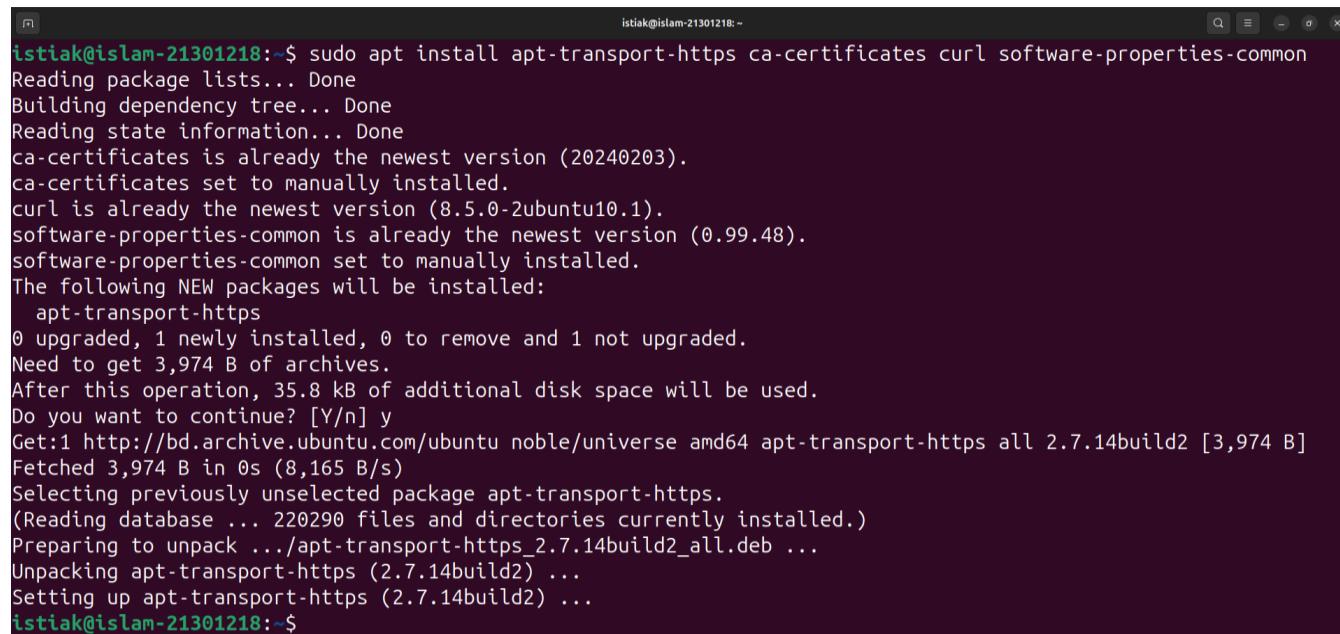


Install Docker

For installing the latest docker hub we need to install it from the official docker hub repository. To install few prerequisites so that apt use packages over HTTPS.

```
sudo apt install apt-transport-https ca-certificates curl software-properties-common
```



```
istiaik@islam-21301218:~$ sudo apt install apt-transport-https ca-certificates curl software-properties-common
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20240203).
ca-certificates set to manually installed.
curl is already the newest version (8.5.0-2ubuntu10.1).
software-properties-common is already the newest version (0.99.48).
software-properties-common set to manually installed.
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 1 not upgraded.
Need to get 3,974 B of archives.
After this operation, 35.8 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://bd.archive.ubuntu.com/ubuntu noble/universe amd64 apt-transport-https all 2.7.14build2 [3,974 B]
Fetched 3,974 B in 0s (8,165 B/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 220290 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.7.14build2_all.deb ...
Unpacking apt-transport-https (2.7.14build2) ...
Setting up apt-transport-https (2.7.14build2) ...
istiaik@islam-21301218:~$
```

To add the GPG key to my system for official Docker repository we will use,

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
```

To add the docker repository,

```
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"
```

```
istiak@islam-21301218:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).
OK
istiak@islam-21301218:~$ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"
Repository: 'deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable'
Description:
Archive for codename: focal components: stable
More info: https://download.docker.com/linux/ubuntu
Adding repository.
Press [ENTER] to continue or Ctrl-c to cancel.
Adding deb entry to /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-noble.list
Adding disabled deb-src entry to /etc/apt/sources.list.d/archive_uri-https_download_docker_com_linux_ubuntu-noble.list
Get:1 https://download.docker.com/linux/ubuntu focal InRelease [57.7 kB]
Get:2 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages [46.4 kB]
Hit:3 https://packages.microsoft.com/repos/code stable InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:5 http://bd.archive.ubuntu.com/ubuntu noble InRelease
Ign:6 https://ppa.launchpadcontent.net/obsproject/obs-studio/ubuntu noble InRelease
Err:7 https://ppa.launchpadcontent.net/obsproject/obs-studio/ubuntu noble Release
  404 Not Found [IP: 185.125.190.80 443]
Hit:8 http://bd.archive.ubuntu.com/ubuntu noble-updates InRelease
```

Installing Docker from the Docker repository,

```
apt-cache policy docker-ce
```

Finally, we can install docker in our host by,

```
sudo apt install docker-ce
```

```
istiak@islam-21301218:~$ apt-cache policy docker-ce
docker-ce:
  Installed: (none)
  Candidate: 5:27.0.3-1~ubuntu.20.04~focal
  Version table:
    5:27.0.3-1~ubuntu.20.04~focal 500
      500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
    5:27.0.2-1~ubuntu.20.04~focal 500
      500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
    5:27.0.1-1~ubuntu.20.04~focal 500
      500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
    5:26.1.4-1~ubuntu.20.04~focal 500
      500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
    5:26.1.3-1~ubuntu.20.04~focal 500
      500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
    5:26.1.2-1~ubuntu.20.04~focal 500
      500 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages
    5:26.1.1-1~ubuntu.20.04~focal 500
```

```
istiaik@islam-21301218:~$ sudo apt install docker-ce
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  containerd.io docker-buildx-plugin docker-ce-cli docker-ce-rootless-extras docker-compose-plugin pigz
  slirp4netns
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite
The following NEW packages will be installed:
  containerd.io docker-buildx-plugin docker-ce docker-ce-rootless-extras docker-compose-plugin
  pigz slirp4netns
0 upgraded, 8 newly installed, 0 to remove and 1 not upgraded.
Need to get 122 MB of archives.
After this operation, 436 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 https://download.docker.com/linux/ubuntu focal/stable amd64 containerd.io amd64 1.7.18-1 [30.5 MB]
Get:2 http://bd.archive.ubuntu.com/ubuntu noble/universe amd64 pigz amd64 2.8-1 [65.6 kB]
Get:3 http://bd.archive.ubuntu.com/ubuntu noble/universe amd64 slirp4netns amd64 1.2.1-1build2 [34.9 kB]
Get:4 https://download.docker.com/linux/ubuntu focal/stable amd64 docker-buildx-plugin amd64 0.15.1-1~ubuntu.20.04~focal [29.8 MB]
```

We can check our docker using this command,

```
sudo systemctl status docker
```

```
istiaik@islam-21301218:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
   Active: active (running) since Sat 2024-07-06 12:13:29 +06; 14min ago
     TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
      Main PID: 17159 (dockerd)
        Tasks: 13
       Memory: 23.3M (peak: 24.8M)
         CPU: 295ms
        CGroup: /system.slice/docker.service
                  └─17159 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Jul 06 12:13:29 islam-21301218 systemd[1]: Starting docker.service - Docker Application Container Engine...
Jul 06 12:13:29 islam-21301218 dockerd[17159]: time="2024-07-06T12:13:29.363442194+06:00" level=info msg="Start"
Jul 06 12:13:29 islam-21301218 dockerd[17159]: time="2024-07-06T12:13:29.366136134+06:00" level=info msg="detec"
Jul 06 12:13:29 islam-21301218 dockerd[17159]: time="2024-07-06T12:13:29.442201755+06:00" level=info msg="Loadi"
Jul 06 12:13:29 islam-21301218 dockerd[17159]: time="2024-07-06T12:13:29.610430156+06:00" level=info msg="Loadi"
Jul 06 12:13:29 islam-21301218 dockerd[17159]: time="2024-07-06T12:13:29.623441498+06:00" level=info msg="Docker"
Jul 06 12:13:29 islam-21301218 dockerd[17159]: time="2024-07-06T12:13:29.623525706+06:00" level=info msg="Daemo"
Jul 06 12:13:29 islam-21301218 dockerd[17159]: time="2024-07-06T12:13:29.660698155+06:00" level=info msg="API l"
Jul 06 12:13:29 islam-21301218 systemd[1]: Started docker.service - Docker Application Container Engine.
lines 1-21/21 (END)
```

So far, our docker is running successfully.

Basic Commands of Docker

First, check the docker version,

```
sudo docker --version
```

```
istiak@islam-21301218:~$ sudo docker --version
Docker version 27.0.3, build 7d4bcd8
istiak@islam-21301218:~$
```

Then, we can check the configuration of docker,

```
sudo docker config
```

```
istiak@islam-21301218:~$ sudo docker config

Usage: docker config COMMAND

Manage Swarm configs

Commands:
  create      Create a config from a file or STDIN
  inspect     Display detailed information on one or more configs
  ls          List configs
  rm          Remove one or more configs

Run 'docker config COMMAND --help' for more information on a command.
istiak@islam-21301218:~$
```

Then, using pull command we pull ubuntu,

```
sudo docker pull ubuntu
```

```
istiak@islam-21301218:~$ sudo docker pull ubuntu
[sudo] password for istiak:
Using default tag: latest
latest: Pulling from library/ubuntu
9c704ecd0c69: Pull complete
Digest: sha256:2e863c44b718727c860746568e1d54af13b2fa71b160f5cd9058fc436217b30
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
istiak@islam-21301218:~$
```

Then, use ‘search’ command,

```
sudo docker search ubuntu
```

```
istiak@islam-21301218:~$ sudo docker search ubuntu
NAME          DESCRIPTION               STARS      OFFICIAL
ubuntu        Ubuntu is a Debian-based Linux operating sys... 17143      [OK]
ubuntu-debootstrap DEPRECATED; use "ubuntu" instead           52         [OK]
open-liberty   Open Liberty multi-architecture images based... 65         [OK]
neurodebian   NeuroDebian provides neuroscience research s... 110        [OK]
ubuntu-upstart DEPRECATED, as is Upstart (find other proces... 115        [OK]
websphere-liberty WebSphere Liberty multi-architecture images ... 299        [OK]
ubuntu/nginx  Nginx, a high-performance reverse proxy & we... 114
ubuntu/squid   Squid is a caching proxy for the Web. Long-t... 92
ubuntu/cortex   Cortex provides storage for Prometheus. Long... 4
ubuntu/apache2  Apache, a secure & extensible open-source HT... 74
ubuntu/kafka    Apache Kafka, a distributed event streaming ...
ubuntu/bind9    BIND 9 is a very flexible, full-featured DNS...
ubuntu/mysql   MySQL open source fast, stable, multi-thread...
ubuntu/redis   Redis, an open source key-value store. Long-...
ubuntu/jre     Distroless Java runtime based on Ubuntu. Lon...
ubuntu/prometheus Prometheus is a systems and service monitori...
ubuntu/zookeeper ZooKeeper maintains configuration informatio...
ubuntu/postgres PostgreSQL is an open source object-relation...
ubuntu/memcached Memcached, in-memory keyvalue store for smal...
ubuntu/cassandra Cassandra, an open source NoSQL distributed ...
ubuntu/prometheus-alertmanager Alertmanager handles client alerts from Prom...
ubuntu/grafana  Grafana, a feature rich metrics dashboard & ...
ubuntu/dotnet-aspNet Chiselled Ubuntu runtime image for ASP.NET a...
ubuntu/dotnet-deps Chiselled Ubuntu for self-contained .NET & A...
ubuntu/dotnet-runtime Chiselled Ubuntu runtime image for .NET apps... 20
istiak@islam-21301218:~$
```

Then, we can use ‘*docker run*’ to run a container,

```
sudo docker run hello-world
```

```
istiak@islam-21301218:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
c1ec31eb5944: Pull complete
Digest: sha256:94323f3e5e09a8b9515d74337010375a456c909543e1ff1538f5116d38ab3989
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/
```

Then, we can see all the running container,

```
sudo docker ps --all
```

```
istiak@islam-21301218:~$ sudo docker ps --all
CONTAINER ID   IMAGE     COMMAND   CREATED      STATUS          PORTS     NAMES
0c4656a97c06   hello-world   "/hello"  12 minutes ago   Exited (0)  12 minutes ago   kind_black
istiak@islam-21301218:~$
```

Then, to start a docker using container id,

```
sudo docker start <id>
```

```
istiak@islam-21301218:~$ sudo docker start 0c4656a97c06
0c4656a97c06
```

Then, we can ‘images’ to see all the pulled Images in our system,

```
sudo docker images
```

```
istiak@islam-21301218:~$ sudo docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
ubuntu          latest   35a88802559d  4 weeks ago  78.1MB
hello-world     latest   d2c94e258dcb  14 months ago 13.3kB
istiak@islam-21301218:~$
```

Create a Docker image using Dockerfile

For creating a Docker file first, we need to create a directory as ‘*myDocker*’ then create a file named ‘*Dockerfile*’ and write the following code as show, Use ‘*cat ‘Dockerfile’*’ to see the contents of that file.

Now the crucial part, to make a *Dockerfile* inside myDocker. Create a file named ‘*Dockerfile*’ and write some basic commands into the file.

```
istiak@islam-21301218:~$ cd
istiak@islam-21301218: $ cd myDocker/
istiak@islam-21301218:~/myDocker$ ls
Dockerfile hello.sh
istiak@islam-21301218:~/myDocker$ cat Dockerfile
# Use the official Ubuntu image as the base image
FROM ubuntu:latest

# Set the maintainer label
LABEL maintainer="your-email@example.com"

# Update the package repository and install some packages
RUN apt-get update && apt-get install -y \
    curl \
    vim \
    && rm -rf /var/lib/apt/lists/*

# Set the working directory
WORKDIR /app

# Copy a local file into the container
COPY hello.sh /app/

# Make the script executable
```

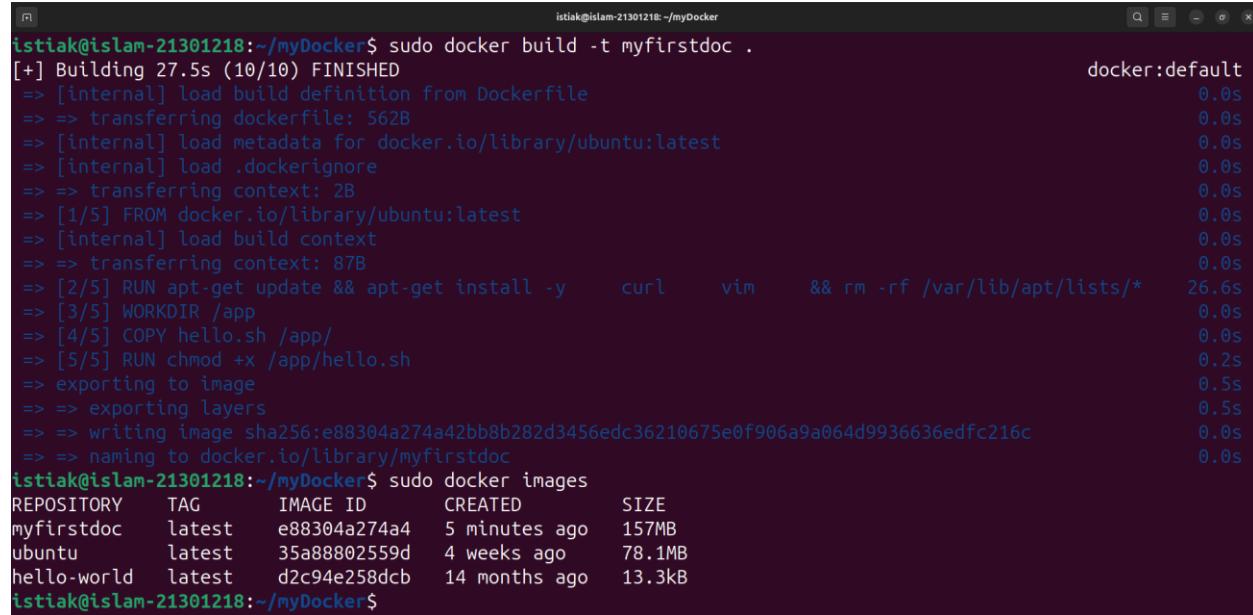
```
mkdir myDocker
cd myDocker
nano Dockerfile
```

Now we can create a Dockerfile. I have named myfirstdoc as the repo name.

'sudo docker build -t myfirstdoc'

Moreover, we can see our newly created and existing repo using,

```
sudo docker images
```



A screenshot of a terminal window titled 'istiak@islam-21301218: ~/myDocker'. The window shows the output of a 'sudo docker build -t myfirstdoc .' command, which completed successfully. It then lists the images with 'sudo docker images', showing three entries: 'myfirstdoc' (latest tag, 157MB), 'ubuntu' (latest tag, 78.1MB), and 'hello-world' (latest tag, 13.3kB). The terminal has a dark background with light-colored text.

```
istiak@islam-21301218:~/myDocker$ sudo docker build -t myfirstdoc .
[+] Building 27.5s (10/10) FINISHED
--> [internal] load build definition from Dockerfile
--> => transferring dockerfile: 562B
--> [internal] load metadata for docker.io/library/ubuntu:latest
--> [internal] load .dockerignore
--> => transferring context: 2B
--> [1/5] FROM docker.io/library/ubuntu:latest
--> [internal] load build context
--> => transferring context: 87B
--> [2/5] RUN apt-get update && apt-get install -y curl vim && rm -rf /var/lib/apt/lists/*
--> [3/5] WORKDIR /app
--> [4/5] COPY hello.sh /app/
--> [5/5] RUN chmod +x /app/hello.sh
--> exporting to image
--> => exporting layers
--> => writing image sha256:e88304a274a42bb8b282d3456edc36210675e0f906a9a064d9936636edfc216c
--> => naming to docker.io/library/myfirstdoc
istiak@islam-21301218:~/myDocker$ sudo docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
myfirstdoc      latest   e88304a274a4  5 minutes ago  157MB
ubuntu          latest   35a88802559d  4 weeks ago   78.1MB
hello-world     latest   d2c94e258dcb  14 months ago  13.3kB
istiak@islam-21301218:~/myDocker$
```

Run a container as a single task

To run the container, I have use 'run' and the '*Image Id*'. First I need to check the existing images by 'sudo docker images' then apply this command to run the container,

```
sudo docker run e88304a274a4
```

```
istiak@islam-21301218:~/myDocker$ sudo docker images
REPOSITORY      TAG          IMAGE ID      CREATED        SIZE
myfirstdoc      latest       e88304a274a4  9 minutes ago  157MB
ubuntu          latest       35a88802559d  4 weeks ago   78.1MB
hello-world     latest       d2c94e258dcb  14 months ago  13.3kB
istiak@islam-21301218:~/myDocker$ sudo docker run e88304a274a4
Hello from the Docker container!
istiak@islam-21301218:~/myDocker$
```

For displaying the status, we will use ‘*docker ps -a*’

```
istiak@islam-21301218:~$ sudo docker ps -a
CONTAINER ID   IMAGE      COMMAND           CREATED        STATUS          PORTS     NAMES
7a40843cbaa6   e88304a274a4  "/app/hello.sh"   4 minutes ago  Exited (0) 4 minutes ago
_mayer
0c4656a97c06   hello-world  "/hello"         About an hour ago   Exited (0) 49 minutes ago
lack
istiak@islam-21301218:~$
```

Run a container in iterative mode

To run container in the iterative mood we will use –it. First I need to check all the existed container. Then run this command,

```
sudo docker run -it 35a88802559d
```

```

istiaik@islam-21301218:~$ sudo docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
myfirstdoc     latest    e88304a274a4  16 minutes ago  157MB
ubuntu          latest    35a88802559d  4 weeks ago   78.1MB
hello-world     latest    d2c94e258dcf  14 months ago  13.3kB
istiaik@islam-21301218:~$ sudo docker run -it 35a88802559d
root@c8c410ccf2c3:/# apt update
Get:1 http://archive.ubuntu.com/ubuntu noble InRelease [256 kB]
Get:2 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:3 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [239 kB]
Get:4 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:5 http://archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:6 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [181 kB]
Get:7 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [12.7 kB]
Get:8 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [78.1 kB]
Get:9 http://archive.ubuntu.com/ubuntu noble/universe amd64 Packages [19.3 MB]
Get:10 http://archive.ubuntu.com/ubuntu noble/restricted amd64 Packages [117 kB]
Get:11 http://archive.ubuntu.com/ubuntu noble/main amd64 Packages [1808 kB]
Get:12 http://archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [331 kB]
Get:13 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [275 kB]
Get:14 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [142 kB]
Get:15 http://archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [181 kB]
Get:16 http://archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [16.9 kB]
Get:17 http://archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [9711 B]
Fetched 23.3 MB in 13s (1849 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done

```

Then '*apt install mypaint*' for install my paint software.

```

root@c8c410ccf2c3:/# apt install mypaint
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  adduser adwaita-icon-theme at-spi2-common at-spi2-core ca-certificates dbus dbus-bin dbus-daemon
  dbus-session-bus-common dbus-system-bus-common dbus-user-session dconf-gsettings-backend dconf-service
  dmsetup fontconfig fontconfig-config fonts-dejavu-core fonts-dejavu-mono gir1.2-atk-1.0 gir1.2-freedesktop
  gir1.2-gdkpixbuf-2.0 gir1.2-girepository-2.0 gir1.2-glib-2.0 gir1.2-gtk-3.0 gir1.2-harfbuzz-0.0
  gir1.2-pango-1.0 gsettings-desktop-schemas gtk-update-icon-cache hicolor-icon-theme humanity-icon-theme
  krb5-locales libapparmor1 libargon2-1 libatk-bridge2.0-0t64 libatk1.0-0t64 libatspi2.0-0t64 libavahi-client3
  libavahi-common-data libavahi-common3 libblas3 libbrotlii libbsd0 libcairo-gobject2 libcairo2 libcolord2
  libcryptsetup12 libcurl2t64 libdatrie1 libdbus-1-3 libdconf1 libdeflate0 libdevmapper1.02.1 libepoxy0
  libexpat1 libfdisk1 libfontconfig1 libfreetype6 libfribidi0 libgdk-pixbuf2.0-0 libgdk-pixbuf2.0-bin

```

So far, we have installed a different package in the container.

Run a database container in the background

For run a database we need to pull the images, in my case I have use mySql database.

```
sudo docker pull mysql
```

```
istiak@islam-21301218:~$ sudo docker pull mysql
Using default tag: latest
latest: Pulling from library/mysql
7af76bb36546: Pull complete
db774776bbe8: Pull complete
8b850c913cab: Pull complete
f3d9d23107fd: Pull complete
1e5123b24fcc: Pull complete
1c0467c26f4a: Pull complete
f65dd49246d7: Pull complete
08151edac83e: Pull complete
7b4cbb0e2b3a: Pull complete
36c68f7d2e61: Pull complete
Digest: sha256:8b879a3959bc59adcb7281a41950d39cf8c9b3fb23b87b9b62318ce884a7c383
Status: Downloaded newer image for mysql:latest
docker.io/library/mysql:latest
istiak@islam-21301218:~$
```

Now for running mysql container we need to run this command,

```
sudo docker run --name=istsql -d mysql/mysql-server
```

```
istiak@islam-21301218:~$ sudo docker run --name=istsql -d mysql/mysql-server
Unable to find image 'mysql/mysql-server:latest' locally
latest: Pulling from mysql/mysql-server
6a4a3ef82cdc: Pull complete
5518b09b1089: Pull complete
b6b576315b62: Pull complete
349b52643cc3: Pull complete
abe8d2406c31: Pull complete
c7668948e14a: Pull complete
c7e93886e496: Pull complete
Digest: sha256:d6c8301b7834c5b9c2b733b10b7e630f441af7bc917c74dba379f24eeeb6a313
Status: Downloaded newer image for mysql/mysql-server:latest
2dd3d27a9d9c53035e90e8ff229e16ad4ba37a39aa1df42e8a69e6bf105f364e
istiak@islam-21301218:~$
```

To see the root password, we need to run this command,

```
sudo docker logs istsql
```

```
istiak@islam-21301218:~$ sudo docker logs istsql
[Entrypoint] MySQL Docker Image 8.0.32-1.2.11-server
[Entrypoint] No password option specified for new database.
[Entrypoint] A random onetime password will be generated.
[Entrypoint] Initializing database
2024-07-06T09:44:47.378954Z 0 [Warning] [MY-011068] [Server] The syntax '--skip-host-cache' is deprecated and wi
ll be removed in a future release. Please use SET GLOBAL host_cache_size=0 instead.
2024-07-06T09:44:47.379046Z 0 [System] [MY-013169] [Server] /usr/sbin/mysqld (mysqld 8.0.32) initializing of ser
ver in progress as process 17
2024-07-06T09:44:47.385528Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
2024-07-06T09:44:47.884294Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
2024-07-06T09:44:49.044150Z 6 [Warning] [MY-010453] [Server] root@localhost is created with an empty password !
Please consider switching off the --initialize-insecure option.
[Entrypoint] Database initialized
2024-07-06T09:44:52.941013Z 0 [Warning] [MY-011068] [Server] The syntax '--skip-host-cache' is deprecated and wi
ll be removed in a future release. Please use SET GLOBAL host_cache_size=0 instead.
2024-07-06T09:44:52.942113Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld (mysqld 8.0.32) starting as process
60
2024-07-06T09:44:52.958859Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
2024-07-06T09:44:53.062663Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
2024-07-06T09:44:53.274039Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem is self signed.
2024-07-06T09:44:53.274083Z 0 [System] [MY-013602] [Server] Channel mysql_main configured to support TLS. Encryp
ted connections are now supported for this channel.
2024-07-06T09:44:53.299295Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '8
.0.32' socket: '/var/lib/mysql/mysql.sock' port: 0 MySQL Community Server - GPL.
2024-07-06T09:44:53.299298Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Socket: /var/run/mys
qld/mysqlx.sock
Warning: Unable to load '/usr/share/zoneinfo/iso3166.tab' as time zone. Skipping it.
```

Here is the root password.

GENERATED ROOT PASSWORD: ..*q2%2mNp5mpaZn8#79EyC8,&Ur;8nT

```
istiak@islam-21301218:~$
Warning: Unable to load '/usr/share/zoneinfo/iso3166.tab' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/leapseconds' as time zone. Skipping it.
Warning: Unable to load '/usr/share/zoneinfo/tzdata.zi' 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.
[Entrypoint] GENERATED ROOT PASSWORD: ..*q2%2mNp5mpaZn8#79EyC8,&Ur;8nT

[Entrypoint] ignoring /docker-entrypoint-initdb.d/*
2024-07-06T09:44:54.106925Z 11 [System] [MY-013172] [Server] Received SHUTDOWN from user root. Shutting down mys
qld (Version: 8.0.32).
2024-07-06T09:44:55.301780Z 0 [System] [MY-010910] [Server] /usr/sbin/mysqld: Shutdown complete (mysqld 8.0.32)
MySQL Community Server - GPL.
[Entrypoint] Server shut down
[Entrypoint] Setting root user as expired. Password will need to be changed before database can be used.

[Entrypoint] MySQL init process done. Ready for start up.
```

I can enter the database. For that I need to run this command,

```
sudo docker exec -it istsql mysql -u root -p
```

In the password section we have provided the generated password which is

..*q2%2mNp5mpaZn8#79EyC8,&Ur;8nT

```
istiak@islam-21301218:~$ sudo docker exec -it istsql mysql -u root -p  
Enter password:  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 30  
Server version: 8.0.32  
  
Copyright (c) 2000, 2023, 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>
```

For run SQL we need to change the password. So, we need to run this command,

```
ALTER USER 'root'@'localhost' IDENTIFIED BY '8878';
```

```
istiak@islam-21301218:~$  
Query OK, 0 rows affected (0.01 sec)  
  
mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY '8878';  
Query OK, 0 rows affected (0.01 sec)  
  
mysql>
```

Now we will run some SQL queries and see the output. In that case, I have created database named '*doc_database*'

```
mysql> create database doc_database
      -> ;
Query OK, 1 row affected (0.01 sec)

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      |
+-----+
| doc_database  |
| information_schema |
| mysql          |
| performance_schema |
| sys            |
+-----+
5 rows in set (0.01 sec)

mysql> 
```

Push your own image into the Docker public registry/Hub

First, we need to sign up to docker hub. Then we can log in from cli using,

```
docker login
```

```
istiak@islam-21301218:~$ sudo docker login
Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com/ to create one.
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better security and is required for organizations using SSO. Learn more at https://docs.docker.com/go/access-tokens/
Username: istiak99
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credential-stores

Login Succeeded
istiak@islam-21301218:~$
```

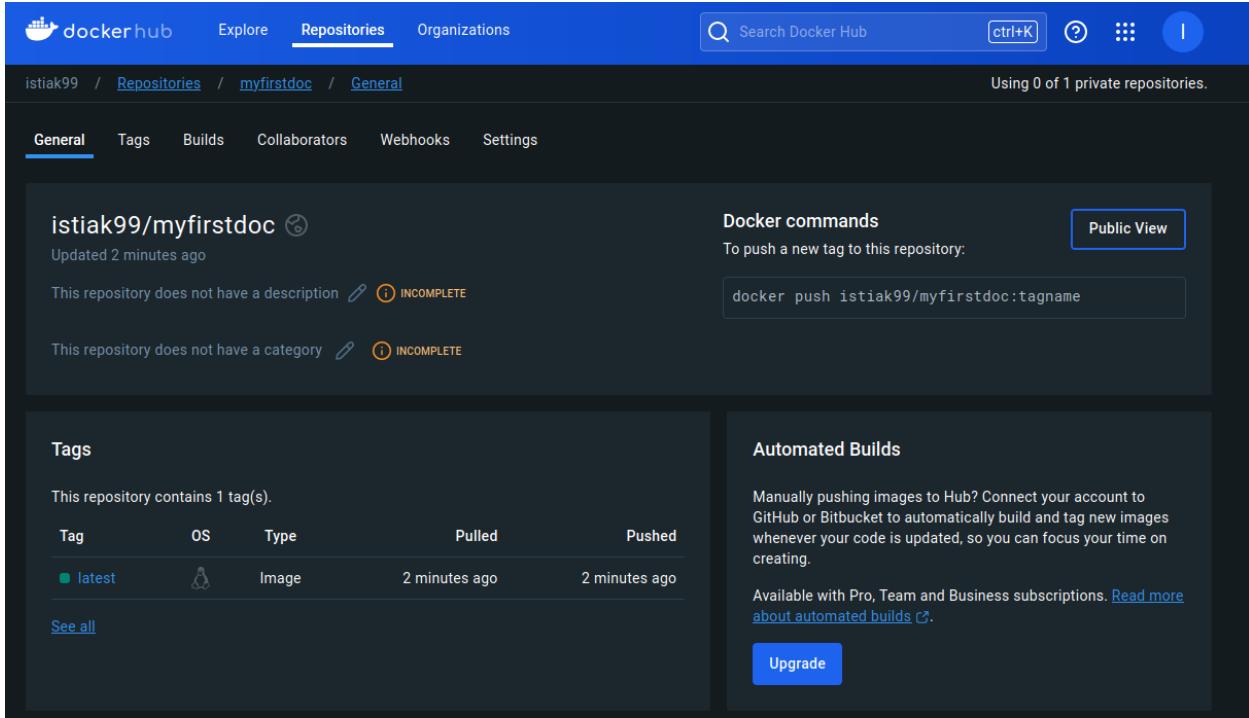
After successful login, we need to tag docker Image and push the docker image.

So, we need run the following command,

```
sudo docker tag myfirstdoc istiak99/myfirstdoc:latest
sudo docker push istiak99/myfirstdoc:lates
```

```
istiak@islam-21301218:~$ sudo docker tag myfirstdoc istiak99/myfirstdoc:latest
istiak@islam-21301218:~$ sudo docker push istiak99/myfirstdoc:latest
The push refers to repository [docker.io/istiak99/myfirstdoc]
e251a9c53d31: Pushed
fc742a1d7bd6: Pushed
6d74fd3565fb: Pushed
f378794b31c1: Pushed
a30a5965a4f7: Mounted from library/ubuntu
latest: digest: sha256:9374cd360cf0c871c1680b3bfb3d6200a6dbd07a92ddc22c7ffc4272bcc621da size: 1361
istiak@islam-21301218:~$
```

After successful pushing we can go to docker hub repository and can see the '*myfirstdoc*' has been created.



The screenshot shows the Docker Hub interface for a repository named 'istiak99/myfirstdoc'. The 'General' tab is selected. At the top right, there's a search bar with placeholder text 'Search Docker Hub' and a 'ctrl+K' keyboard shortcut. Below the search bar are icons for help, settings, and a three-dot menu. The repository path 'istiak99 / Repositories / myfirstdoc / General' is displayed at the top left, along with a message 'Using 0 of 1 private repositories.' On the left, there are tabs for 'General', 'Tags', 'Builds', 'Collaborators', 'Webhooks', and 'Settings'. The 'General' tab is active. The main content area shows the repository name 'istiak99/myfirstdoc' with a circular progress icon. It says 'Updated 2 minutes ago'. Below that, it states 'This repository does not have a description' with an edit icon and 'INCOMPLETE' status. Another note says 'This repository does not have a category' with an edit icon and 'INCOMPLETE' status. To the right, under 'Docker commands', it says 'To push a new tag to this repository:' followed by a command line box containing 'docker push istiak99/myfirstdoc:tagname'. A 'Public View' button is also present. In the bottom left, there's a 'Tags' section showing one tag: 'latest' (Image type, pushed 2 minutes ago). In the bottom right, there's a 'Automated Builds' section with a note about connecting GitHub or Bitbucket for automatic builds, availability for Pro, Team, and Business subscriptions, and an 'Upgrade' button.

How to make your own private registry

For making own private registry we can run this command,

```
sudo docker pull registry
```

```
istiak@islam-21301218:~$ sudo docker pull registry
Using default tag: latest
latest: Pulling from library/registry
73baa7ef167e: Pull complete
d49090716641: Pull complete
bc8f2b8a18ff: Pull complete
9d41963883ad: Pull complete
ad02dd2076d6: Pull complete
Digest: sha256:79b29591e1601a73f03fc413e655b72b9abfae5a23f1ad2e883d4942fb4351
Status: Downloaded newer image for registry:latest
docker.io/library/registry:latest
istiak@islam-21301218:~$
```

Then we need to install docker compose by,

```
sudo apt install docker-compose
```

```
istiak@islam-21301218:~$ sudo apt install docker-compose
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  python3-compose python3-docker python3-dockerpty python3-docopt python3-dotenv
  python3-texttable python3-websocket
Recommended packages:
  docker.io
The following NEW packages will be installed:
  docker-compose python3-compose python3-docker python3-dockerpty python3-docopt
  python3-dotenv python3-texttable python3-websocket
0 upgraded, 8 newly installed, 0 to remove and 1 not upgraded.
Need to get 297 kB of archives.
After this operation, 1,589 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://bd.archive.ubuntu.com/ubuntu noble/universe amd64 python3-websocket all 1.7.0-1 [38.1 kB]
Get:2 http://bd.archive.ubuntu.com/ubuntu noble/universe amd64 python3-docker all 5.0.3-1ubuntu1 [89.1 kB]
```

Run the following command to run, tag and push it to private registry.

```
sudo docker run -d -p 5000:5000 --name registry registry:2
sudo docker tag myfirstdoc localhost:5000/myfirstdoc
sudo docker push localhost:5000/myfirstdoc
```

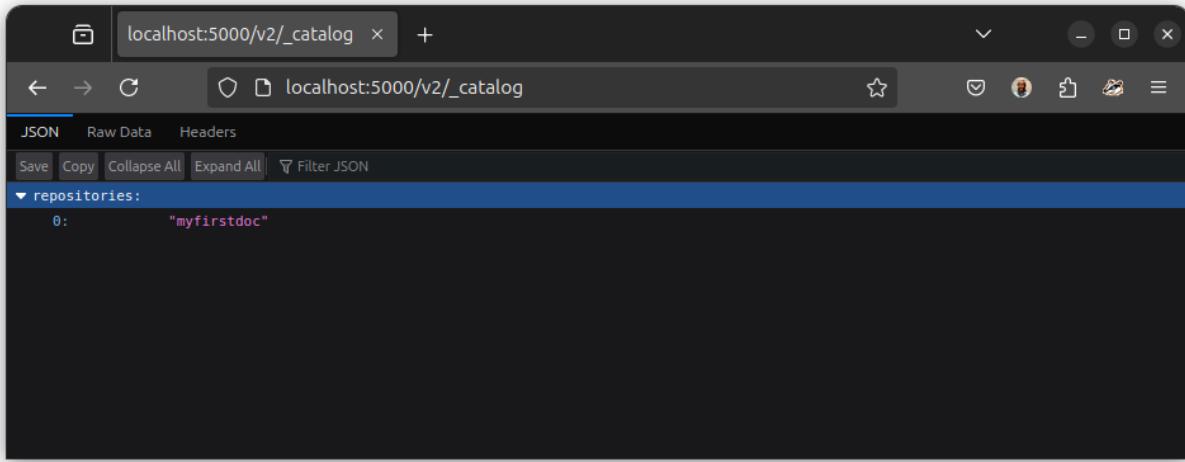
```
istiak@islam-21301218:~$ sudo docker run -d -p 5000:5000 --name registry registry:2
Unable to find image 'registry:2' locally
2: Pulling from library/registry
Digest: sha256:79b29591e1601a73f03fc413e655b72b9abfae5a23f1ad2e883d4942fb4351
Status: Downloaded newer image for registry:2
356aba008101cf9b7908c1611d0e2a649597b85add110adb1902f75d6842ae5e
istiak@islam-21301218:~$ sudo docker tag myfirstdoc localhost:5000/myfirstdoc
istiak@islam-21301218:~$ sudo docker push localhost:5000/myfirstdoc
Using default tag: latest
The push refers to repository [localhost:5000/myfirstdoc]
e251a9c53d31: Pushed
fc742a1d7bd6: Pushed
6d74fd3565fb: Pushed
f378794b31c1: Pushed
a30a5965a4f7: Pushed
latest: digest: sha256:8bfa1aaa3860f4cdd8dbcef0b63c7aa7b5d0f8ff36d41a387d90d6ca6e7f
0fe size: 1361
istiak@islam-21301218:~$
```

Let's see the private registered localhost container using docker images.

```
istiak@islam-21301218:~$ sudo docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
myfirstdoc          latest   e88304a274a4  3 hours ago  157MB
localhost:5000/myfirstdoc  latest   e88304a274a4  3 hours ago  157MB
istiak99/myfirstdoc  latest   e88304a274a4  3 hours ago  157MB
mysql               latest   31ebb0b19998  4 days ago   586MB
ubuntu              latest   35a88802559d  4 weeks ago  78.1MB
registry             2       6a3edb1d5eb6  9 months ago  25.4MB
registry             latest   6a3edb1d5eb6  9 months ago  25.4MB
hello-world          latest   d2c94e258dcf  14 months ago 13.3kB
mysql/mysql-server   latest   1d9c2219ff69  17 months ago 496MB
istiak@islam-21301218:~$
```

We can visit and see our private registry by using this link:

http://localhost:5000/v2/_catalog



Create a small website

To do that we need to do the following,

```
mkdir docweb  
cd docweb  
nano index.html  
cat index.html
```

```
istiak@islam-21301218:~$ mkdir docweb
istiak@islam-21301218:~$ cd docweb/
istiak@islam-21301218:~/docweb$ nano index.html
istiak@islam-21301218:~/docweb$ cat index.html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>My First Docker Webpage</title>
</head>
<body>
    <div>
        <h1>My First Docker Webpage</h1>
        <a href="https://hub.docker.com" target="https://hub.docker.com/repository/docker/istiak99/myfirstdoc/general">Go to Docker Hub</a>
    </div>
</body>
</html>
istiak@islam-21301218:~/docweb$
```

After creating the index.html we need to create Dockerfile for create container as like as before.

```
istiak@islam-21301218:~/docweb$ touch Dockerfile
istiak@islam-21301218:~/docweb$ ls
Dockerfile  index.html
istiak@islam-21301218:~/docweb$ nano Dockerfile
istiak@islam-21301218:~/docweb$ cat Dockerfile
FROM nginx:latest

COPY index.html /usr/share/nginx/html/
istiak@islam-21301218:~/docweb$
```

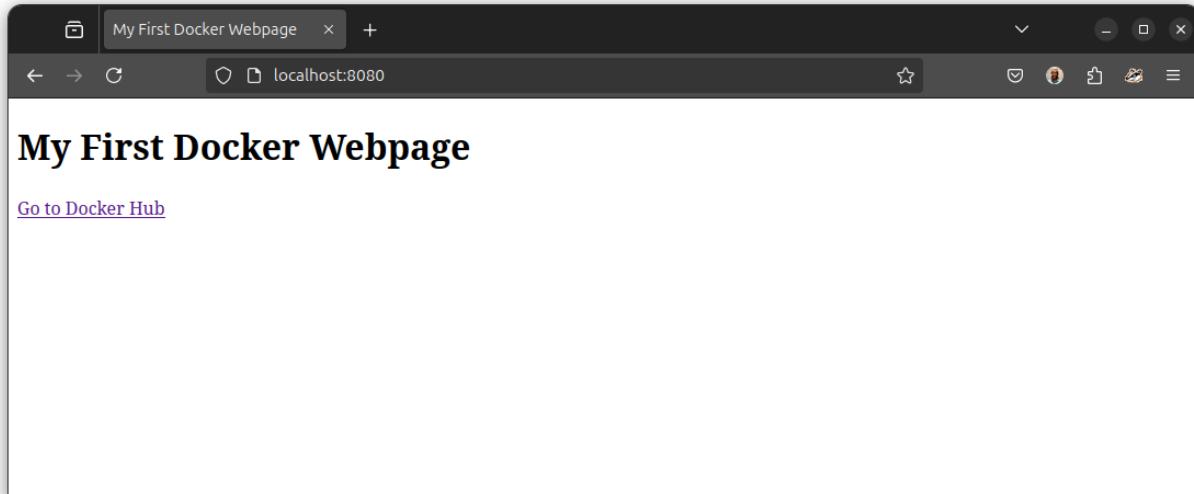
For build the docker image and run we use the following command,

```
docker build -t mydockerwebsite .
```

```
istiaik@islam-21301218:~/docweb$ sudo docker build -t mydockerwebsite .
[sudo] password for istiaik:
[+] Building 36.7s (8/8) FINISHED
=> [internal] load build definition from Dockerfile                               docker:default
=> => transferring dockerfile: 98B                                              0.0s
=> [internal] load metadata for docker.io/library/nginx:latest                  0.0s
=> [auth] library/nginx:pull token for registry-1.docker.io                      3.4s
=> [internal] load .dockerignore                                                 0.0s
=> => transferring context: 2B                                                 0.0s
=> [internal] load build context                                                0.0s
=> => transferring context: 457B                                              0.0s
=> [1/2] FROM docker.io/library/nginx:latest@sha256:67682bda769fae1ccf5183192b 33.0s
=> => resolve docker.io/library/nginx:latest@sha256:67682bda769fae1ccf5183192b8 0.0s
=> => sha256:67682bda769fae1ccf5183192b8daf37b64cae99c6c33026 10.27kB / 10.27kB 0.0s
=> => sha256:db5e49f40979ce521f05f0bc9f513d0abacce47904e229f3a9 2.29kB / 2.29kB 0.0s
=> => sha256:fffffc90d343c9cb01a5032edac86db5998c536cd0a3665141 7.30kB / 7.30kB 0.0s
=> => sha256:f11c1adaa26e078479ccdd45312ea3b88476441b91be0ec 29.13MB / 29.13MB 31.6s
=> => sha256:c6b156574604a095a5847d3b34cf36d484bb49862365e99 41.83MB / 41.83MB 24.8s
=> => sha256:ea5d7144c337402f813ea7c05c11dab58b7841f4c41fb5f5058abe 628B / 628B 1.0s
=> => sha256:1bbc9df2c93e03db739f7e49ce73eda0325b8087ef8e88386d303 955B / 955B 1.3s
```

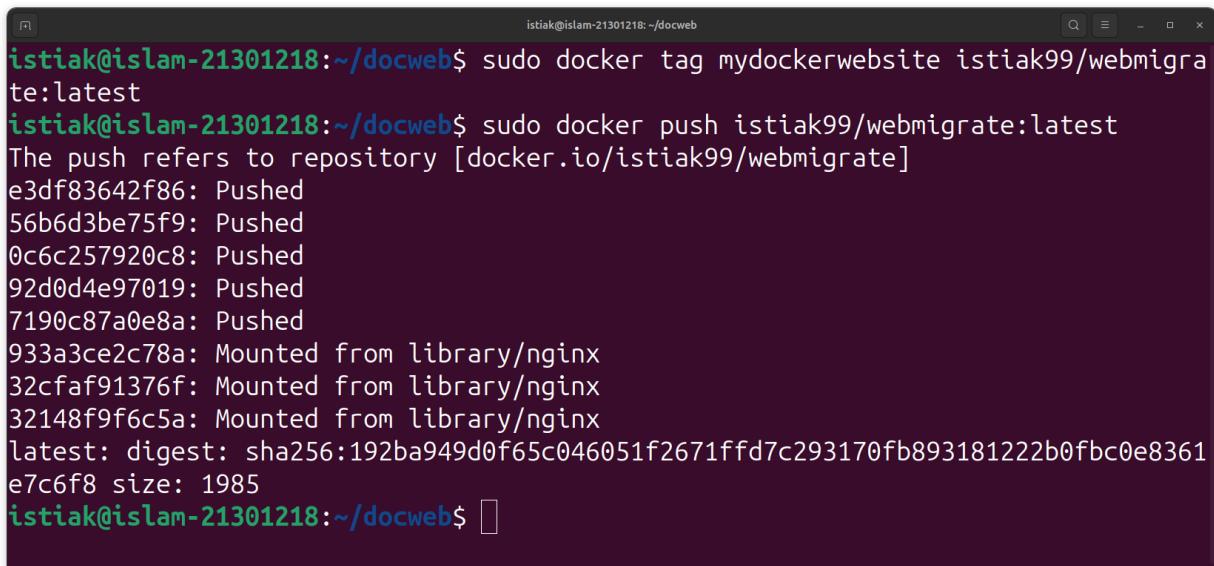
```
sudo docker run -d -p 8080:80 mydockerwebsite
```

```
istiaik@islam-21301218:~/docweb$ sudo docker run -d -p 8080:80 mydockerwebsite
a6809fd89877f9024d7783230c3c8e9324859aca350db8a0ba39a7a18207367b
istiaik@islam-21301218:~/docweb$
```



Migrate the new container having the application into another machine

To migrate first we need to push this into docker hub, using the same process.



```
istiak@islam-21301218:~/docweb$ sudo docker tag mydockerwebsite istiak99/webmigrate:latest
istiak@islam-21301218:~/docweb$ sudo docker push istiak99/webmigrate:latest
The push refers to repository [docker.io/istiak99/webmigrate]
e3df83642f86: Pushed
56b6d3be75f9: Pushed
0c6c257920c8: Pushed
92d0d4e97019: Pushed
7190c87a0e8a: Pushed
933a3ce2c78a: Mounted from library/nginx
32cfaf91376f: Mounted from library/nginx
32148f9f6c5a: Mounted from library/nginx
latest: digest: sha256:192ba949d0f65c046051f2671ffd7c293170fb893181222b0fbc0e8361
e7c6f8 size: 1985
istiak@islam-21301218:~/docweb$ 
```

Now we need to go to my friend's machine to see the migrated site. The following command should be run in my friend machine,

```
sudo docker pull istiak99.webmigrate:latest
sudo docker run -d -p 80:80 istiak99/webmigrate:latest
```

```
ahmad@habibullah-21301236:~$ sudo docker pull istiak99/webmigrate:latest
latest: Pulling from istiak99/webmigrate
f11c1adaa26e: Pull complete

c6b156574604: Pull complete

ea5d7144c337: Pull complete

1bbc9df2c93: Pull complete

537a6cfe3404: Pull complete

767bff2cc03e: Pull complete

adc73cb74f25: Pull complete

ae8e8f699a71: Pull complete

Digest: sha256:192ba949d0f65c046051f2671ffd7c293170fb893181222b0fb0e8361e7c6f8
Status: Downloaded newer image for istiak99/webmigrate:latest
docker.io/istiak99/webmigrate:latest
ahmad@habibullah-21301236:~$
```

```
ahmad@habibullah-21301236:~$ sudo docker run -d -p 80:80 istiak99/webmigrate:latest
25d44187a0e412da57eae8675e3a0892d2266061daa3b67617ea32c5e963274e
```

Here we can see the successful migration on my friend's machine. The website is running on my friend's machine. The site is running in

localhost:80

