

Lab 8: Docker

Ahmed Baha Eddine Alimi

Assignment Report

SD-01

I. Questions to Answer :

1. Compare and contrast **ENTRYPOINT** and **CMD** in Dockerfile. In what situation would you use each of them?

ENTRYPOINT is used to define the main command that always runs when the container starts. It is preferred when you want to enforce a specific behavior, like running a script or application.

CMD provides default arguments to **ENTRYPOINT** or specifies a command to run if no arguments are supplied.

We use **ENTRYPOINT** when you need the container to run as an executable with mandatory behavior.

We use **CMD** for default behavior that can be overridden at runtime.

2. List five security precautions you will take when building or deploying a Docker resource (image or container).

1. Use minimal base images (like alpine or distroless) to reduce attack surface.
2. Do not run the container as root; create a dedicated user.
3. Enable Docker Content Trust (DCT) to verify image authenticity.
4. Regularly update and scan images for vulnerabilities using tools like trivy.
5. Restrict container privileges (like `--cap-drop=ALL` and `--read-only filesystem`).

3. Show a single line command that will remove all exited Docker containers. Do not use any text filtering editor. Show test results.

`docker container prune -f`: This command removes all stopped (exited) containers without prompting for confirmation (`-f` flag).

```
allimi@Ubuntu:~$ docker run --name test1 alpine echo "Hello"
Unable to find image 'alpine:latest' locally
latest: Pulling from library/alpine
f18232174bc9: Pull complete
Digest: sha256:a8560b36e8b8210634f77d9f7f9efd7ffa463e380b75e2e74aff4511df3ef88c
Status: Downloaded newer image for alpine:latest
Hello
allimi@Ubuntu:~$ docker run --name test2 alpine echo "World"
World
allimi@Ubuntu:~$
```

```
allimi@Ubuntu:~$ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
f375d5acbf59	alpine	"echo World"	41 seconds ago	Exited (0) 40 seconds ago		test2
2df437d31e18	alpine	"echo Hello"	49 seconds ago	Exited (0) 47 seconds ago		test1

```
allimi@Ubuntu:~$
```

```
allimi@Ubuntu:~$ docker container prune -f
Deleted Containers:
f375d5acbf59b39969c5353c13702ab5d0ed1e87dd3fe7da8adc40ee5a227cc9
2df437d31e18f0ae546c8a10aba57d2df1c476ff58ce16e53be34e83fa91c01a

Total reclaimed space: 0B
allimi@Ubuntu:~$
```

```
allimi@Ubuntu:~$ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

```
allimi@Ubuntu:~$
```

```
allimi@Ubuntu:~$ docker run --name test1 alpine echo "Hello"
Unable to find image 'alpine:latest' locally
latest: Pulling from library/alpine
f18232174bc9: Pull complete
Digest: sha256:a8560b36e8b8210634f77d9f7f9efd7ffa463e380b75e2e74aff4511df3ef88c
Status: Downloaded newer image for alpine:latest
Hello
allimi@Ubuntu:~$ docker run --name test2 alpine echo "World"
World
allimi@Ubuntu:~$ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
f375d5acbf59	alpine	"echo World"	41 seconds ago	Exited (0) 40 seconds ago		test2
2df437d31e18	alpine	"echo Hello"	49 seconds ago	Exited (0) 47 seconds ago		test1

```
allimi@Ubuntu:~$ docker container prune -f
Deleted Containers:
f375d5acbf59b39969c5353c13702ab5d0ed1e87dd3fe7da8adc40ee5a227cc9
2df437d31e18f0ae546c8a10aba57d2df1c476ff58ce16e53be34e83fa91c01a

Total reclaimed space: 0B
allimi@Ubuntu:~$ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

```
allimi@Ubuntu:~$
```

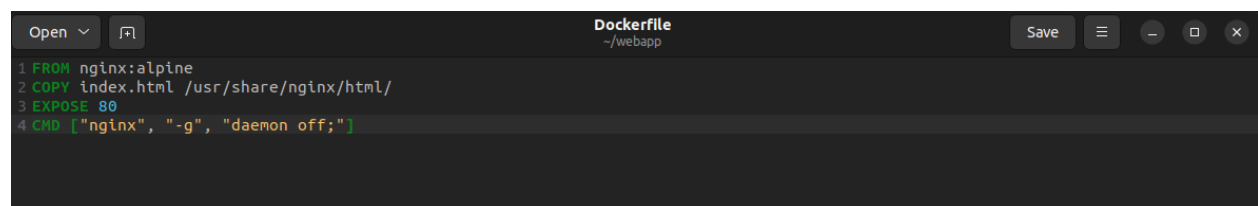
4. Show how you can copy files to a running container without entering the container's interactive shell.

```
allimi@Ubuntu:~/Desktop$ docker run -d --name test-container alpine sleep 1000
2bf10c2573dbc899633e696a2bda6b081bca465cbff5081ac481f1819f81660
allimi@Ubuntu:~/Desktop$ echo "Hello from host" > file.txt
allimi@Ubuntu:~/Desktop$ docker cp file.txt test-container:/tmp/
Successfully copied 2.05kB to test-container:/tmp/
allimi@Ubuntu:~/Desktop$ docker exec test-container cat /tmp/file.txt
Hello from host
allimi@Ubuntu:~/Desktop$
```

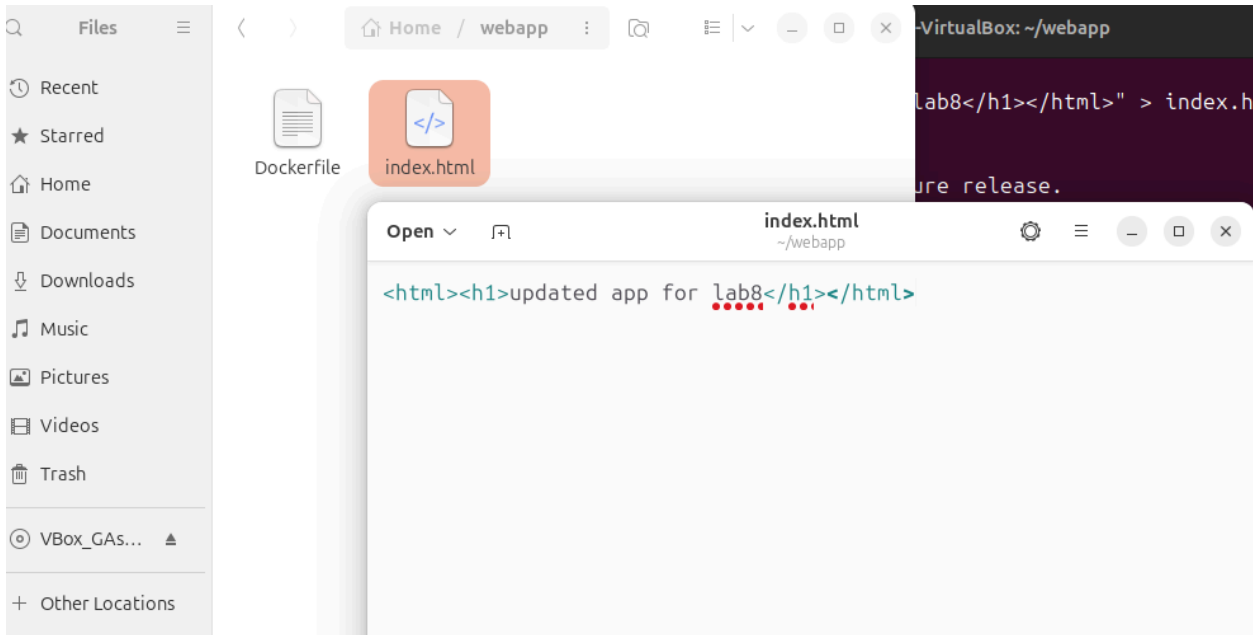
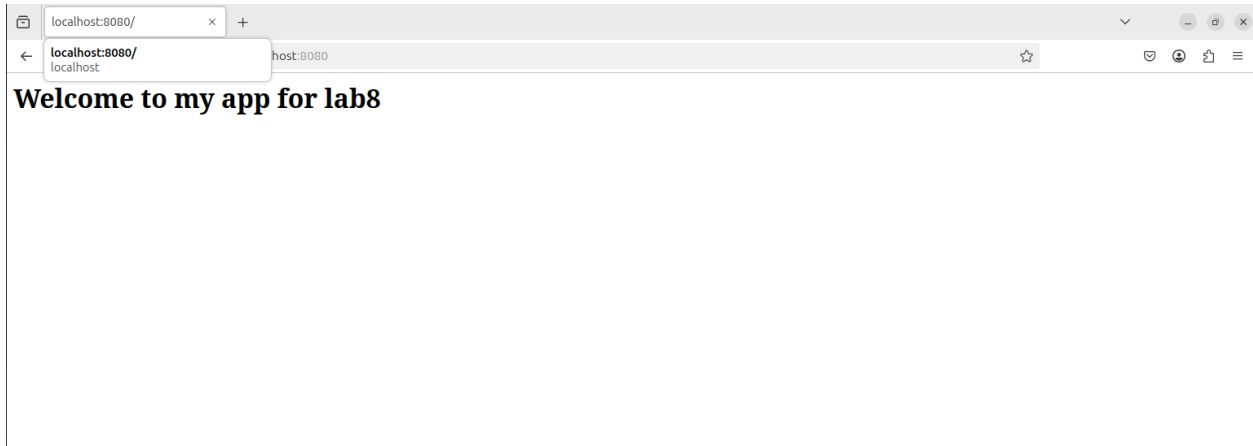
5. Create a dockerized web application running on nginx. The web index page index.html should be located on your host machine. The directory containing the index page should be mounted to the container and served from there.

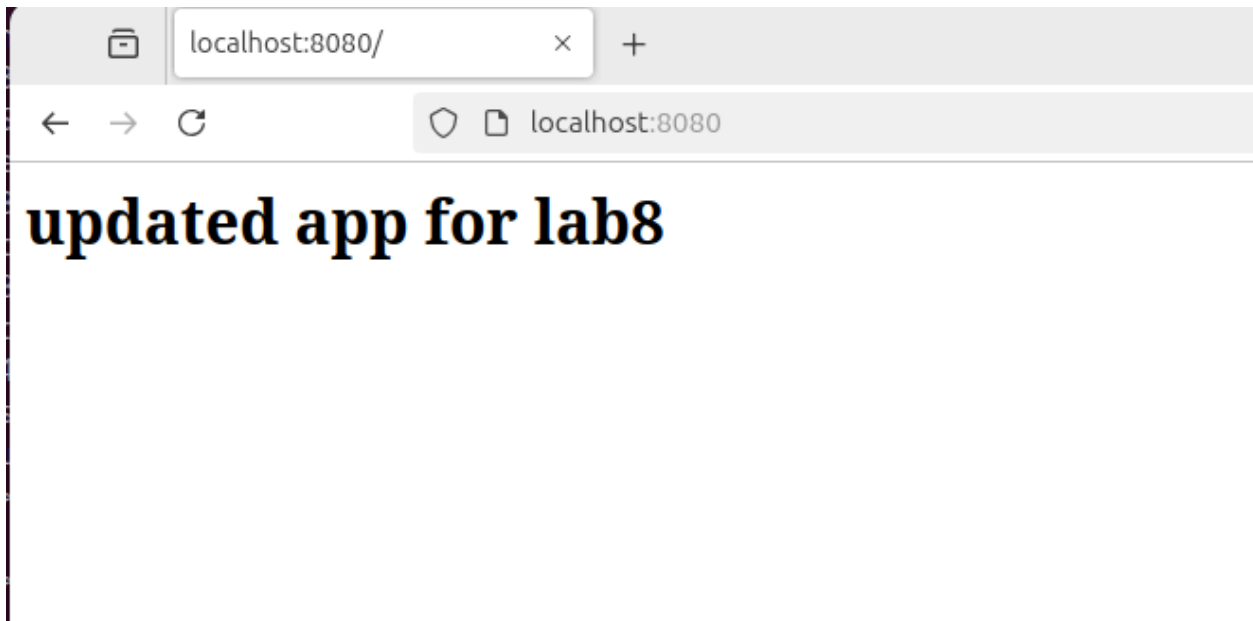
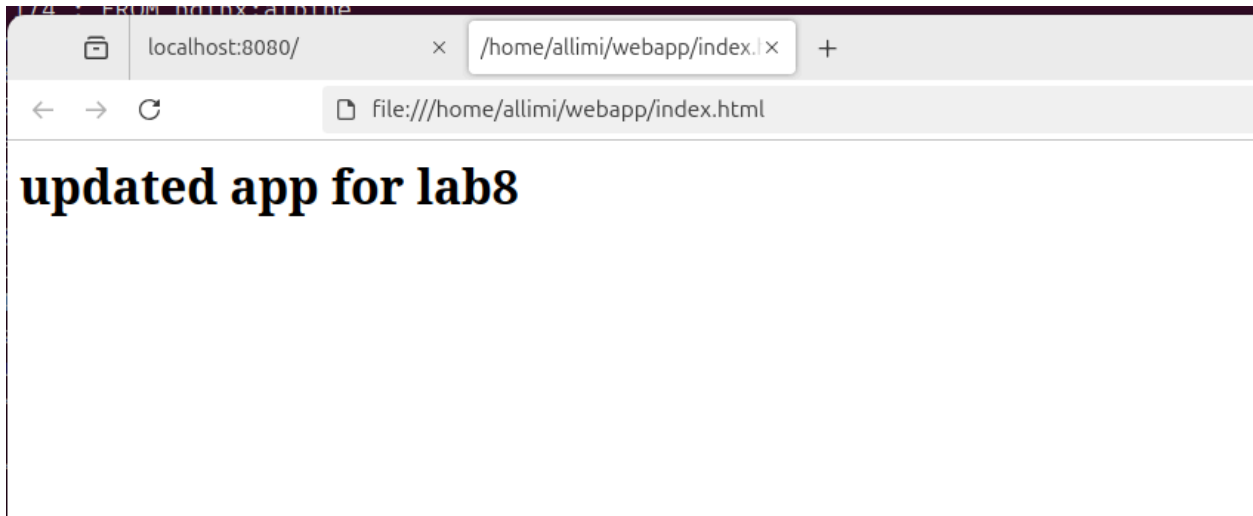
```
allimi@allimi-VirtualBox:~/Desktop$ mkdir ~/webapp
allimi@allimi-VirtualBox:~/webapp$ echo "<html><h1>Welcome to my app for lab8</h1></html>" > index.html
allimi@allimi-VirtualBox:~/webapp$ touch Dockerfile
allimi@allimi-VirtualBox:~/webapp$ docker build -t my-webapp .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
             Install the buildx component to build images with BuildKit:
             https://docs.docker.com/go/buildx/

Sending build context to Docker daemon  3.072kB
Step 1/4 : FROM nginx:alpine
alpine: Pulling from library/nginx
f18232174bc9: Pull complete
ccc35e35d420: Pull complete
43f2ec460bdf: Pull complete
984583bcf083: Pull complete
8d27c072a58f: Pull complete
ab3286a73463: Pull complete
6d79cc6084d4: Pull complete
0c7e4c092ab7: Pull complete
Digest: sha256:4ff102c5d78d254a6f0da062b3cf39eaf07f01eec0927fd21e219d0af8bc0591
Status: Downloaded newer image for nginx:alpine
--> 1ff4bb4faebc
Step 2/4 : COPY index.html /usr/share/nginx/html/
--> f62820e093ea
Step 3/4 : EXPOSE 80
--> Running in 6c0e38a04525
--> Removed intermediate container 6c0e38a04525
--> 18b715c0aefd
Step 4/4 : CMD ["nginx", "-g", "daemon off;"]
--> Running in 675138c006e9
--> Removed intermediate container 675138c006e9
--> 7b95ec22242f
Successfully built 7b95ec22242f
Successfully tagged my-webapp:latest
allimi@allimi-VirtualBox:~/webapp$ docker run -d -p 8080:80 -v $(pwd):/usr/share/nginx/html --name webapp my-webapp
703272e0a18ff70ecf0f06796564b3c701364f584224bcb9f08f008ecd7fb7d4
```



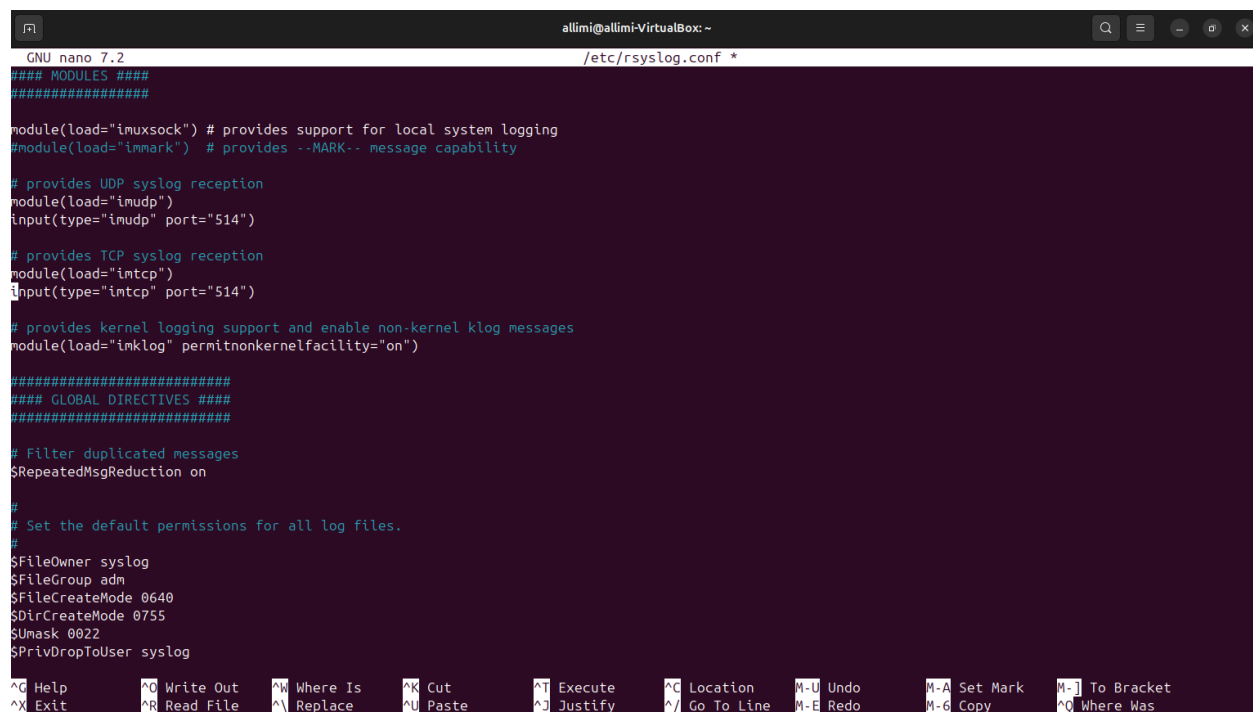
```
Open  Dockerfile  Save  ~/webapp
1 FROM nginx:alpine
2 COPY index.html /usr/share/nginx/html/
3 EXPOSE 80
4 CMD ["nginx", "-g", "daemon off;"]
```





6. Setup rsyslog on your host machine as a central logging server.
Create a Docker container and configure it to forward its log to your central logging server.

```
allimi@allimi-VirtualBox:~$ sudo apt-get install rsyslog
[sudo] password for allimi:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
rsyslog is already the newest version (8.2312.0-3ubuntu9).
rsyslog set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 68 not upgraded.
allimi@allimi-VirtualBox:~$ sudo nano /etc/rsyslog.conf
```



```
allimi@allimi-VirtualBox: ~
GNU nano 7.2 /etc/rsyslog.conf *
##### MODULES #####
#####
module(load="imuxsock") # provides support for local system logging
#module(load="imark") # provides --MARK-- message capability

# provides UDP syslog reception
module(load="imudp")
input(type="imudp" port="514")

# provides TCP syslog reception
module(load="intcp")
input(type="intcp" port="514")

# provides kernel logging support and enable non-kernel klog messages
module(load="inklog" permitnonkernelfacility="on")

##### GLOBAL DIRECTIVES #####
#####
# Filter duplicated messages
$RepeatedMsgReduction on

#
# Set the default permissions for all log files.
#
$FileOwner syslog
$FileGroup adm
$FileCreateMode 0640
$DirCreateMode 0755
$Umask 0022
$PrivDropToUser syslog

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^Q Location   M-U Undo      M-A Set Mark   M-J To Bracket
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^_ Justify    ^/ Go To Line  M-E Redo      M-G Copy      ^C Where Was
```

```
allimi@allimi-VirtualBox:~$ sudo systemctl restart rsyslog
allimi@allimi-VirtualBox:~$
```

```
allini@allini-virtualbox: /Desktop$ docker run -d --name log-container --log-driver=syslog --log-opt syslog-address=udp://172.17.0.1:514 alpine sh -c "echo 'Test log message'; sleep 10"
f791fb149cd671616e3d46014f69a140d4a389e1f8822c6c62b70177c884c666

allini@allini-virtualbox: /Desktop$ docker run -d --name log-container --log-driver=syslog --log-opt syslog-address=udp://172.17.0.1:514 alpine sh -c "echo 'Test log message'; sleep 10"
f791fb149cd671616e3d46014f69a140d4a389e1f8822c6c62b70177c884c666
allini@allini-virtualbox: /Desktop$ sudo tail -F /var/log/syslog
[sudo] password for allini:
2025-03-17T23:36:53.127566+03:00 allini-virtualbox containerd[1303]: time="2025-03-17T23:36:53.123928156+03:00" level=info msg="loading plugin \"io.containerd.trtpr.v1.pause\"..." runtime=io.containerd.runc.v2 type=io.containerd.trtpr.v1
2025-03-17T23:36:53.204717+03:00 allini-virtualbox systemd[1]: Started docker-f791fb149cd671616e3d46014f69a140d4a389e1f8822c6c62b70177c884c666.scope - libcontainer container f791fb149cd671616e3d46014f69a140d4a389e1f8822c6c62b70177c884c666.
2025-03-17T23:36:53.384664+03:00 allini-virtualbox kernel: eth0: renamed from veth73aa307
2025-03-17T23:36:53.386525+03:00 allini-virtualbox kernel: docker0: port 2(veth2bda19e) entered blocking state
2025-03-17T23:36:53.386538+03:00 allini-virtualbox kernel: docker0: port 2(veth2bda19e) entered forwarding state
2025-03-17T23:36:53.388920+03:00 allini-virtualbox NetworkManager[972]: <info> [1742243813.3885] device (veth2bda19e): carrier: link connected
2025-03-17T23:36:53+03:00 allini-virtualbox f791fb149cd6[1481]: Test log message
2025-03-17T23:36:55.208894+03:00 allini-virtualbox avahi-daemon[917]: Joining mDNS multicast group on interface veth2bda19e.IPv6 with address fe80::2c7b:5aff:fe8c:43d1.
2025-03-17T23:36:55.209366+03:00 allini-virtualbox avahi-daemon[917]: New relevant interface veth2bda19e.IPv6 for mDNS.
2025-03-17T23:36:55.209633+03:00 allini-virtualbox avahi-daemon[917]: Registering new address record for fe80::2c7b:5aff:fe8c:43d1 on veth2bda19e.*.
2025-03-17T23:37:03.473820+03:00 allini-virtualbox systemd[1]: docker-f791fb149cd671616e3d46014f69a140d4a389e1f8822c6c62b70177c884c666.scope: Deactivated successfully.
2025-03-17T23:37:03.737465+03:00 allini-virtualbox containerd[1303]: time="2025-03-17T23:37:03.733513391+03:00" level=info msg="shim disconnected" id=f791fb149cd671616e3d46014f69a140d4a389e1f8822c6c62b70177c884c666 namespace=moby
2025-03-17T23:37:03.737520+03:00 allini-virtualbox containerd[1303]: time="2025-03-17T23:37:03.733566810+03:00" level=warning msg="cleaning up after shim disconnected" id=f791fb149cd671616e3d46014f69a140d4a389e1f8822c6c62b70177c884c666 namespace=moby
2025-03-17T23:37:03.737531+03:00 allini-virtualbox containerd[1303]: time="2025-03-17T23:37:03.733572471+03:00" level=info msg="cleaning up dead shim" namespace=moby
2025-03-17T23:37:03.772434+03:00 allini-virtualbox dockerd[1401]: time="2025-03-17T23:37:03.747891381+03:00" level=info msg="ignoring event" container=f791fb149cd671616e3d46014f69a140d4a389e1f8822c6c62b70177c884c666 module=libcontainerd namespace=moby topics/tasks/delete type="events.TaskDelete"
2025-03-17T23:37:03.949377+03:00 allini-virtualbox kernel: docker0: port 2(veth2bda19e) entered disabled state
2025-03-17T23:37:03.949486+03:00 allini-virtualbox kernel: veth73aa307: renamed from eth0
2025-03-17T23:37:03.986545+03:00 allini-virtualbox NetworkManager[972]: <info> [1742243823.9861] manager: (veth73aa307): new Veth device (/org/freedesktop/NetworkManager/Devices/17)
2025-03-17T23:37:04.004598+03:00 allini-virtualbox avahi-daemon[917]: Interface veth2bda19e.IPv6 no longer relevant for mDNS.
2025-03-17T23:37:04.005741+03:00 allini-virtualbox avahi-daemon[917]: Leaving mDNS multicast group on interface veth2bda19e.IPv6 with address fe80::2c7b:5aff:fe8c:43d1.
2025-03-17T23:37:04.006620+03:00 allini-virtualbox kernel: docker0: port 2(veth2bda19e) entered disabled state
2025-03-17T23:37:04.013443+03:00 allini-virtualbox kernel: veth2bda19e (unregistering): left allmulticast mode
2025-03-17T23:37:04.029173+03:00 allini-virtualbox kernel: veth2bda19e (unregistering): left promiscuous mode
2025-03-17T23:37:04.029194+03:00 allini-virtualbox kernel: docker0: port 2(veth2bda19e) entered disabled state
2025-03-17T23:37:04.062698+03:00 allini-virtualbox avahi-daemon[917]: Withdrawing address record for fe80::2c7b:5aff:fe8c:43d1 on veth2bda19e.
2025-03-17T23:37:04.110625+03:00 allini-virtualbox systemd[1]: run-docker-netns-d1269d4c4160.mount: Deactivated successfully.
2025-03-17T23:37:04.127979+03:00 allini-virtualbox systemd[1]: var-lib-docker-overlay2-915d0aad4c88010e0f35e905b62b27833c852f18a4cb4f51a828ce40c0be952b-merged.mount: Deactivated successfully
^
```

Bonus:

7. Dockerize any open source application of your choice, and host it on Docker hub. Share link to the repository.

(NOT ANSWERED)

8. Find and fix the problems in the following Dockerfile. There are some issues building the image and also running the container:

```
FROM alpine
```

```
RUN apt-get update && apt-get install -y python3 --no-install-recommends
```

```
RUN touch index.html
```

```
RUN echo "<html><h1>Testing web</h1></html>" >> index.html
```

```
CMD ["python", "-m", "http.server"]
```

Issues in the Dockerfile

- apt-get is not available in Alpine Linux: Alpine Linux uses the apk package manager instead of apt-get.
- Redirection issue with >>: use only one >

- **Python is not installed by default:**The python command is not available in Alpine Linux. You need to install python3.
- **Incorrect CMD:**The CMD uses python, but python3 should be used instead.
- **Inefficient use of RUN commands:**Multiple RUN commands can be combined to reduce the number of layers in the Docker image.

```
allimi@allimi-VirtualBox: ~/docker-task8
allimi@allimi-VirtualBox:~/Desktop$ mkdir ~/docker-task8
allimi@allimi-VirtualBox:~/Desktop$ cd ~/docker-task8
allimi@allimi-VirtualBox:~/docker-task8$ touch Dockerfile
allimi@allimi-VirtualBox:~/docker-task8$
```

Corrected File:

```
Open ▾ Dockerfile
~/docker-task8

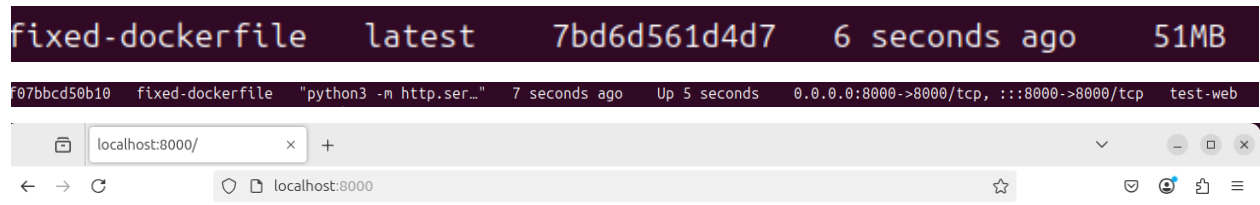
FROM alpine
RUN apk update && apk add --no-cache python3
RUN echo "<html><h1>Testing web</h1></html>" > index.html
CMD ["python3", "-m", "http.server"]
```

```
allimi@allimi-VirtualBox:~/docker-task8$ docker build -t fixed-dockerfile .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
            Install the buildx component to build images with BuildKit:
            https://docs.docker.com/go/buildx/
```

```
Sending build context to Docker daemon  2.048kB
Step 1/4 : FROM alpine
--> aded1e1a5b37
Step 2/4 : RUN apk update && apk add --no-cache python3
--> Running in 47283accb3b6
fetch https://dl-cdn.alpinelinux.org/alpine/v3.21/main/x86_64/APKINDEX.tar.gz
fetch https://dl-cdn.alpinelinux.org/alpine/v3.21/community/x86_64/APKINDEX.tar.gz
v3.21.3-182-gb7f398fb35c [https://dl-cdn.alpinelinux.org/alpine/v3.21/main]
v3.21.3-183-g40d2ae30bee [https://dl-cdn.alpinelinux.org/alpine/v3.21/community]
OK: 25395 distinct packages available
fetch https://dl-cdn.alpinelinux.org/alpine/v3.21/main/x86_64/APKINDEX.tar.gz
fetch https://dl-cdn.alpinelinux.org/alpine/v3.21/community/x86_64/APKINDEX.tar.gz
(1/17) Installing libbz2 (1.0.8-r6)
(2/17) Installing libexpat (2.7.0-r0)
(3/17) Installing libffi (3.4.7-r0)
(4/17) Installing gdbm (1.24-r0)
(5/17) Installing xz-libs (5.6.3-r0)
(6/17) Installing libgcc (14.2.0-r4)
(7/17) Installing libstdc++ (14.2.0-r4)
(8/17) Installing mpdecimal (4.0.0-r0)
(9/17) Installing ncurses-terminfo-base (6.5_p20241006-r3)
(10/17) Installing libncursesw (6.5_p20241006-r3)
(11/17) Installing libpanelw (6.5_p20241006-r3)
(12/17) Installing readline (8.2.13-r0)
(13/17) Installing sqlite-libs (3.48.0-r0)
(14/17) Installing python3 (3.12.9-r0)
(15/17) Installing python3-pycache-pyc0 (3.12.9-r0)
(16/17) Installing pyc (3.12.9-r0)
(17/17) Installing python3-pyc (3.12.9-r0)
```

```
(8/17) Installing mpdecimal (4.0.0-r0)
(9/17) Installing ncurses-terminfo-base (6.5_p20241006-r3)
(10/17) Installing libncursesw (6.5_p20241006-r3)
(11/17) Installing libpanelw (6.5_p20241006-r3)
(12/17) Installing readline (8.2.13-r0)
(13/17) Installing sqlite-libs (3.48.0-r0)
(14/17) Installing python3 (3.12.9-r0)
(15/17) Installing python3-pycache-pyc0 (3.12.9-r0)
(16/17) Installing pyc (3.12.9-r0)
(17/17) Installing python3-pyc (3.12.9-r0)
Executing busybox-1.37.0-r12.trigger
OK: 46 MiB in 32 packages
--> Removed intermediate container 47283accb3b6
--> 6922a87d23e7
Step 3/4 : RUN echo "<html><h1>Testing web</h1></html>" > index.html
--> Running in 18faaf525226
--> Removed intermediate container 18faaf525226
--> a494941576b3
Step 4/4 : CMD ["python3", "-m", "http.server"]
--> Running in 855af6bb94ef
--> Removed intermediate container 855af6bb94ef
--> 7bd6d561d4d7
Successfully built 7bd6d561d4d7
Successfully tagged fixed-dockerfile:latest
allimi@allimi-VirtualBox:~/docker-task8$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
fixed-dockerfile    latest             7bd6d561d4d7       6 seconds ago      51MB
my-webapp           latest            7b95ec22242f       39 minutes ago     47.9MB
alpine              latest            aded1e1a5b37       4 weeks ago        7.83MB
nginx              alpine            1ff4bb4faebc       5 weeks ago        47.9MB
allimi@allimi-VirtualBox:~/docker-task8$ docker run -d -p 8000:8000 --name test-web fixed-dockerfile
f07bbcd50b1084b47425489ef66d78a44a78969b7239ecfa58655323cc4d80c1
allimi@allimi-VirtualBox:~/docker-task8$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES
f07bbcd50b10   fixed-dockerfile  "python3 -m http.ser..." 7 seconds ago  Up 5 seconds  0.0.0.0:8000->8000/tcp, :::8000->8000/tcp  test-web
703272e0a18f   my-webapp       "/docker-entrypoint...." 39 minutes ago  Up 39 minutes  0.0.0.0:8080->80/tcp, :::8080->80/tcp  webapp
allimi@allimi-VirtualBox:~/docker-task8$
```

The image was built successfully



Testing web