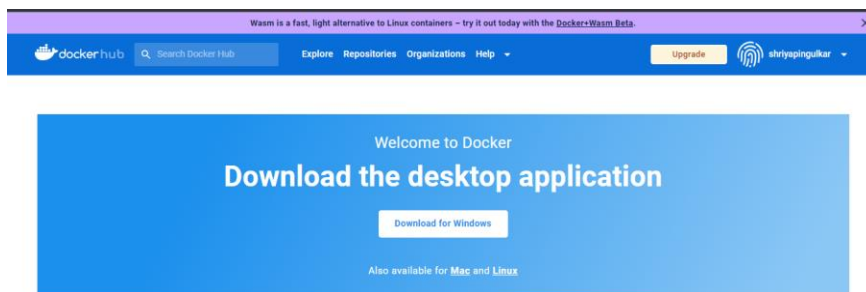


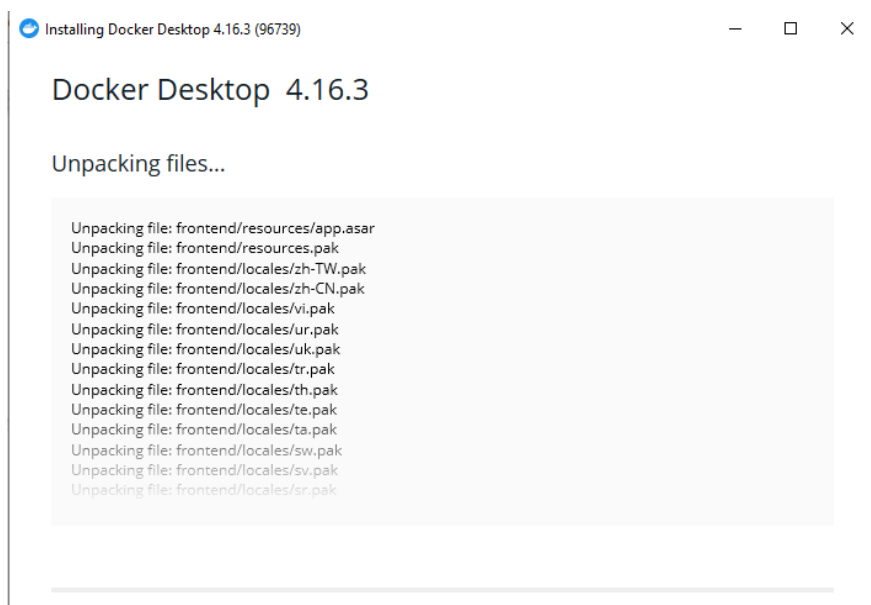
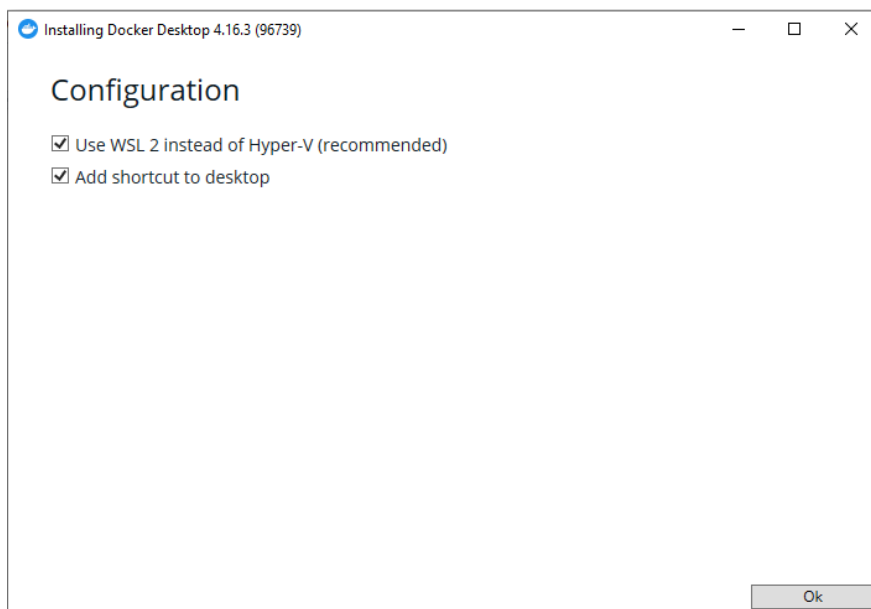
Aim: To installed Docker, pull and run images form Docker Hub, modify containers to run an application and push it on the Docker Hub.

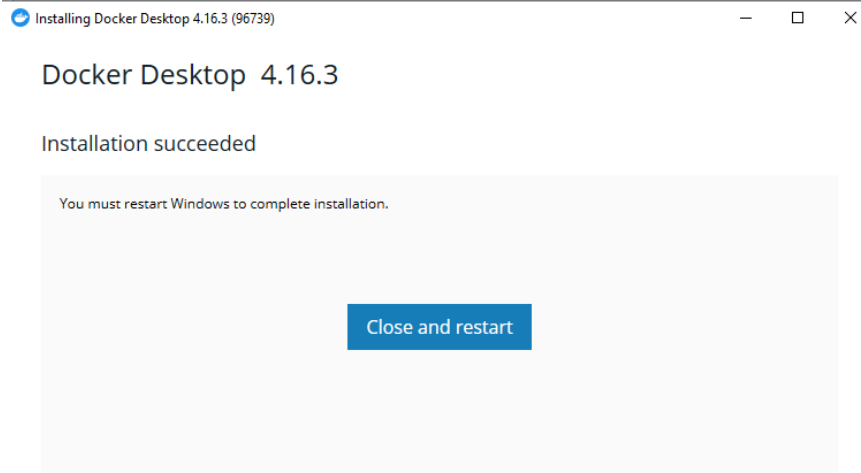
Results: (Document with screenshots)

1. All steps to Create Docker Hub Account and install Docker
2. 10 Docker commands
3. All commands to pull Nginx, set path, modify, website hosting and push
4. Updated Docker hub repository (before push and after push)
5. Repeat the same on VM or local machine for any other Image of your choice to run an application/program. [Hello world, Ubuntu etc.]

Docker login and installation



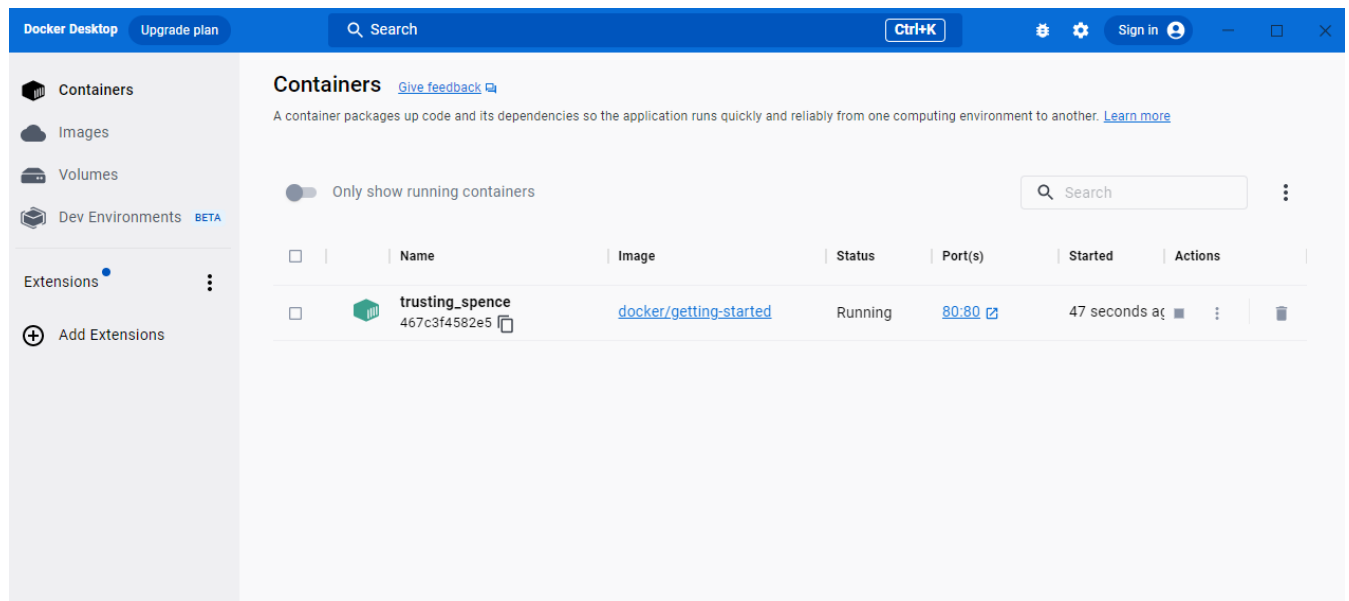




Getting started with docker and its commands

```
C:\Users\Exam>docker run -d -p 80:80 docker/getting-started
Unable to find image 'docker/getting-started:latest' locally
latest: Pulling from docker/getting-started
c158987b0551: Pull complete
1e35f6679fab: Pull complete
cb9626c74200: Pull complete
b6334b6ace34: Pull complete
f1d1c9928c82: Pull complete
9b6f639ec6ea: Pull complete
ee68d3549ec8: Pull complete
33e0cbbb4673: Pull complete
4f7e34c2de10: Pull complete
Digest: sha256:d79336f4812b6547a53e735480dde67f8f8f7071b414fbd9297609ffb989abc1
Status: Downloaded newer image for docker/getting-started:latest
467c3f4582e51bab0ef71a917dbc29b13fd096239309484efaa86bf58f6257ae

C:\Users\Exam>_
```



Commands:

1. docker --version

```
Command Prompt

C:\Users\Exam>docker --version
Docker version 20.10.22, build 3a2c30b

C:\Users\Exam>_
```

2. docker pull ubuntu

```
Command Prompt
C:\Users\Exam>docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
677076032cca: Pull complete
Digest: sha256:9a0bddde4188b896a372804be2384015e90e3f84906b750c1a53539b585fbb7f
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest

C:\Users\Exam>
```

3. docker run -it -d ubuntu

```
Command Prompt
C:\Users\Exam>docker run -it -d ubuntu
2408a3c88b20dca4b8bd407e31798153cf67bfb1abc6069c971bb83d26d5e

C:\Users\Exam>
```

4. docker ps

```
Command Prompt
C:\Users\Exam>docker run -it -d ubuntu
2408a3c88b20dca4b8bd407e31798153cf67bfb1abc6069c971bb83d26d5e

C:\Users\Exam>docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                    NAMES
2408a3c88b20   ubuntu    "/bin/bash"             38 seconds ago Up 36 seconds  0.0.0.0:80->80/tcp      sad_robinson
467c3f4582e5   docker/getting-started "/docker-entrypoint..." 5 minutes ago Up 5 minutes    0.0.0.0:80->80/tcp      trusting_spence

C:\Users\Exam>
```

5. docker ps -a

```
Command Prompt
C:\Users\Exam>docker ps -a
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                    NAMES
2408a3c88b20   ubuntu    "/bin/bash"             About a minute ago Up About a minute  0.0.0.0:80->80/tcp      sad_robinson
467c3f4582e5   docker/getting-started "/docker-entrypoint..." 6 minutes ago Up 6 minutes    0.0.0.0:80->80/tcp      trusting_spence

C:\Users\Exam>
```

6. docker exec -it 2408a3c88b20 bash

```
root@2408a3c88b20: /
C:\Users\Exam> docker exec -it 2408a3c88b20 bash
root@2408a3c88b20:/#
```

7. docker stop 2408a3c88b20

```
Command Prompt
exit
C:\Users\Exam> docker stop 2408a3c88b20
2408a3c88b20
C:\Users\Exam>
```

8. docker commit 2408a3c88b20 aayush/ubuntunew

```
Command Prompt
2408a3c88b20
C:\Users\Exam> docker commit 2408a3c88b20 aayush/ubuntunew
sha256:609374707a8b126dc70724952b9b40bd318551538ec6f13d23540641e5869627
C:\Users\Exam>
```

9. docker push aayush/ubuntunew

```
Command Prompt
C:\Users\Exam> docker commit 2408a3c88b20 aayush/ubuntunew
sha256:34461130bd009f280123dd5d09339f2a40e1f4f5d16af4bf687c0519d272021c
C:\Users\Exam> docker push aayush /ubuntunew
Using default tag: latest
The push refers to repository [docker.io/ aayush/ubuntunew]
a1fe17778315: Preparing
c5ff2d88f679: Preparing
denied: requested access to the resource is denied
C:\Users\Exam>
```

10. docker login

```
Command Prompt
Username: aayushshah
Password:
Error response from daemon: Get "https://registry-1.docker.io/v2/": unauthorized: incorrect username or password

C:\Users\Exam>docker login
Login with your Docker ID 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.
Username: aayushshah
Password:
Login Succeeded

Logging in with your password grants your terminal complete access to your account.
For better security, log in with a limited-privilege personal access token. Learn more at https://docs.docker.com/go/access-tokens/
```

Running a basic web server

1. `docker run -it --rm -d -p 8080:80 --name web nginx`

```
Select Command Prompt
Microsoft Windows [Version 10.0.19045.2604]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Exam>docker run -it --rm -d -p 8080:80 --name web nginx
ef77eb306b4d9db7a7941bc50828d0e7ce15ecde61262b56eb9c9760e8ba6bb3

C:\Users\Exam>
```

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

2. `docker stop web`

```
Command Prompt
C:\Users\Exam>docker stop web
web
C:\Users\Exam>
```

Adding Custom HTML

1. `docker run -it --rm -d -p 8080:80 --name web -v C:/Users/Exam/Desktop/docker:/usr/share/nginx/html nginx`

```
Command Prompt
Microsoft Windows [Version 10.0.19045.2604]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Exam>docker run -it --rm -d -p 8080:80 --name web -v C:/Users/Exam/Desktop/docker:/usr/share/nginx/html nginx
e3cf9a3f80c798ac13dba5b036594ecb924c18847c3912095b2fd8aede4cf4a1
C:\Users\Exam>
```

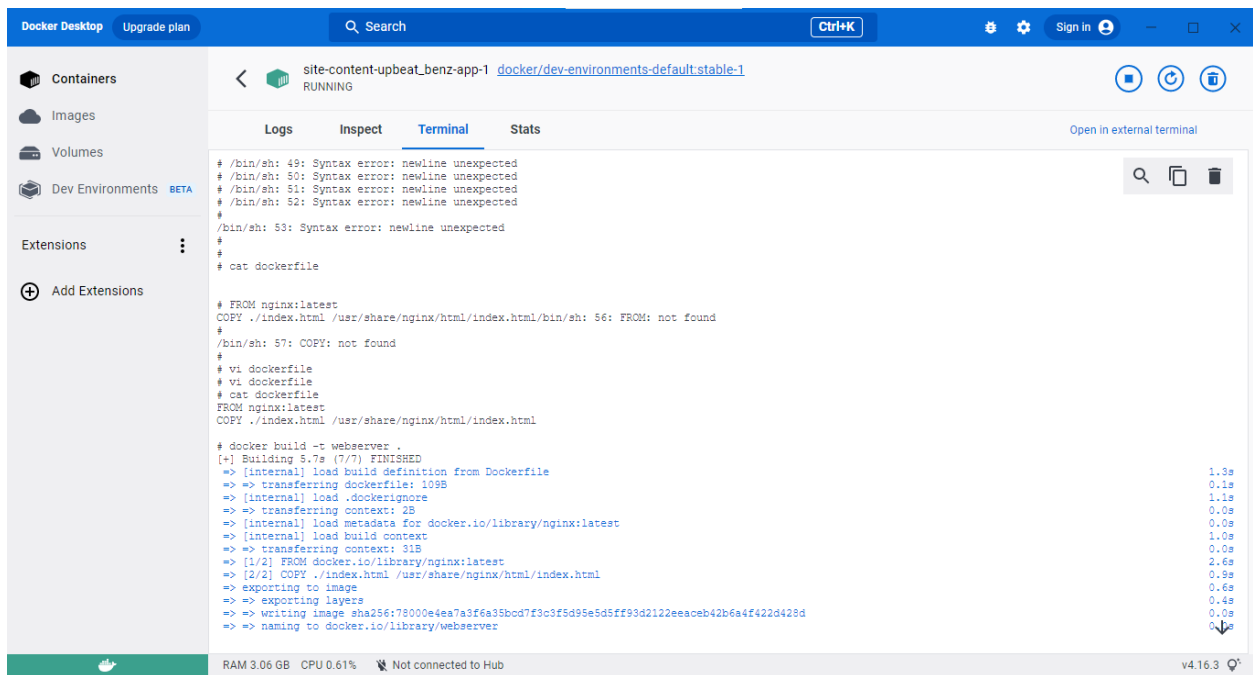


Build Custom NGINX Image

1. Creating a Dev Environment in the local directory and pulling the latest nginx image.

3. Building a webserver from using the index.html created earlier

docker build -t webserver .



The screenshot shows the Docker Desktop interface. On the left, there's a sidebar with 'Containers', 'Images', 'Volumes', 'Dev Environments', and 'Extensions'. The main area displays the 'Terminal' tab for a container named 'site-content-upbeat_benz-app-1'. The terminal output shows the execution of 'cat dockerfile' and 'docker build -t webserver .'. The build process is successful, showing progress bars for various steps like 'load build definition from Dockerfile', 'transferring dockerfile', 'load .dockerignore', 'load metadata for docker.io/library/nginx:latest', 'transferring context', 'FROM docker.io/library/nginx:latest', 'COPY ./index.html /usr/share/nginx/html/index.html', 'exporting to image', 'writing image sha256:78000e4ea7a3f6a35bod7f3c3f5d95e5d5ff93d2122eeaceb42b6a4f422d428d', and 'naming to docker.io/library/webserver'. The status bar at the bottom indicates 'RAM 3.06 GB', 'CPU 0.61%', and 'Not connected to Hub'.

```
# /bin/sh: 49: Syntax error: newline unexpected
# /bin/sh: 50: Syntax error: newline unexpected
# /bin/sh: 51: Syntax error: newline unexpected
# /bin/sh: 52: Syntax error: newline unexpected
#
/bin/sh: 53: Syntax error: newline unexpected
#
# cat dockerfile

# FROM nginx:latest
COPY ./index.html /usr/share/nginx/html/index.html/bin/sh: 56: FROM: not found
#
/bin/sh: 57: COPY: not found
#
# vi dockerfile
# vi dockerfile
# cat dockerfile
FROM nginx:latest
COPY ./index.html /usr/share/nginx/html/index.html

# docker build -t webserver .
[+] Building 5.7s (7/7) FINISHED
=> [internal] load build definition from Dockerfile 1.3s
=> => transferring dockerfile: 109B 0.1s
=> [internal] load .dockerignore 1.1s
=> => transferring context: 2B 0.0s
=> [internal] load metadata for docker.io/library/nginx:latest 1.0s
=> [internal] load build context 0.0s
=> => transferring context: 31B 0.0s
=> [1/2] FROM docker.io/library/nginx:latest 2.6s
=> [2/2] COPY ./index.html /usr/share/nginx/html/index.html 0.3s
=> exporting to image 0.6s
=> exporting layers 0.4s
=> writing image sha256:78000e4ea7a3f6a35bod7f3c3f5d95e5d5ff93d2122eeaceb42b6a4f422d428d 0.0s
=> naming to docker.io/library/webserver 0.0s
```



Questions:

1. What is Docker file?

A Docker Image is a read-only file with a bunch of instructions. When these instructions are executed, it creates a Docker container. Docker file is a simple text file that consists of instructions to build Docker images.

Outcomes:

CO2: Study the Evolution of Cloud Computing and its models.

Conclusion: (Conclusion to be based on the Results and outcomes achieved)

We created a Docker hub account and learnt the different commands used in Docker. We pulled official images from Docker hub, ran containers, modified images and pushed it to our Docker hub account's repository.