192.168.0.13 ~
$
```

Docker for Beginners - Linux

training.play-with-docker.com/beginner-linux/#Task\_0

Play with Docker classroom

`docker container exec` to run the command-line equivalent of `mysql --user=root --password=$MYSQL_ROOT_PASSWORD --version` inside our MySQL container.

```
docker exec -it mydb \
mysql --user=root --password=$MYSQL_ROOT_PASSWORD --version
```

You will see the MySQL version number, as well as a handy warning.

```
mysql: [Warning] Using a password on the command line interface can be insecure.
mysql Ver 14.14 Distrib 5.7.19, for Linux (x86_64) using Editline wrapper
```

5. You can also use `docker container exec` to connect to a new shell process inside an already-running container. Executing the command below will give you an interactive shell (`sh`) inside your MySQL container.

```
docker exec -it mydb sh
```

Notice that your shell prompt has changed. This is because your shell is now connected to the `sh` process running inside of your container.

6. Let's check the version number by running the same command again, only this time from within the new shell session in the container.

```
mysql --user=root --password=$MYSQL_ROOT_PASSWORD --version
```

Notice the output is the same as before.

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

If the commandline doesn't appear in the terminal, make sure popups are enabled or try resizing the browser window.

```
2022-01-08T05:21:21.594986Z 10 [System] [MY-013172] [Server] Received SHUTDOWN from user root. Shutting down mysqld (Version: 8.0.27).
2022-01-08T05:21:22.916119Z 0 [System] [MY-010910] [Server] /usr/sbin/mysqld: Shutdown complete (mysqld 8.0.27) MySQL Community Server - GPL.
[node1] (local) root@192.168.0.13 ~
$ docker container top mydb
PID USER TIME COMMAND
1133 999 0:00 mysqld
[node1] (local) root@192.168.0.13 ~
$ docker exec -it mydb \
> mysql --user=root --password=$MYSQL_ROOT_PASSWORD --version
mysql: [Warning] Using a password on the command line interface can be insecure.
mysql Ver 8.0.27 for Linux on x86_64 (MySQL Community Server - GPL)
[node1] (local) root@192.168.0.13 ~
$ docker exec -it mydb sh
# mysql --user=root --password=$MYSQL_ROOT_PASSWORD --version
mysql: [Warning] Using a password on the command line interface can be insecure.
mysql Ver 8.0.27 for Linux on x86_64 (MySQL Community Server - GPL)
```

Docker for Beginners - Linux

training.play-with-docker.com/beginner-linux/#Task\_0

Play with Docker classroom

1. Make sure you're in the `linux_tweet_app` directory.

```
cd ~/linux_tweet_app
```

2. Display the contents of the Dockerfile.

```
cat Dockerfile
```

```
FROM nginx:latest
COPY index.html /usr/share/nginx/html
COPY linux.png /usr/share/nginx/html
EXPOSE 80 443
CMD ["nginx", "-g", "daemon off;"]
```

Let's see what each of these lines in the Dockerfile do.

- `FROM` specifies the base image to use as the starting point for this new image you're creating. For this example we're starting from `nginx:latest`.
- `COPY` copies files from the Docker host into the image, at a known location. In this example, `COPY` is used to copy two files into the image: `index.html` and a graphic that will be used on our webpage.
- `EXPOSE` documents which ports the application uses.

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

f t

10:52 08-01-2022 ENG IN

Docker for Beginners - Linux

training.play-with-docker.com/beginner-linux/#Task\_0

Play with Docker classroom

5. Use the `docker image build` command to create a new Docker image using the instructions in the Dockerfile.

- `--tag` allows us to give the image a custom name. In this case it's comprised of our DockerID, the application name, and a version. Having the Docker ID attached to the name will allow us to store it on Docker Hub in a later step
- `.` tells Docker to use the current directory as the build context

Be sure to include period `(.)` at the end of the command.

```
docker image build --tag $DOCKERID/linux_tweet_app:1.0 .
```

The output below shows the Docker daemon executing each line in the Dockerfile

```
Sending build context to Docker daemon 32.77kB
Step 1/5 : FROM nginx:latest
latest: Pulling from library/nginx
afeb2bfd31c0: Pull complete
7ff5d10493db: Pull complete
d2562f1ae1d0: Pull complete
Digest: sha256:af32e714a9c3157157374e68c818b05ebe9e0737aac06b55a09da374209a8f9
Status: Downloaded newer image for nginx:latest
--> da5939581ac8
Step 2/5 : COPY index.html /usr/share/nginx/html
--> 605c77e624dd
```

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

f t

10:52 08-01-2022 ENG IN

Docker for Beginners - Linux | ip172-18-0-5-c7chu2tmrpg00a | +

training.play-with-docker.com/beginner-linux/#Task\_0

Play with Docker classroom

image, and run a new version every time you make a change to your source code.

One way to streamline this process is to mount the source code directory on the local machine into the running container. This will allow any changes made to the files on the host to be immediately reflected in the container.

We do this using something called a [bind mount](#).

When you use a bind mount, a file or directory on the host machine is mounted into a container running on the same host.

**Start our web app with a bind mount**

1. Let's start the web app and mount the current directory into the container.

In this example we'll use the `--mount` flag to mount the current directory on the host into `/usr/share/nginx/html` inside the container.

Be sure to run this command from within the `linux_tweet_app` directory on your Docker host.

```
docker container run \
--detach \
--publish 80:80 \
--name linux_tweet_app \
--mount type=bind,source="$(pwd)",target=/usr/share/nginx/html \
$DOCKERID/linux_tweet_app:1.0
```

Remember from the Dockerfile, `usr/share/nginx/html` is where the html

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

If the commandline doesn't appear in the terminal, make sure popups are enabled or try resizing the browser window.

```
Successfully tagged 9656562186/linux_tweet_app:1.0
[node1] (local) root@192.168.0.13 ~/linux_tweet_app
$ docker container run \
> --detach \
> --publish 80:80 \
> --name linux_tweet_app \
> $DOCKERID/linux_tweet_app:1.0
2ab2bfc30e2930cce7d9faf398fc1c2ca3fd4c135e6191d4ddb2d9d4d4036f2
[node1] (local) root@192.168.0.13 ~/linux_tweet_app
$ docker container rm --force linux_tweet_app
linux_tweet_app
[node1] (local) root@192.168.0.13 ~/linux_tweet_app
$ docker container run \
> --detach \
> --publish 80:80 \
> --name linux_tweet_app \
> --mount type=bind,source="$(pwd)",target=/usr/share/nginx/html \
> $DOCKERID/linux_tweet_app:1.0
bb673b4a4dceec9e9e0bb15c867617c3d2c52e92ef62f01d7d1b3c59cf38bf14
[node1] (local) root@192.168.0.13 ~/linux_tweet_app
$
```

Docker for Beginners - Linux | ip172-18-0-5-c7chu2tmrpg00a | +

training.play-with-docker.com/beginner-linux/#Task\_0

Play with Docker classroom

reload the webpage. If you are really adventurous, why not try using `exec` to access the running container and modify the files stored there.

Even though we've modified the `index.html` local filesystem and seen it reflected in the running container, we've not actually changed the Docker image that the container was started from.

To show this, stop the current container and re-run the `1.0` image without a bind mount.

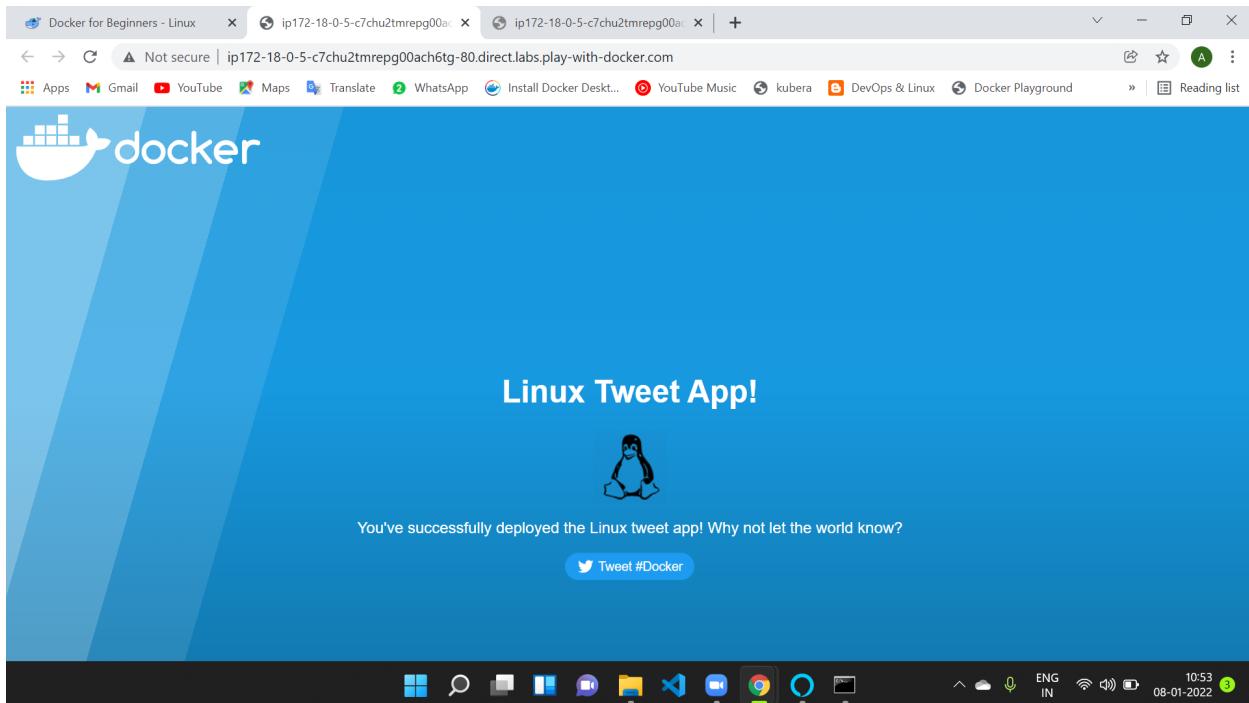
1. Stop and remove the currently running container.  
docker rm --force linux\_tweet\_app
2. Rerun the current version without a bind mount.  
docker container run \
--detach \
--publish 80:80 \
--name linux\_tweet\_app \
\$DOCKERID/linux\_tweet\_app:1.0
3. Notice the [website](#) is back to the original version.
4. Stop and remove the current container  
docker rm --force linux\_tweet\_app

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

If the commandline doesn't appear in the terminal, make sure popups are enabled or try resizing the browser window.

```
$ docker container run \
> --detach \
> --publish 80:80 \
> --name linux_tweet_app \
> --mount type=bind,source="$(pwd)",target=/usr/share/nginx/html \
> $DOCKERID/linux_tweet_app:1.0
bb673b4a4dceec9e9e0bb15c867617c3d2c52e92ef62f01d7d1b3c59cf38bf14
[node1] (local) root@192.168.0.13 ~/linux_tweet_app
$ cp index-new.html index.html
[node1] (local) root@192.168.0.13 ~/linux_tweet_app
$ docker rm --force linux_tweet_app
linux_tweet_app
[node1] (local) root@192.168.0.13 ~/linux_tweet_app
$ docker container run \
> --detach \
> --publish 80:80 \
> --name linux_tweet_app \
> $DOCKERID/linux_tweet_app:1.0
372d5ff4c986c9cb71a1282dfc4672288f882a631f76f769edae3adecb74db25
[node1] (local) root@192.168.0.13 ~/linux_tweet_app
$
```



Docker for Beginners - Linux | training.play-with-docker.com/beginner-linux/#Task\_0 | ip172-18-0-5-c7chu2tmrpg00a | ip172-18-0-5-c7chu2tmrpg00a | +

Play with Docker classroom

```
docker image build --tag $DOCKERID/linux_tweet_app:2.0 .
```

Notice how fast that built! This is because Docker only modified the portion of the image that changed vs. rebuilding the whole image.

2. Let's look at the images on the system.

```
docker image ls
```

You now have both versions of the web app on your host.

REPOSITORY	TAG	IMAGE ID	CREATED
<docker id>/linux_tweet_app	2.0	01612e05312b	16 seconds ago
go	108MB		
<docker id>/linux_tweet_app	1.0	bb32b5783cd3	4 minutes ago
o	108MB		
mysql	latest	b4e78b89bcf3	2 weeks ago
412MB			
ubuntu	latest	2d696327ab2e	2 weeks ago
122MB			
nginx	latest	d45939581ac8	3 weeks ago
108MB			
alpine	latest	76da55c8019d	3 weeks ago
3.97MB			

If the commandline doesn't appear in the terminal, make sure popups are enabled or try resizing the browser window.

```
Removing intermediate container 8057880e66c
--> 15b1769d19ad
Successfully built 15b1769d19ad
Successfully tagged 9656562186/linux_tweet_app:2.0
[node1] (local) root@192.168.0.13 ~/linux_tweet_app
$ docker image ls
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
9656562186/linux_tweet_app    2.0      15b1769d19ad  2 seconds ago  142M
B
9656562186/linux_tweet_app    1.0      cd976c0d34b9  59 seconds ago  142M
B
ubuntu              latest   d13c942271d6  27 hours ago  72.8M
MB
nginx               latest   605c77e624dd  9 days ago   141M
B
mysql               latest   3218b38490ce  2 weeks ago   516M
B
alpine              latest   c059bfaa849c  6 weeks ago   5.59M
MB
[node1] (local) root@192.168.0.13 ~/linux_tweet_app
$
```

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

f t 📺

10:53 08-01-2022

Docker for Beginners - Linux | training.play-with-docker.com/beginner-linux/#Task\_0 | ip172-18-0-5-c7chu2tmrpg00a | ip172-18-0-5-c7chu2tmrpg00a | +

Play with Docker classroom

REPOSITORY	TAG	IMAGE ID	CREATED
412MB			
ubuntu	latest	2d696327ab2e	2 weeks ago
122MB			
nginx	latest	d45939581ac8	3 weeks ago
108MB			
alpine	latest	76da55c8019d	3 weeks ago
3.97MB			

If the commandline doesn't appear in the terminal, make sure popups are enabled or try resizing the browser window.

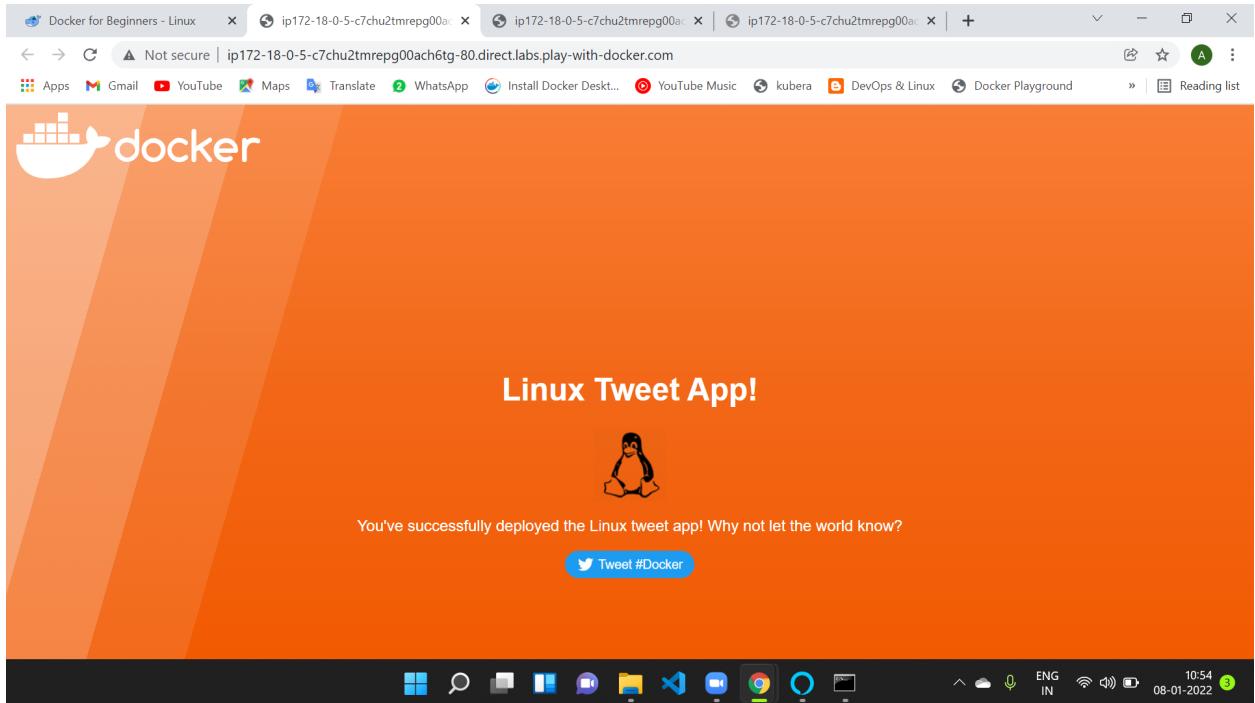
```
9656562186/linux_tweet_app    2.0      15b1769d19ad  2 seconds ago  142M
B
9656562186/linux_tweet_app    1.0      cd976c0d34b9  59 seconds ago  142M
B
ubuntu              latest   d13c942271d6  27 hours ago  72.8M
MB
nginx               latest   605c77e624dd  9 days ago   141M
B
mysql               latest   3218b38490ce  2 weeks ago   516M
B
alpine              latest   c059bfaa849c  6 weeks ago   5.59M
MB
[node1] (local) root@192.168.0.13 ~/linux_tweet_app
$ docker container run \
> --detach \
> --publish 80:80 \
> --name linux_tweet_app \
> $DOCKERID/linux_tweet_app:2.0
99a344b3dc0c1ff179c123352ed21d76aa7b518eb06ebda467555eae2fb9edd6
[node1] (local) root@192.168.0.13 ~/linux_tweet_app
$
```

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

f t 📺

10:53 08-01-2022



Docker for Beginners - Linux | ip172-18-0-5-c7chu2tmrpg00ach6tg-80.direct.labs.play-with-docker.com | ip172-18-0-5-c7chu2tmrpg00ach6tg-80.direct.labs.play-with-docker.com | ip172-18-0-5-c7chu2tmrpg00ach6tg-80.direct.labs.play-with-docker.com | +

training.play-with-docker.com/beginner-linux/#Task\_0

Apps Gmail YouTube Maps Translate WhatsApp Install Docker Deskt... YouTube Music kubera DevOps & Linux Docker Playground

Play with Docker classroom

after the workshop. In this step we'll push the images to a public repository so you can run them from any Linux machine with Docker.

Distribution is built into the Docker platform. You can build images locally and push them to a public or private [registry](#), making them available to other users. Anyone with access can pull that image and run a container from it. The behavior of the app in the container will be the same for everyone, because the image contains the fully-configured app - the only requirements to run it are Linux and Docker.

Docker Hub is the default public registry for Docker images.

2. Before you can push your images, you will need to log into Docker Hub.

docker login

You will need to supply your Docker ID credentials when prompted.

Username: <your docker id>  
Password: <your docker id password>  
Login Succeeded

3. Push version 1.0 of your web app using `docker image push`.

docker image push \$DOCKERID/linux\_tweet\_app:1.0

You'll see the progress as the image is pushed up to Docker Hub.

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

f t

10:54 08-01-2022

Docker for Beginners - Linux | training.play-with-docker.com/beginner-linux/#Task\_0 | Docker Playground

You'll see the progress as the image is pushed up to Docker Hub.

```
The push refers to a repository [docker.io/<your docker id>/linux_tweet_app]
910e84bcf7a1: Pushed
1dee161c8b4a: Pushed
110566462efaf1: Pushed
305e2b66ef45a1: Pushed
24e065a5f3281: Pushed
1.0: digest: sha256:51e937ec18c7757879722f15fa1044cbfb2f6b7eaeb578c7c352babaa6d
c size: 1363
```

4. Now push version 2.0.

```
docker image push $DOCKERID/linux_tweet_app:2.0
```

Notice that several lines of the output say `Layer already exists`. This is because Docker will leverage read-only layers that are the same as any previously uploaded image layers.

```
The push refers to a repository [docker.io/<your docker id>/linux_tweet_app]
0b171f8f76be221: Pushed
70d38c767c001: Pushed
110566462efaf1: Layer already exists
305e2b66ef45a1: Layer already exists
24e065a5f3281: Layer already exists
2.0: digest: sha256:7c51f77f90b81e5a598a13f129c95543172bae8f5850537225eae0c78e4f3ad
d size: 1363
```

Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

f t

Docker for Beginners - Linux | training.play-with-docker.com/beginner-linux/#Task\_0 | Docker Playground

Play with Docker classroom

If the commandline doesn't appear in the terminal, make sure popups are enabled or try resizing the browser window.

```
696c9432b357: Pushed
4825d40a1clee: Pushed
d874fd2bc83b: Mounted from library/nginx
32ce5f6a5106: Mounted from library/nginx
f1db227348d0: Mounted from library/nginx
b8d6e692a25e: Mounted from library/nginx
e379e8aedd4d: Mounted from library/nginx
2edcec3590a4: Layer already exists
1.0: digest: sha256:3le76139bd22a5dadffbbd937168e75facdf5924eee34995567336d
cd4c79f4b size: 1985
[node1] (local) root@192.168.0.13 ~/linux_tweet_app
$ 
```

If the commandline doesn't appear in the terminal, make sure popups are enabled or try resizing the browser window.

```
696c9432b357: Pushed
4825d40a1clee: Pushed
d874fd2bc83b: Mounted from library/nginx
32ce5f6a5106: Mounted from library/nginx
f1db227348d0: Mounted from library/nginx
b8d6e692a25e: Mounted from library/nginx
e379e8aedd4d: Mounted from library/nginx
2edcec3590a4: Layer already exists
1.0: digest: sha256:3le76139bd22a5dadffbbd937168e75facdf5924eee34995567336d
cd4c79f4b size: 1985
[node1] (local) root@192.168.0.13 ~/linux_tweet_app
$ 
```

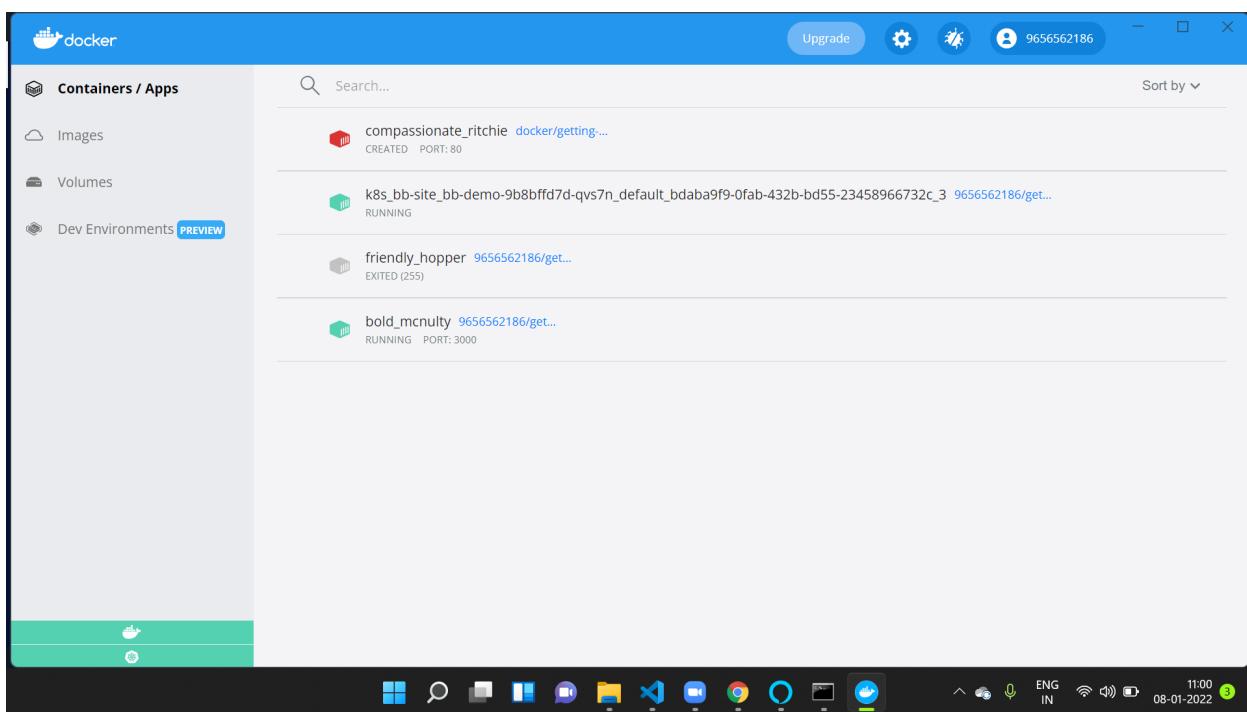
Your use of Play With Docker is subject to the Docker Terms of Service which can be accessed [here](#). Site created by Tutorius

docker build automated

f t

A screenshot of Visual Studio Code showing a terminal window running a Docker build command. The terminal output shows the process of building a Docker image from a Dockerfile in the 'app' directory. The build process includes transferring the Dockerfile, loading metadata, and running commands like apk add and yarn install. The terminal window is titled '.gitignore - getting-started-master - Visual Studio Code'.

```
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master> cd app
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker image build -t 9656562186/getting-started-master .
[+] Building 3.8s (10/10) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 32B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/node:12-alpine
=> [1/5] FROM docker.io/library/node:12-alpine
=> [internal] load build context
=> => transferring context: 597.80kB
=> CACHED [2/5] RUN apk add --no-cache python2 g++ make
=> CACHED [3/5] WORKDIR /app
=> CACHED [4/5] COPY .
=> CACHED [5/5] RUN yarn install --production
=> exporting to image
=> exporting layers
=> => writing image sha256:9f6dd5db7a854f2e9b8b869c422108670098e728a6772780f0861858d3c53f04
=> => naming to docker.io/9656562186/getting-started-master
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker run -dp 3000:3000 9656562186/getting-started-master
14dbc69b7dbbf811b85643f91bc5ea0f24bcd5a0079fd9ecbbf627f408c6e83e
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app>
```



File Edit Selection View Go Run Terminal Help .gitignore - getting-started-master - Visual Studio Code

EXPLORER PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

TERMINAL

```
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker image build -t 9656562186/getting-started-master .
[+] Building 3.8s (10/10) FINISHED
=> [internal] load build definition from Dockerfile
=> transferring dockerfile: 32B
=> [internal] load .dockerignore
=> transferring context: 2B
=> [internal] load metadata for docker.io/library/node:12-alpine
=> [1/5] FROM docker.io/library/node:12-alpine
=> [internal] load build context
=> transferring context: 597.80kB
=> CACHED [2/5] RUN apk add --no-cache python2 g++ make
=> CACHED [3/5] WORKDIR /app
=> CACHED [4/5] COPY . .
=> CACHED [5/5] RUN yarn install --production
=> exporting to image
=> exporting layers
=> writing image sha256:9f6dd5db7a854f2e9b8b869c422108670098e728a6772780f0861858d3c53f04
=> naming to docker.io/9656562186/getting-started-master
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker run -dp 3000:3000 9656562186/getting-started-master
14dbc69b7dbbf811b85643f91bcSeaf0f24bcd5a0079fd9ecbf627f408c6e83e
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker push 9656562186/getting-started-master
Using default tag: latest
The push refers to repository [docker.io/9656562186/getting-started-master]
4012ac85cf17: Pushed
2f78579c7454: Pushed
7f218d44002d: Mounted from 9656562186/newwork123
def1f3ead64: Mounted from 9656562186/newwork123
c99d9bb91c1e: Mounted from 9656562186/new
743fabfeeb083: Mounted from 9656562186/newwork123
529fc2dc4666c: Mounted from 9656562186/new
1a058d5342cc: Mounted from 9656562186/new
latest: digest: sha256:32a4fccce991dd15eff53f62b659f7e556e22e3626582a0812411978702c382f size: 2001
PS C:\Users\amalu\Downloads\getting-started-master\app>
```

Ln 2, Col 1 Spaces: 4 UTF-8 LF Ignore ⌂

master\* ⌂ 0 ▲ 0

11:07 08-01-2022

Kubera Email - Amalu Sant... Docker - Container knowledge mission Untitled document Docker Hub

hub.docker.com/repository/docker/9656562186/getting-started-master

Advanced Image Management

View all your images and tags in this repository, clean up unused content, recover untagged images. Available with Pro, Team and Business subscriptions.

**9656562186 / getting-started-master**

This repository does not have a description

Last pushed: a few seconds ago

Docker commands

To push a new tag to this repository,

`docker push 9656562186/getting-started-master:tagname`

**Tags and Scans**

VULNERABILITY SCANNING - DISABLED

Enable

This repository contains 1 tag(s).

TAG	OS	PULLED	PUSHED
latest	🐧	a few seconds ago	a few seconds ago

[See all](#)

**Automated Builds**

Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.

Available with Pro, Team and Business subscriptions.

[Upgrade to Pro](#) [Learn more](#)

11:08 08-01-2022

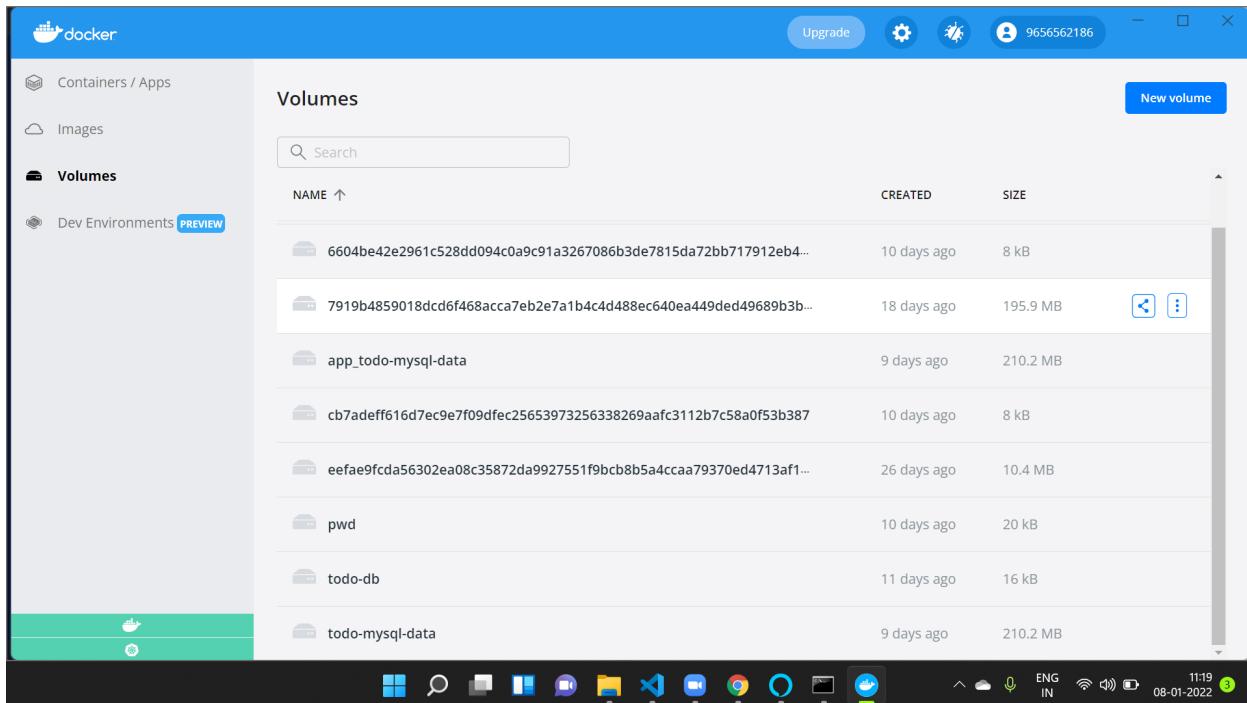
The screenshot shows a web browser window with the URL `localhost:3000`. The page displays a list titled "New Item" with two items: "one" and "two". Each item has a red trash can icon to its right. At the top right of the list area is a green "Add Item" button.

The browser's address bar shows multiple tabs, including "Kubera", "Email - Amalu", "Docker - Cont...", "knowledge mi...", "Untitled docu...", "Docker Hub", "Todo App", and "Reading list". Below the browser is the Windows taskbar, which includes icons for File Explorer, Task View, Task Manager, and several pinned applications like Visual Studio Code, Google Chrome, and Microsoft Edge.

The screenshot shows a Visual Studio Code terminal window titled ".gitignore - getting-started-master - Visual Studio Code". The terminal output shows the following sequence of Docker commands:

```
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker run -dp 3000:3000 9656562186/getting-started-master
14dbc69b7dbbf811b85643f91b5ea0f24bcd5a0879fd9ecbbf627f408c6e83e
PS C:\Users\amalu\Downloads\getting-started-master\app> docker push 9656562186/getting-started-master
Using default tag: latest
The push refers to repository [docker.io/9656562186/getting-started-master]
4012ac85cf17: Pushed
2f70579c7454: Pushed
7f218d44002d: Mounted from 9656562186/newwork123
de1f13eaad64: Mounted from 9656562186/newwork123
c99dbbb91c1c: Mounted from 9656562186/new
743fabfe0b03: Mounted from 9656562186/newwork123
529f2cdaf66c: Mounted from 9656562186/new
1a058d5342cc: Mounted from 9656562186/new
latest: digest: sha256:32a4fcce991dd15eff53f62b659f7e556e22e3626582a0812411978702c382f size: 2001
PS C:\Users\amalu\Downloads\getting-started-master\app> docker volume inspect todo-db
[{"Name": "todo-db", "Driver": "local", "Scope": "local", "Mountpoint": "/var/lib/docker/volumes/todo-db/_data", "Labels": {}, "Options": {}}]
```

The terminal also shows the status bar at the bottom indicating "Ln 2, Col 1", "Spaces: 4", "UTF-8", "LF", "Ignore", and the date/time "08-01-2022 11:14".



The screenshot shows a Visual Studio Code window with a terminal tab open. The terminal output shows a series of Docker commands being run in a Windows environment (PS C:\). The commands include creating a network, running a container, and connecting to a MySQL database.

```
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker exec -it c20987f18b13 mysql -u root -p
Error: No such container: c20987f18b13
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker network create todo-app
docker: 'newtwork' is not a docker command.
See 'docker --help'.
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker network create todo-app
Error response from daemon: network with name todo-app already exists
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker run -d \
--network todo-app --network-alias mysql \
>> -v todo-mysql-data:/var/lib/mysql \
>> -e MYSQL_ROOT_PASSWORD=123 \
>>
"docker run" requires at least 1 argument.
See 'docker run --help'.

Usage: docker run [OPTIONS] IMAGE [COMMAND] [ARG...]

Run a command in a new container
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker run -d \
>> --network todo-app --network-alias mysql \
>> -v todo-mysql-data:/var/lib/mysql \
>> -e MYSQL_ROOT_PASSWORD=123 \
>> -e MYSQL_DATABASE=todos \
>> mysql:5.7
15daf176da32efd86d47413865b61512bb6895e1dbb00992f8d30e277535634b
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app>
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app>
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app>
```

File Edit Selection View Go Run Terminal Help .gitignore - getting-started-master - Visual Studio Code

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS C:\Users\amal\Downloads\getting-started-master\getting-started-master\app> docker ps

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
15daf1768a32	mysql:5.7	"docker-entrypoint.s..."	About a minute ago	Up About a minute	3306/tcp, 33060/tcp	upbeat_hofstadter
0879433c22dc	node:12-alpine	"docker-entrypoint.s..."	6 minutes ago	Up 6 minutes	0.0.0.0:3005->3000/tcp	relaxed_euclid
14dbc69b7dbb	9656562186/getting-started-master	"docker-entrypoint.s..."	26 minutes ago	Up 26 minutes	0.0.0.0:3000->3000/tcp	bold_mcnulty
2b808ff7d33	9656562186/getting-started	"docker-entrypoint.s..."	2 hours ago	Up 2 hours		k8s_bb-site_bb-demo

PS C:\Users\amal\Downloads\getting-started-master\getting-started-master\app> docker exec -it 15daf1768a32 mysql -u root -p

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 2

Server version: 5.7.36 MySQL Community Server (GPL)

Copyright (c) 2000, 2021, 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>

Ln 2, Col 1 Spaces: 4 UTF-8 LF Ignore ⚙️ ⚙️

11:27 08-01-2022

File Edit Selection View Go Run Terminal Help .gitignore - getting-started-master - Visual Studio Code

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

pp> docker exec -it 15daf1768a32 mysql -u root -p

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 2

Server version: 5.7.36 MySQL Community Server (GPL)

Copyright (c) 2000, 2021, 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> show database;

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'database' at line 1

mysql> SHOW DATABASE;

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'DATABASE' at line 1

mysql> show databases;

Database
information_schema
mysql
performance_schema
sys
todos

5 rows in set (0.01 sec)

mysql>

Ln 2, Col 1 Spaces: 4 UTF-8 LF Ignore ⚙️ ⚙️

11:27 08-01-2022

A screenshot of Visual Studio Code showing a terminal session. The terminal tab is active, displaying a series of Docker commands and MySQL logs. The logs show a MySQL container being started, entering a password, and then listing databases.

```
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS
15daf1768a32      mysql:5.7          "docker-entrypoint.s..."   About a minute ago   Up About a minute   3306/tcp, 33060/tcp   upbeat_hofstadter
f1768a32          mysql:5.7           "mysql -u root -p"   6 minutes ago     Up 6 minutes      0.0.0.0:3000->3000/tcp   relax

pp> docker exec -it 15daf1768a32 mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2
f1768a32 mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.7.36 MySQL Community Server (GPL)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'databas
e' at line 1
mysql> SHOW DATABASE;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'DATABAS
E' at line 1
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
| todos |
+-----+
5 rows in set (0.01 sec)

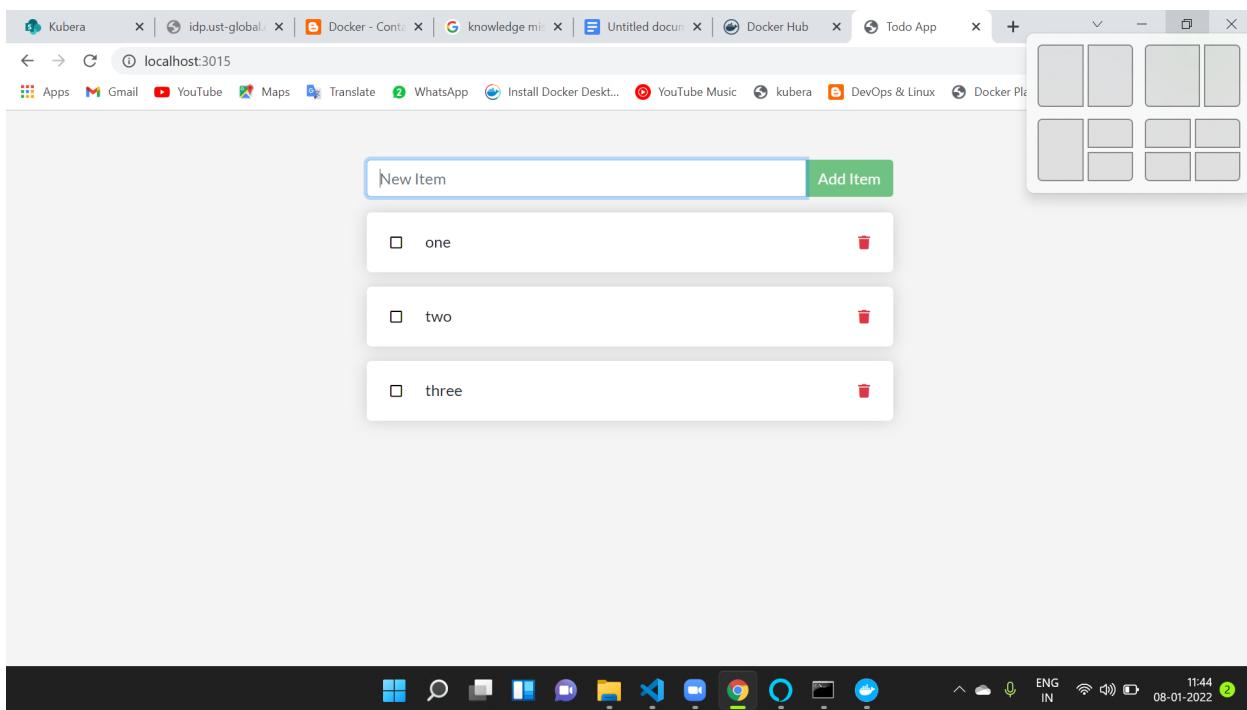
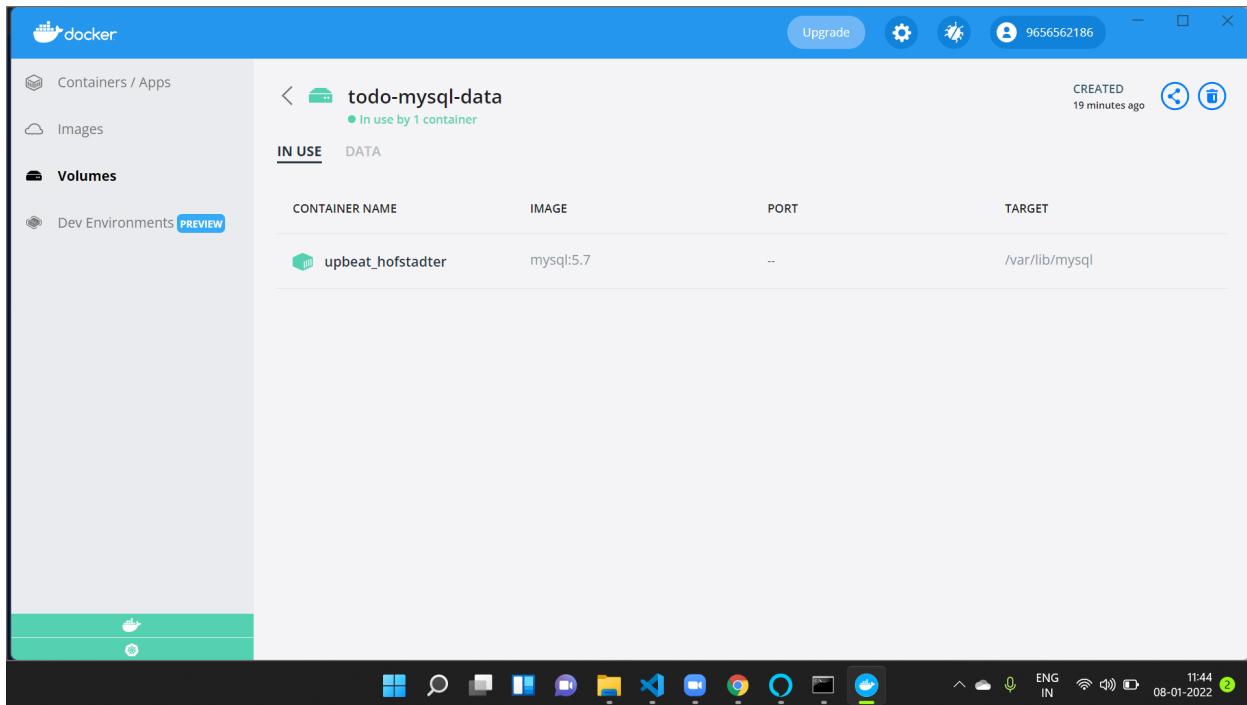
mysql> docker run -it --network=host nicolaka/netshoot
-> dig mysql
```

A screenshot of the Docker desktop application. The left sidebar shows 'Containers / Apps', 'Images', 'Volumes' (which is selected), and 'Dev Environments'. The main area is titled 'Volumes' and lists several volumes with their names, creation times, and sizes. One volume, 'todo-mysql-data', is marked as 'IN USE'.

NAME	CREATED	SIZE
6604be42e2961c528dd094c0a9c91a3267086b3de7815da72bb717912...	10 days ago	8 kB
7919b4859018dcdf6f468acca7eb2e7a1b4c4d488ec640ea449ded49689...	18 days ago	195.9 MB
app_todo-mysql-data	9 days ago	210.2 MB
cb7adef616d7ec9e7f09dfec25653973256338269aafc3112b7c58a0f53...	10 days ago	8 kB
eefae9fcda56302ea08c35872da9927551f9bcb8b5a4ccaa79370ed471...	26 days ago	10.4 MB
pwd	10 days ago	20 kB
todo-db	11 days ago	16 kB
todo-mysql-data	9 days ago	210.2 MB

"docker run" requires at least 1 argument.  
See 'docker run --help'.  
Usage: docker run [OPTIONS] IMAGE [COMMAND] [ARG...]  
Run a command in a new container  
PS C:\Users\amal\Downloads\getting-started-master\getting-started-master\app> docker run -dp 3005:3000 <br/>>> -w /app -v "\$(\pwd)":/app<br/>>> --network todo-app<br/>>> -e MYSQL\_HOST=mysql<br/>>> -e MYSQL\_USER=root<br/>>> -e MYSQL\_PASSWORD=1234<br/>>> -e MYSQL\_DB=todos<br/>>> node:12-alpine<br/>>> sh "yarn install && yarn run dev" <br/>3017675487d8bc66e23281c9633cbc02093d459c9ee5b206d9c4d68014dc951<br/>docker: Error response from daemon: driver failed programming external connectivity on endpoint vigorous\_visvesvaraya (4f5f1b3e0087e5b80a40962a64bf56ad43496e430c4 d4ab2d85558e561d7e0805): Bind for 0.0.0.0:3005 failed: port is already allocated.<br/>PS C:\Users\amal\Downloads\getting-started-master\getting-started-master\app> docker run -dp 3015:3000 <br/>>> -w /app -v "\$(\pwd)":/app<br/>>> --network todo-app<br/>>> -e MYSQL\_HOST=mysql<br/>>> -e MYSQL\_USER=root<br/>>> -e MYSQL\_PASSWORD=1234<br/>>> -e MYSQL\_DB=todos<br/>>> node:12-alpine<br/>>> sh "yarn install && yarn run dev" <br/>897200459e973d44dfe804ae42ecdabc73d9229d9163bfe4a5986c2eb65b639<br/>PS C:\Users\amal\Downloads\getting-started-master\getting-started-master\app>

22 minutes ago Up 22 minutes 0.0.0.0:3005->3000/tcp relaxed\_euclid  
14dbc69b7dbbb 9656562186/getting-started-master "docker-entrypoint.s..." 42 minutes ago Up 42 minutes 0.0.0.0:3000->3000/tcp bold\_mcnulty  
2b808ff3fd33 9656562186/getting-started "docker-entrypoint.s..." 2 hours ago Up 2 hours k8s\_bb-site\_bb-demo-9b8bffd7  
d-qvs7n\_default\_bdbab9f9-0fab-432b-bd55-23458966732c\_3  
PS C:\Users\amal\Downloads\getting-started-master\app> docker exec -it 15daf1768a32 mysql -p todos  
Enter password:  
Reading table information for completion of table and column names  
You can turn off this feature to get a quicker startup with -A  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 5  
Server version: 5.7.36 MySQL Community Server (GPL)  
Copyright (c) 2000, 2021, 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> select \* from todo\_items;  
+----+----+----+  
| id | name | completed |  
+----+----+----+  
f2908863-8d42-4ad5-8919-75fbff65ad483	one	0
a84fe8ad-2c61-4219-9f96-a0d1eaded7bc	two	0
0abc9818-2314-4973-a8a0-2cc6a4ae56cf	three	0
+----+----+----+  
3 rows in set (0.01 sec)  
mysql>



A screenshot of Visual Studio Code showing a terminal session. The terminal tab is active, displaying MySQL queries and Docker commands. The MySQL queries show data from a 'todo\_items' table. The Docker commands involve running a MySQL container and then a Node.js application ('app') container, mapping port 8000 to 8000.

```
mysql> select * from todo_items;
+----+-----+-----+
| id | name | completed |
+----+-----+-----+
| f2908863-8d42-4ad5-8919-75fbf65ad483 | one | 0 |
| a84fe8ad-2c61-4219-9f96-a0d1eaded7bc | two | 0 |
| 0abc9818-2314-4973-ab80-2cc6a4ae56cf | three | 0 |
+----+-----+-----+
3 rows in set (0.01 sec)

mysql>
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker-compose version
Docker Compose version v2.2.1
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker run -dp 3020:3000 \
>> -v /app -> "${pwd}:/app"
>> --network todo-app
>> -e MYSQL_HOST=mysql
>> -e MYSQL_USER=root
>> -e MYSQL_PASSWORD=1234
>> -e MYSQL_DB=todos
>> node:12-alpine
>> sh -c "yarn install && yarn run dev"
56928d153172f22408c12cdedfc26271e5529dc102a2eefb59ac22c8b21a2
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker run -d \
>> --network todo-app --network-alias mysql
>> -v todo-mysql-data:/var/lib/mysql
>> -e MYSQL_ROOT_PASSWORD=123
>> -e MYSQL_DATABASE=todos
>> mysql:5.7
d9abbd05bd8228b256099d4ff4bae95d369b1ded16a22aeeb4126d2669c19e7
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app>
```

A screenshot of Visual Studio Code showing the file explorer and terminal tabs. The file explorer shows a project structure for 'GETTING-STARTED-MASTER' containing files like Dockerfile, docker-compose.yml, app.js, and package.json. The terminal tab shows a command-line interface with Docker commands. A tooltip in the bottom right corner suggests installing the 'docker-compose.yml' extension for rich language support.

File Explorer:

- GETTING-STARTED-MASTER
  - .github\workflows
  - build.yml
  - app
    - node\_modules
    - spec
    - src
  - Dockerfile
  - package.json
  - yarn.lock
  - docs
  - .dockerrignore
  - .gitignore
  - build.sh
  - docker-compose.yml
  - Dockerfile
  - Jenkinsfile
  - LICENSE
  - mkdocs.yml
  - README.md
  - requirements.txt
  - yarn.lock

Terminal:

```
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app> docker run -d \
>> --network todo-app --network-alias mysql
>> -v todo-mysql-data:/var/lib/mysql
>> -e MYSQL_ROOT_PASSWORD=123
>> -e MYSQL_DATABASE=todos
>> mysql:5.7
d9abbd05bd8228b256099d4ff4bae95d369b1ded16a22aeeb4126d2669c19e7
PS C:\Users\amalu\Downloads\getting-started-master\getting-started-master\app>
```

Extension Recommendation:

Install Show Recommendations

```
> docker-compose up -d
[+] Building 11.8s (10/10) FINISHED
=> [internal] load build definition from Dockerfile          0.1s
=> transferring dockerfile: 1.12kB                         0.0s
=> [internal] load .dockerignore                           0.1s
=> transferring context: 52B                               0.0s
=> [internal] load metadata for docker.io/library/python:alpine    8.7s
=> [auth] library/python:pull token for registry-1.docker.io      0.0s
=> [base 1/4] FROM docker.io/library/python:alpin@sha256:dce56d40d 0.0s
=> [internal] load build context                          0.0s
=> transferring context: 148B                            0.0s
=> CACHED [base 2/4] WORKDIR /app                      0.0s
=> CACHED [base 3/4] COPY requirements.txt .           0.0s
=> CACHED [base 4/4] RUN pip install -r requirements.txt 0.0s
=> exporting to image                                  2.9s
=> => exporting layers                                2.8s
=> => writing image sha256:4ea73ca0c03caa10f88444d9cc2c92bb8779c227 0.0s
=> => naming to docker.io/library/getting-started-master_docs 0.0s

Use 'docker scan' to run Syk tests against images to find vulnerabilities and learn how to fix them
[+] Running 2/2
- Network getting-started-master_default   Created          0.1s
- Container getting-started-master_docs_1 Started        3.2s
PS C:\Users\amalv\Downloads\getting-started-master\getting-started-master\app>
```

Ln 7, Col 29 Spaces: 2 UTF-8 LF Compose ENG IN 11:49 08-01-2022

The screenshot shows the Visual Studio Code interface with the 'TERMINAL' tab selected. The terminal window displays the output of a 'docker-compose up' command for a 'getting-started-master' service. The output includes logs for Dockerfile transfers, context loading, metadata retrieval, authentication, build steps, layer caching, pip installations, and image creation. It also shows the creation of a 'docs\_1' container and its configuration, including a warning about the 'dev\_addr' setting. The terminal prompt is PS C:\Users\amal\Downloads\getting-started-master\getting-started-master\app>.

```
>> => transferring dockerfile: 1.12kB          0.0s
=> [internal] load .dockerignore               0.1s
=> => transferring context: 52B              0.0s
=> [internal] load metadata for docker.io/library/python:alpine      8.7s
=> [auth] library/python:pull token for registry-1.docker.io           0.0s
=> [base 1/4] FROM docker.io/library/python:alpine@sha256:dce56d40d  0.0s
=> [internal] load build context            0.0s
=> => transferring context: 148B             0.0s
=> CACHED [base 2/4] WORKDIR /app           0.0s
=> CACHED [base 3/4] COPY requirements.txt   0.0s
=> CACHED [base 4/4] RUN pip install -r requirements.txt    0.0s
=> exporting to image                      2.9s
=> => exporting layers                   2.8s
=> => writing image sha256:4ea73ca0c03caa10f88444d9cc2c92bb8779c227 0.0s
=> => naming to docker.io/library/getting-started-master_docs        0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
[+] Running 2/2
- Network getting-started-master_default  Created          0.1s
- Container getting-started-master_docs_1 Started       3.2s
PS C:\Users\amal\Downloads\getting-started-master\getting-started-master\app> docker-compose logs -f
docs_1 | INFO  - Building documentation...
docs_1 | WARNING - Config value: 'dev_addr'. Warning: The use of the IP address '0.0.0.0' suggests a production environment or the use of a proxy to connect to the MkDocs server. However, the MkDocs' server is intended for local development purposes only. Please use a third party production-ready server instead.
docs_1 | INFO  - Cleaning site directory
docs_1 | INFO  - The following pages exist in the docs directory, but are not included in the "nav" configuration:
docs_1 | - index.md
docs_1 | INFO  - Documentation built in 1.74 seconds
docs_1 | INFO  - [06:18:52] Serving on http://0.0.0.0:8000/

```

The screenshot shows the Docker desktop application interface. On the left, a sidebar lists 'Containers / Apps', 'Images', 'Volumes', and 'Dev Environments'. The 'Volumes' section is active, showing a table for 'todo-mysql-data'. The table has two rows under the 'IN USE' tab, both pointing to the 'mysql' container. The first row is for 'upbeat\_hofstader' and the second for 'cool\_poincare'. Both entries show the 'IMAGE' as 'mysql:5.7', 'PORT' as '--', and 'TARGET' as '/var/lib/mysql'. The volume was 'CREATED 2 minutes ago'. The Docker desktop tray icon is visible at the bottom right.

CONTAINER NAME	IMAGE	PORT	TARGET
upbeat_hofstader	mysql:5.7	--	/var/lib/mysql
cool_poincare	mysql:5.7	--	/var/lib/mysql