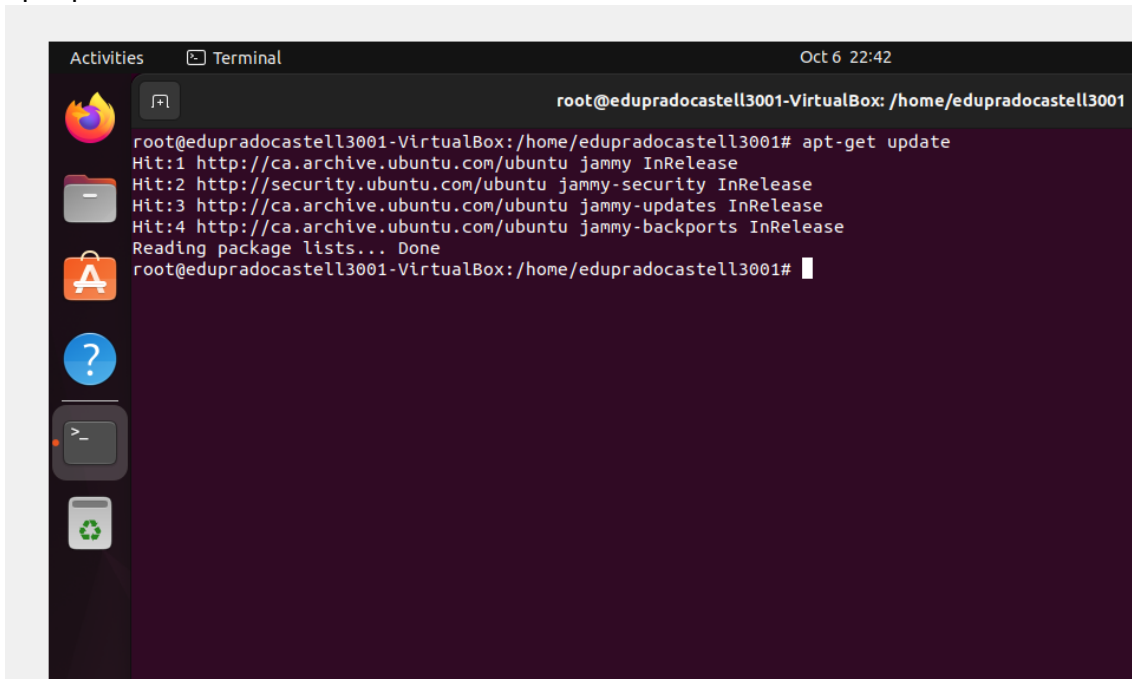


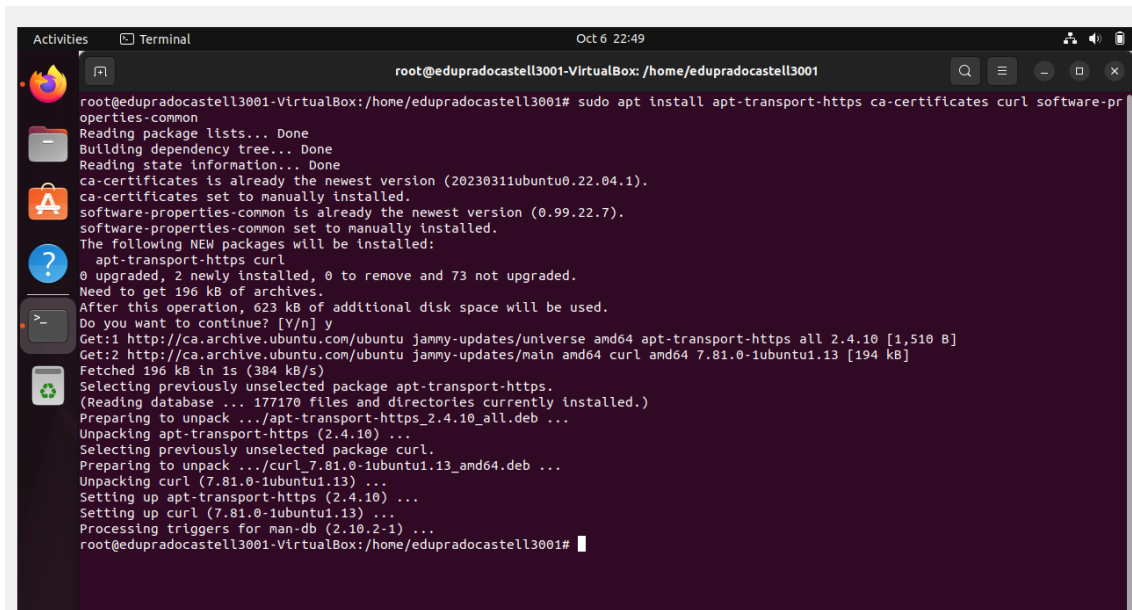
First, update your existing list of packages:

`apt update`



Next, install a few prerequisite packages which let apt use packages over HTTPS:

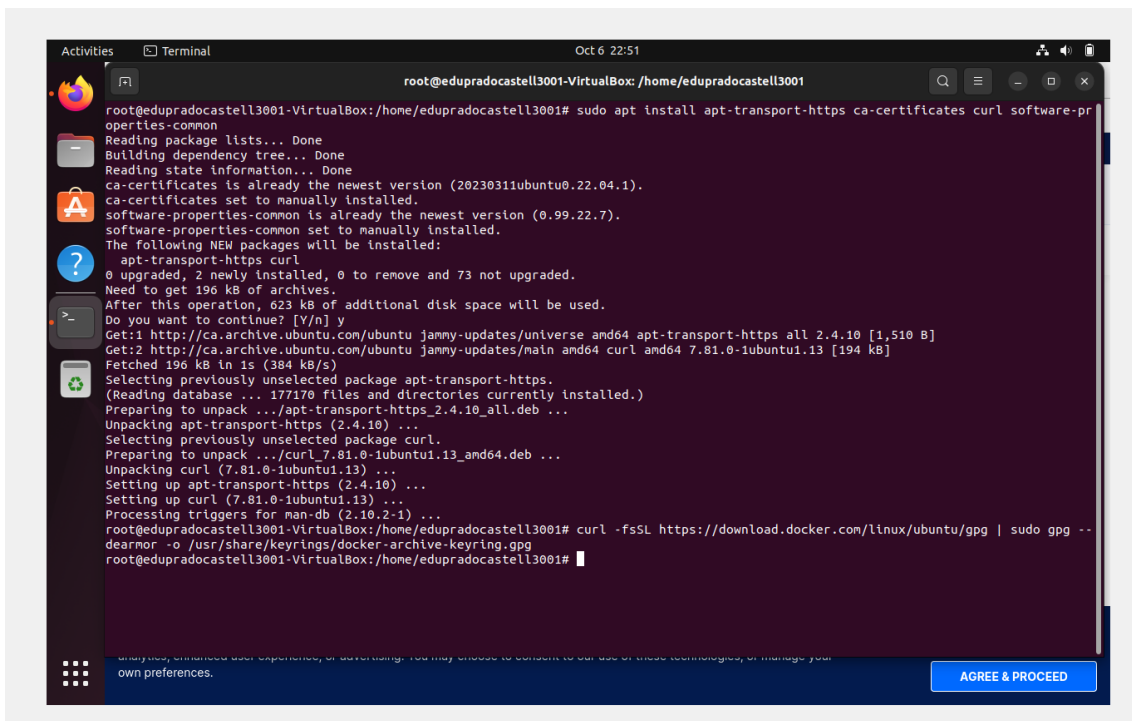
```
sudo apt install apt-transport-https ca-certificates curl software-properties-common
```



```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# sudo apt install apt-transport-https ca-certificates curl software-pr
operties-common
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20230311ubuntu0.22.04.1).
ca-certificates set to manually installed.
software-properties-common is already the newest version (0.99.22.7).
software-properties-common set to manually installed.
The following NEW packages will be installed:
  apt-transport-https curl
0 upgraded, 2 newly installed, 0 to remove and 73 not upgraded.
Need to get 196 kB of archives.
After this operation, 623 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ca.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 apt-transport-https all 2.4.10 [1,510 B]
Get:2 http://ca.archive.ubuntu.com/ubuntu jammy-updates/main amd64 curl amd64 7.81.0-1ubuntu1.13 [194 kB]
Fetched 196 kB in 1s (384 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 177170 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.4.10_all.deb ...
Unpacking apt-transport-https (2.4.10) ...
Selecting previously unselected package curl.
Preparing to unpack .../curl_7.81.0-1ubuntu1.13_amd64.deb ...
Unpacking curl (7.81.0-1ubuntu1.13) ...
Setting up apt-transport-https (2.4.10) ...
Setting up curl (7.81.0-1ubuntu1.13) ...
Processing triggers for man-db (2.10.2-1) ...
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001#
```

Then add the GPG key for the official Docker repository to your system:

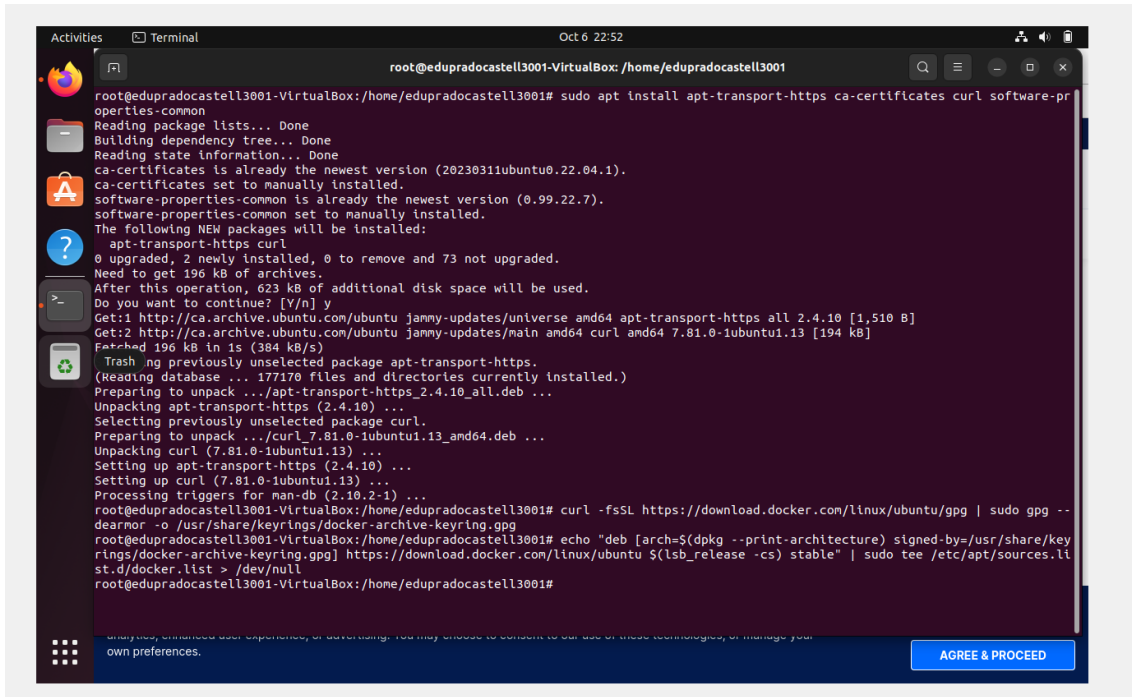
```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
```



```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# sudo apt install apt-transport-https ca-certificates curl software-pr
operties-common
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20230311ubuntu0.22.04.1).
ca-certificates set to manually installed.
software-properties-common is already the newest version (0.99.22.7).
software-properties-common set to manually installed.
The following NEW packages will be installed:
  apt-transport-https curl
0 upgraded, 2 newly installed, 0 to remove and 73 not upgraded.
Need to get 196 kB of archives.
After this operation, 623 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ca.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 apt-transport-https all 2.4.10 [1,510 B]
Get:2 http://ca.archive.ubuntu.com/ubuntu jammy-updates/main amd64 curl amd64 7.81.0-1ubuntu1.13 [194 kB]
Fetched 196 kB in 1s (384 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 177170 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.4.10_all.deb ...
Unpacking apt-transport-https (2.4.10) ...
Selecting previously unselected package curl.
Preparing to unpack .../curl_7.81.0-1ubuntu1.13_amd64.deb ...
Unpacking curl (7.81.0-1ubuntu1.13) ...
Setting up apt-transport-https (2.4.10) ...
Setting up curl (7.81.0-1ubuntu1.13) ...
Processing triggers for man-db (2.10.2-1) ...
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --
dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001#
```

Add the Docker repository to APT sources:

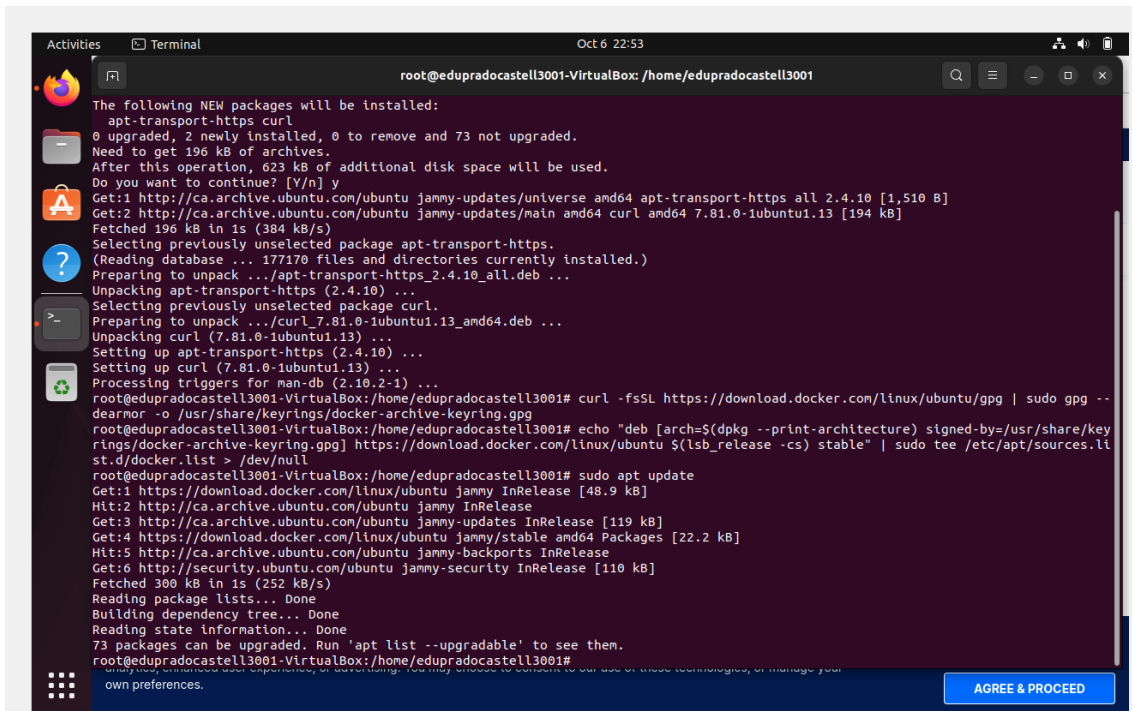
```
echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```



```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# 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 (20230311ubuntu0.22.04.1).
ca-certificates set to manually installed.
software-properties-common is already the newest version (0.99.22.7).
software-properties-common set to manually installed.
The following NEW packages will be installed:
  apt-transport-https curl
0 upgraded, 2 newly installed, 0 to remove and 73 not upgraded.
Need to get 196 kB of archives.
After this operation, 623 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ca.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 apt-transport-https all 2.4.10 [1,510 B]
Get:2 http://ca.archive.ubuntu.com/ubuntu jammy-updates/main amd64 curl amd64 7.81.0-1ubuntu1.13 [194 kB]
Fetched 196 kB in 1s (384 kB/s)
Preparing to unpack .../apt-transport-https_2.4.10_all.deb ...
Unpacking apt-transport-https (2.4.10) ...
Selecting previously unselected package curl.
Preparing to unpack .../curl_7.81.0-1ubuntu1.13_amd64.deb ...
Unpacking curl (7.81.0-1ubuntu1.13) ...
Setting up apt-transport-https (2.4.10) ...
Setting up curl (7.81.0-1ubuntu1.13) ...
Processing triggers for man-db (2.10.2-1) ...
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001#
```

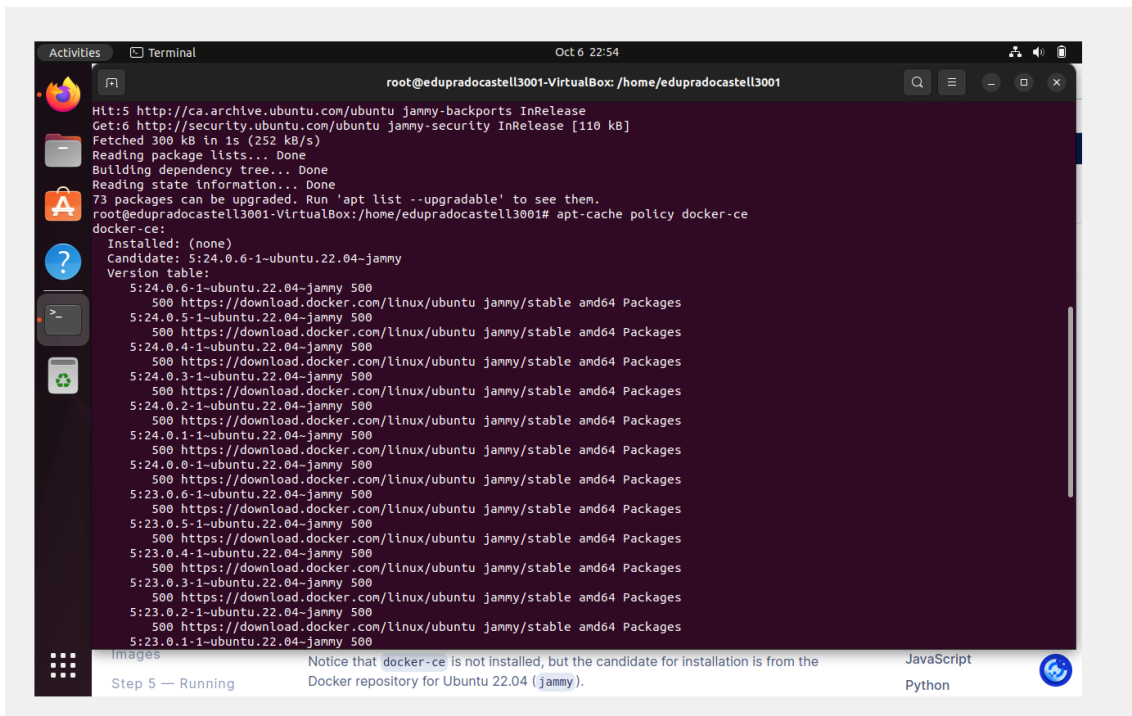
Update your existing list of packages again for the addition to be recognized:

```
sudo apt update
```



```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# sudo apt update
Get:1 https://download.docker.com/linux/ubuntu jammy InRelease [48.9 kB]
Hit:2 http://ca.archive.ubuntu.com/ubuntu jammy InRelease
Get:3 http://ca.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:4 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [22.2 kB]
Hit:5 http://ca.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:6 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Fetched 300 kB in 1s (252 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
73 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001#
```

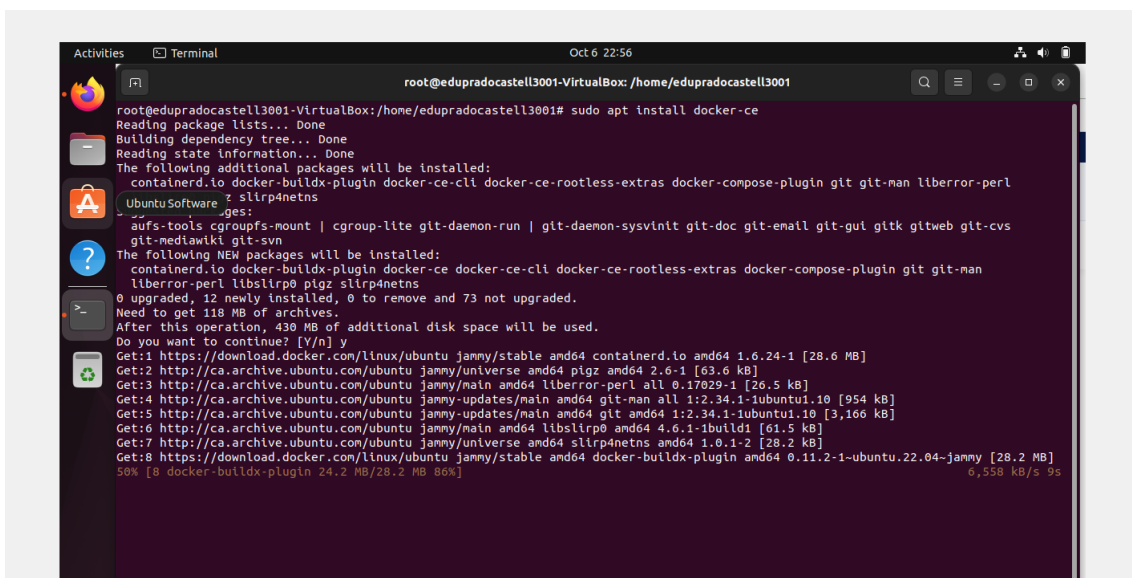
We can see output like this, although the version number for Docker may be different:



```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# apt-cache policy docker-ce
docker-ce:
  Installed: (none)
  Candidate: 5:24.0.6-1-ubuntu.22.04-jammy
  Version table:
   5:24.0.6-1-ubuntu.22.04-jammy 500
      500 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages
   5:24.0.5-1-ubuntu.22.04-jammy 500
      500 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages
   5:24.0.4-1-ubuntu.22.04-jammy 500
      500 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages
   5:24.0.3-1-ubuntu.22.04-jammy 500
      500 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages
   5:24.0.2-1-ubuntu.22.04-jammy 500
      500 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages
   5:24.0.1-1-ubuntu.22.04-jammy 500
      500 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages
   5:24.0.0-1-ubuntu.22.04-jammy 500
      500 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages
   5:23.0.6-1-ubuntu.22.04-jammy 500
      500 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages
   5:23.0.5-1-ubuntu.22.04-jammy 500
      500 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages
   5:23.0.4-1-ubuntu.22.04-jammy 500
      500 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages
   5:23.0.3-1-ubuntu.22.04-jammy 500
      500 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages
   5:23.0.2-1-ubuntu.22.04-jammy 500
      500 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages
   5:23.0.1-1-ubuntu.22.04-jammy 500
      500 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages
Images:
Notice that docker-ce is not installed, but the candidate for installation is from the
Docker repository for Ubuntu 22.04 (jammy).
Step 5 — Running JavaScript Python
```

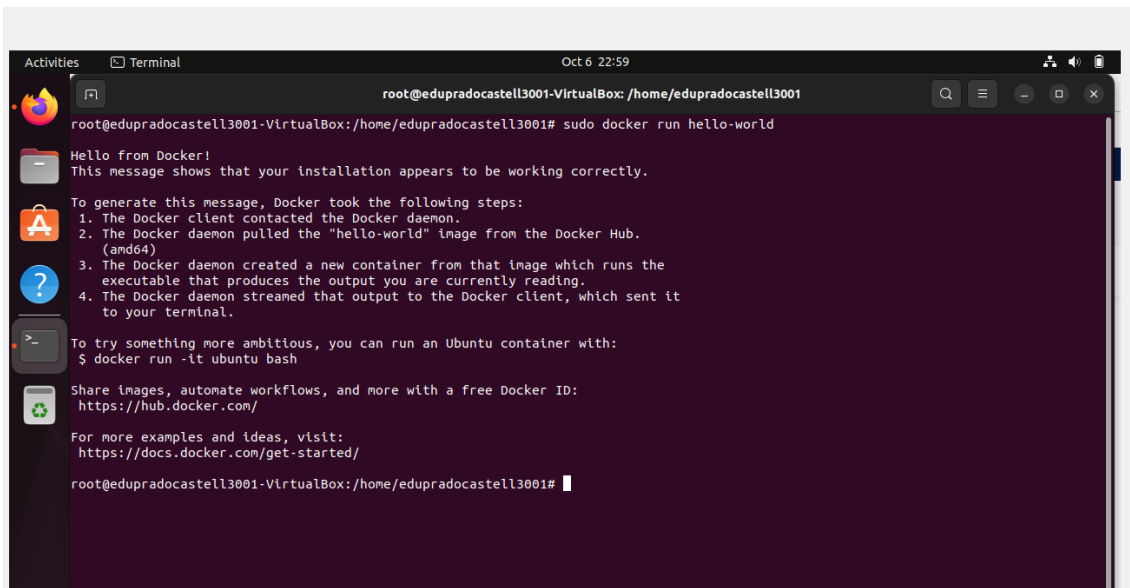
Finally, install Docker:

`sudo apt install docker-ce`



```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# 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 git git-man liberror-perl
  UbuntuSoftware ? slirp4netns
  aufs-tools cgroupfs-mount | cgroup-lite git-daemon-run | git-daemon-sysvint git-doc git-email git-gui gitk gitweb git-cvs
  git-mediawiki git-svn
The following NEW packages will be installed:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin git git-man
  liberror-perl libslirp0 pigz slirp4netns
0 upgraded, 12 newly installed, 0 to remove and 73 not upgraded.
Need to get 118 MB of archives.
After this operation, 430 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 https://download.docker.com/linux/ubuntu jammy/stable amd64 containerd.io amd64 1.6.24-1 [28.6 MB]
Get:2 http://ca.archive.ubuntu.com/ubuntu jammy/universe amd64 pigz amd64 2.6-1 [63.6 kB]
Get:3 http://ca.archive.ubuntu.com/ubuntu jammy/main amd64 liberror-perl all 0.17029-1 [26.5 kB]
Get:4 http://ca.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git-man all 1:2.34.1-1ubuntu1.10 [954 kB]
Get:5 http://ca.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git amd64 1:2.34.1-1ubuntu1.10 [3,166 kB]
Get:6 http://ca.archive.ubuntu.com/ubuntu jammy/main amd64 libslirp0 amd64 4.6.1-1build1 [61.5 kB]
Get:7 http://ca.archive.ubuntu.com/ubuntu jammy/universe amd64 slirp4netns amd64 1.0.1-2 [28.2 kB]
Get:8 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-buildx-plugin amd64 0.11.2-1-ubuntu.22.04-jammy [28.2 MB]
50% [8 docker-buildx-plugin 24.2 MB/28.2 MB 86%] 6,558 kB/s 9s
```

## Docker Run hello-world



A terminal window titled 'Terminal' with the date 'Oct 6 22:59'. The prompt is 'root@edupradocastell3001-VirtualBox: /home/edupradocastell3001'. The command 'sudo docker run hello-world' has been executed. The output is as follows:

```
root@edupradocastell3001-VirtualBox:/home/edupradocastell3001# sudo docker run hello-world
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

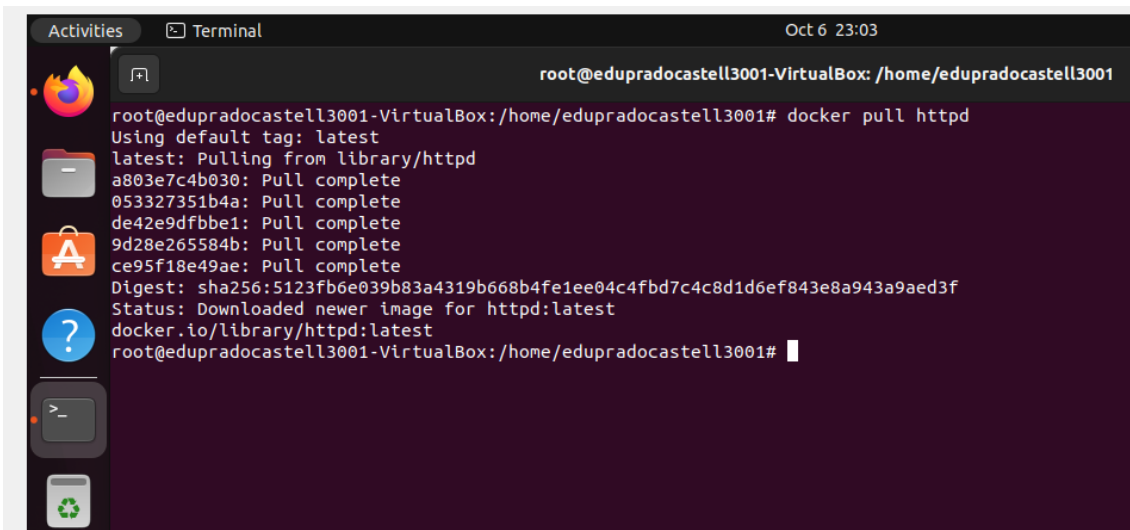
Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

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

root@edupradocastell3001-VirtualBox:/home/edupradocastell3001#
```

## Download the Apache Image for Docker

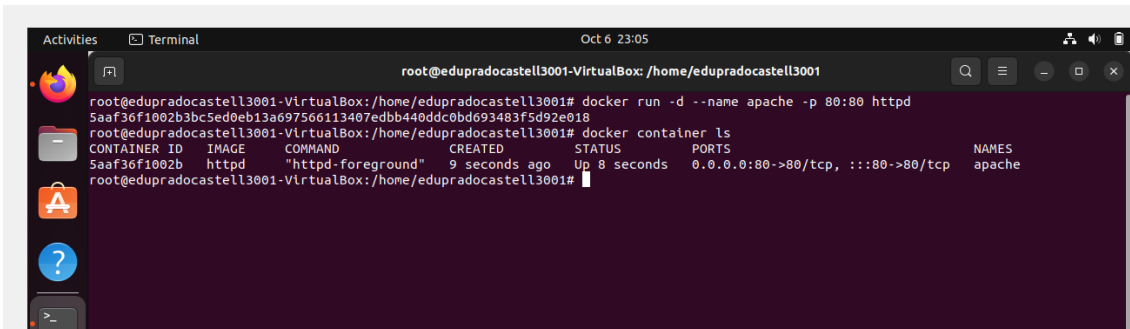
docker pull httpd



A terminal window titled 'Terminal' with the date 'Oct 6 23:03'. The prompt is 'root@edupradocastell3001-VirtualBox: /home/edupradocastell3001'. The command 'docker pull httpd' has been executed. The output is as follows:

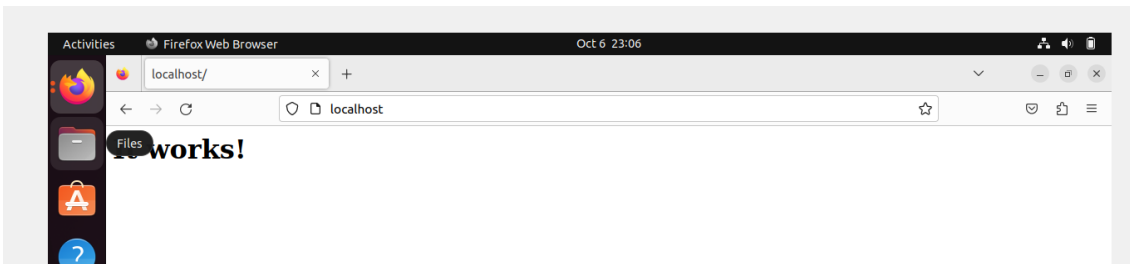
```
root@edupradocastell3001-VirtualBox:/home/edupradocastell3001# docker pull httpd
Using default tag: latest
latest: Pulling from library/httpd
a803e7c4b030: Pull complete
053327351b4a: Pull complete
de42e9dfbbe1: Pull complete
9d28e265584b: Pull complete
ce95f18e49ae: Pull complete
Digest: sha256:5123fb6e039b83a4319b668b4fe1ee04c4fbd7c4c8d1d6ef843e8a943a9aed3f
Status: Downloaded newer image for httpd:latest
docker.io/library/httpd:latest
root@edupradocastell3001-VirtualBox:/home/edupradocastell3001#
```

Docker run -d --name apache -p 80:80 httpd

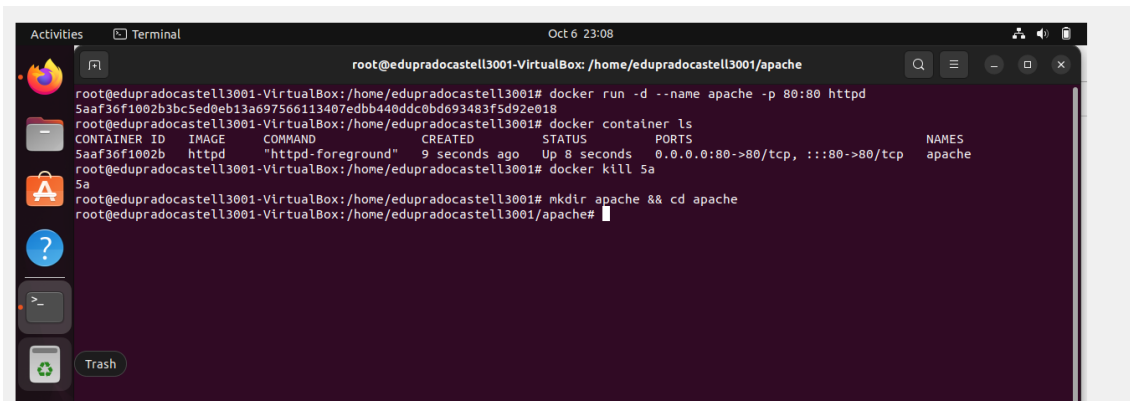


```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# docker run -d --name apache -p 80:80 httpd
5aaf36f1002b3bc5ed0eb13a697566113407edbb440ddc0bd693483f5d92e018
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# docker container ls
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES
5aaf36f1002b   httpd    "httpd-foreground"     9 seconds ago Up 8 seconds   0.0.0.0:80->80/tcp, :::80->80/tcp   apache
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001#
```

Its work!

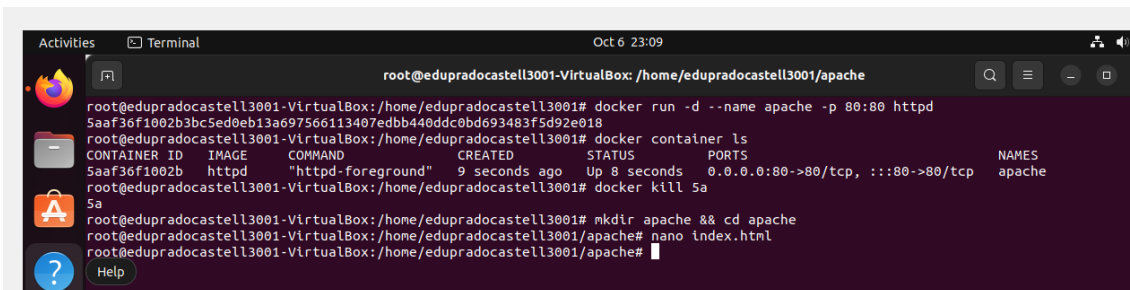


1. Create and go to the apache directory.  
mkdir apache && cd apache



```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker run -d --name apache -p 80:80 httpd
5aaf36f1002b3bc5ed0eb13a697566113407edbb440ddc0bd693483f5d92e018
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker container ls
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES
5aaf36f1002b   httpd    "httpd-foreground"     9 seconds ago Up 8 seconds   0.0.0.0:80->80/tcp, :::80->80/tcp   apache
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker kill 5a
5a
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# mkdir apache && cd apache
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache#
```

2. In the directory, create index.html  
This case we can use nano  
Nano index.html




```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker run -d --name apache -p 80:80 httpd
5aaf36f1002b3bc5ed0eb13a697566113407edbb440ddc0bd693483f5d92e018
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker container ls
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES
5aaf36f1002b   httpd    "httpd-foreground"     9 seconds ago Up 8 seconds   0.0.0.0:80->80/tcp, :::80->80/tcp   apache
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker kill 5a
5a
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# mkdir apache && cd apache
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# nano index.html
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache#
```

Here I code my webpage

My email econestoga: edopradocastell3001

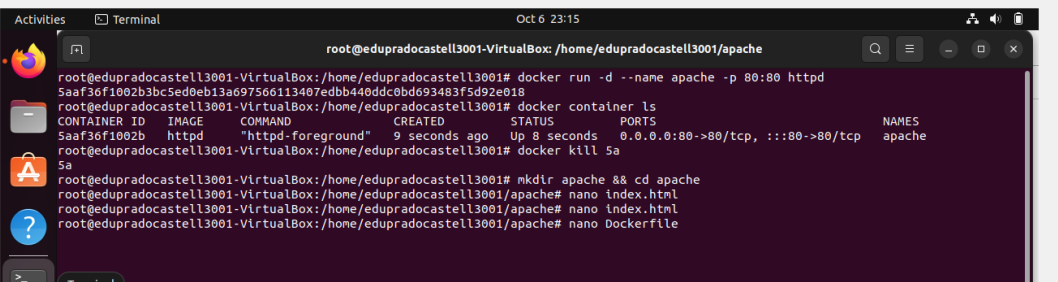
Finally save the file.



```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache
GNU nano 6.2 index.html *
<h1 style="color: #5e9ca0;">Welcome to - Edopradocastell3001 - WebPage!!</h1>
<p>Hello World!</p>
<p>&nbsp;</p>
Help Exit Write Out Where Is Cut Execute Location Undo Set Mark
Read File Replace Paste Justify Go To Line Redo Copy
This is a test page for the Apache deployment in Docker</p>
```

Next, build Docker file

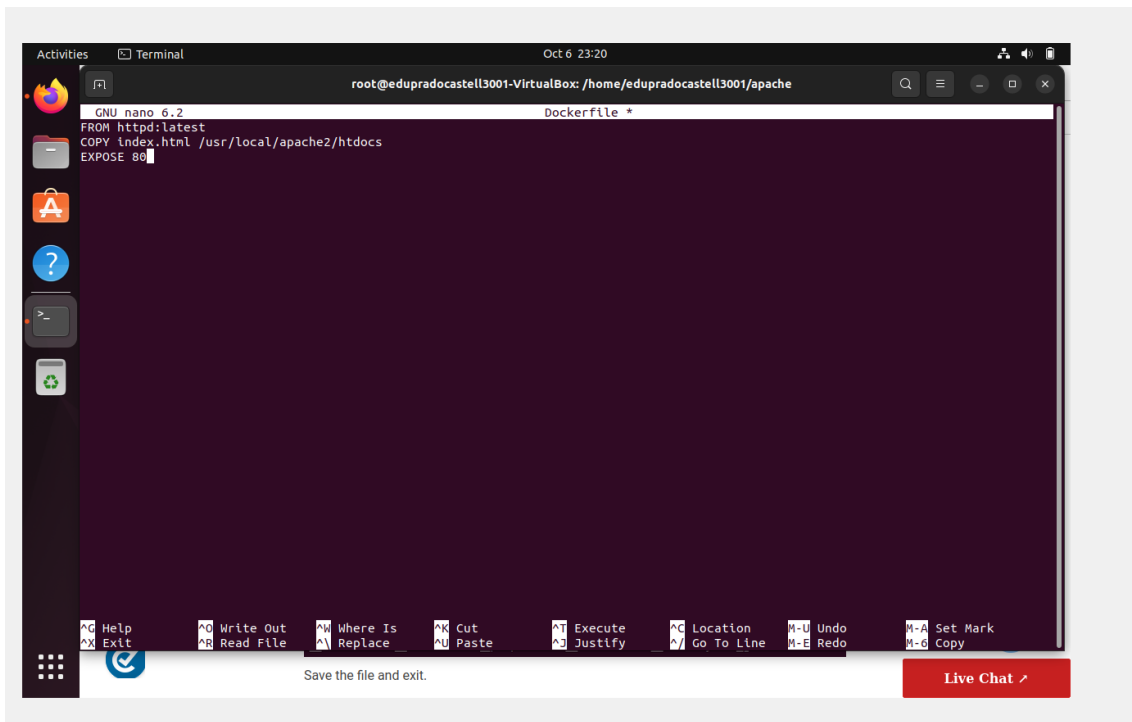
Nano dockerfile



```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# docker run -d --name apache -p 80:80 httpd
5aaf36f1002b3bc5ed0eb13a097566113407edbb440ddc0bd693483f5d92e018
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# docker container ls
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
5aaf36f1002b httpd "httpd-foreground" 9 seconds ago Up 8 seconds 0.0.0.0:80->80/tcp, :::80->80/tcp apache
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# docker kill 5a
5a
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001# mkdir apache && cd apache
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# nano index.html
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# nano Dockerfile
```

Write the image configuration.

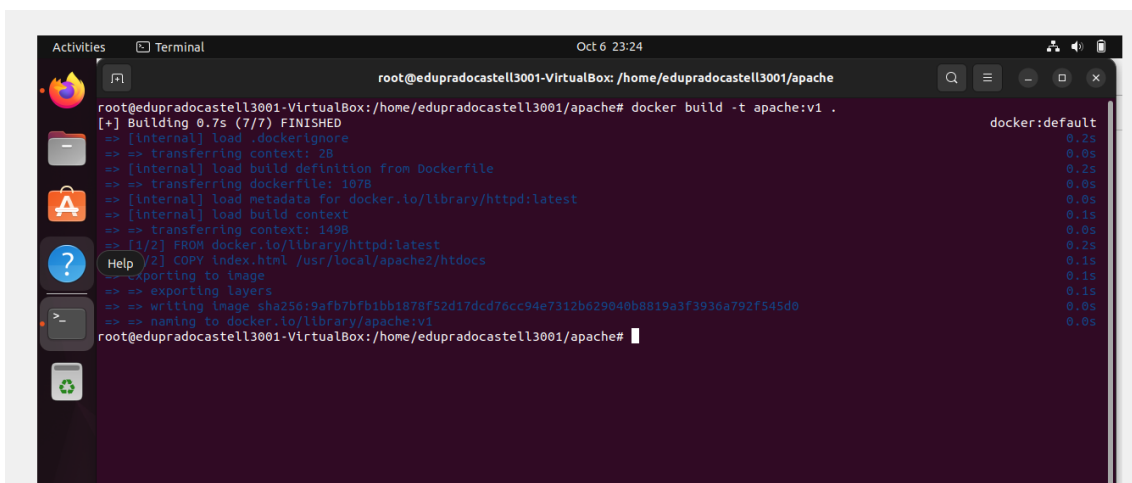
Save the file and exit.



```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache
GNU nano 6.2 Dockerfile *
FROM httpd:latest
COPY index.html /usr/local/apache2/htdocs
EXPOSE 80
```

Use **docker build** to create a Dockerfile-based image.

Docker build -t apache:v1 .

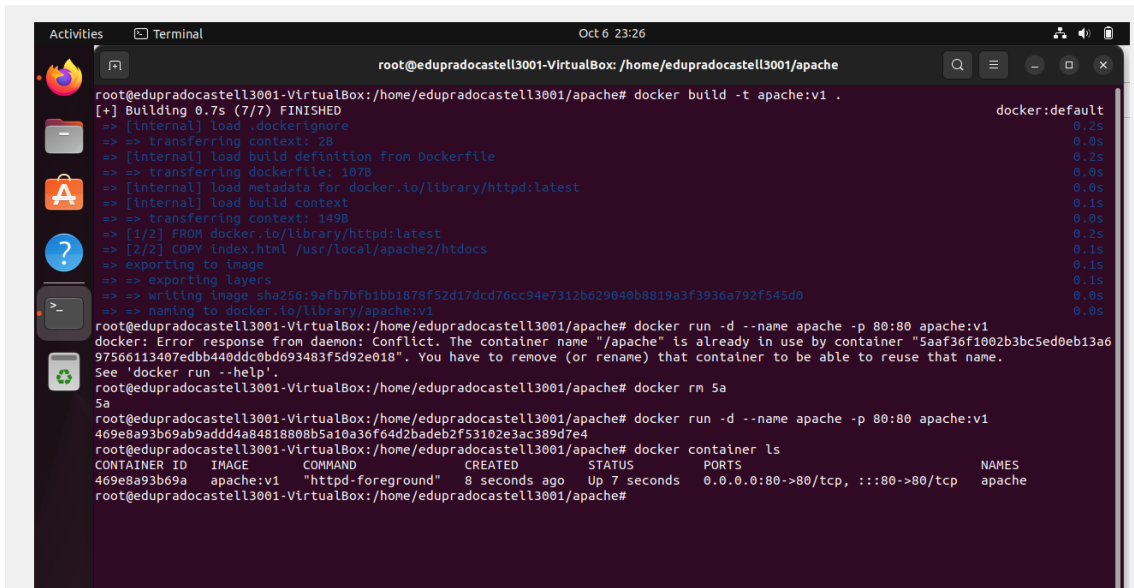


```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker build -t apache:v1 .
[+] Building 0.7s (7/7) FINISHED
=> [internal] load .dockerignore 0.25s
=> => transferring context: 2B 0.05s
=> [internal] load build definition from Dockerfile 0.25s
=> => transferring dockerfile: 107B 0.05s
=> [internal] load metadata for docker.io/library/httpd:latest 0.05s
=> [internal] load build context 0.15s
=> => transferring context: 149B 0.05s
=> [1/2] FROM docker.io/library/httpd:latest 0.25s
=> [2] COPY index.html /usr/local/apache2/htdocs 0.15s
=> exporting to image 0.15s
=> => exporting layers 0.15s
=> => writing image sha256:9afb7bfb1bb1878f52d17dcd76cc94e7312b629040b8819a3f3936a792f545d0 0.05s
=> => naming to docker.io/library/apache:v1 0.05s
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache#
```



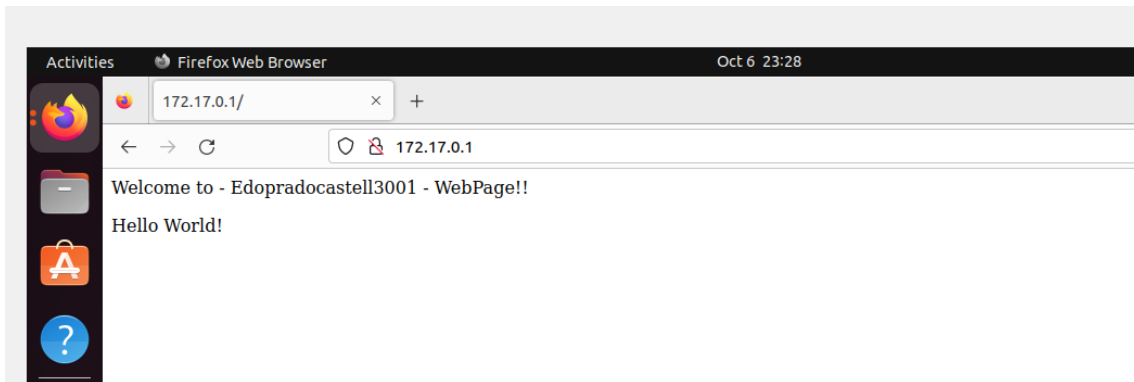
## Run Apache Dockerfile as a Container

Docker run -d --name apache -p 80:80 apache:v1



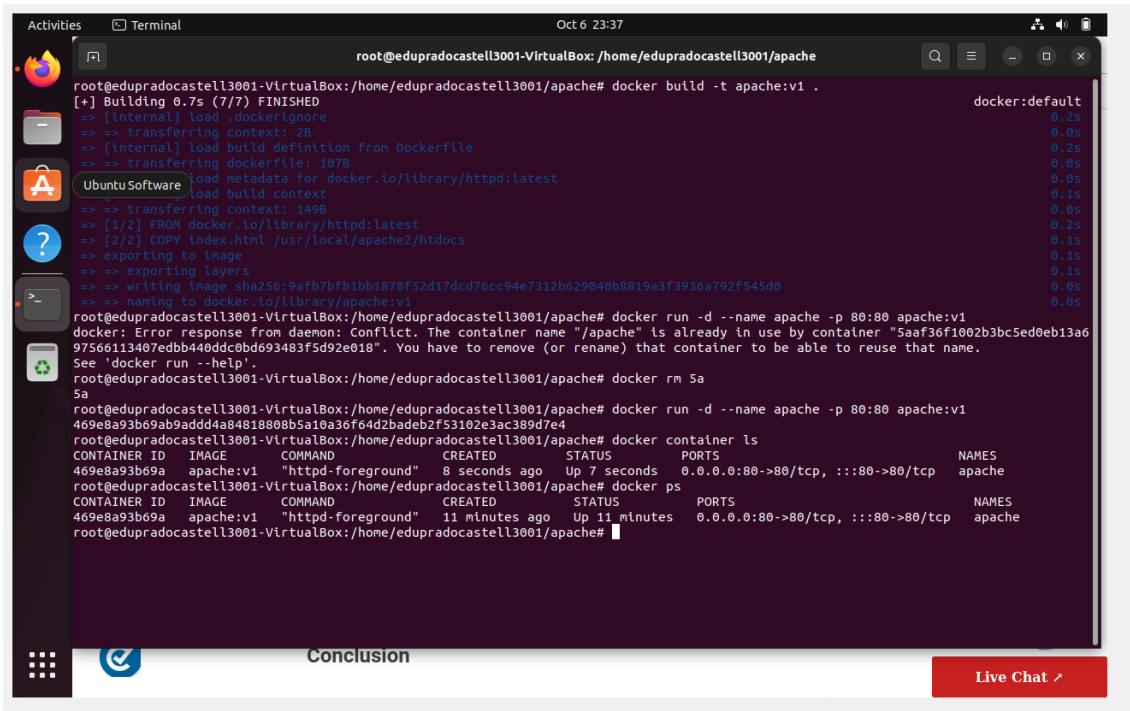
```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker build -t apache:v1 .
[+] Building 0.7s (7/7) FINISHED
=> [internal] load .dockerignore 0.2s
=> => transferring context: 2B 0.0s
=> [internal] load build definition from Dockerfile 0.2s
=> => transferring dockerfile: 107B 0.0s
=> [internal] load metadata for docker.io/library/httpd:latest 0.1s
=> [internal] load build context 0.1s
=> => transferring context: 149B 0.0s
=> [1/2] FROM docker.io/library/httpd:latest 0.2s
=> [2/2] COPY index.html /usr/local/apache2/htdocs 0.1s
=> exporting to image 0.1s
=> => exporting layers 0.1s
=> => writing image sha256:9afb7bfb1bb1878f52d17dcd76cc94e7312b629040b8819a3f3936a792f545d0 0.0s
=> => naming to docker.io/library/apache:v1 0.0s
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker run -d --name apache -p 80:80 apache:v1
docker: Error response from daemon: Conflict. The container name "/apache" is already in use by container "5aaf36f1002b3bc5ed0eb13a697566113407edbb440dc0bd693483f5d92e018". You have to remove (or rename) that container to be able to reuse that name.
See 'docker run --help'.
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker rm 5a
5a
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker run -d --name apache -p 80:80 apache:v1
469e8a93b69ab9add4a84818808b5a10a36f64d2badeb2f53102e3ac389d7e4
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker container ls
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES
469e8a93b69a   apache:v1 "httpd-foreground"      8 seconds ago Up 7 seconds    0.0.0.0:80->80/tcp, :::80->80/tcp   apache
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache#
```

The image below shows the webpage it's my index.html



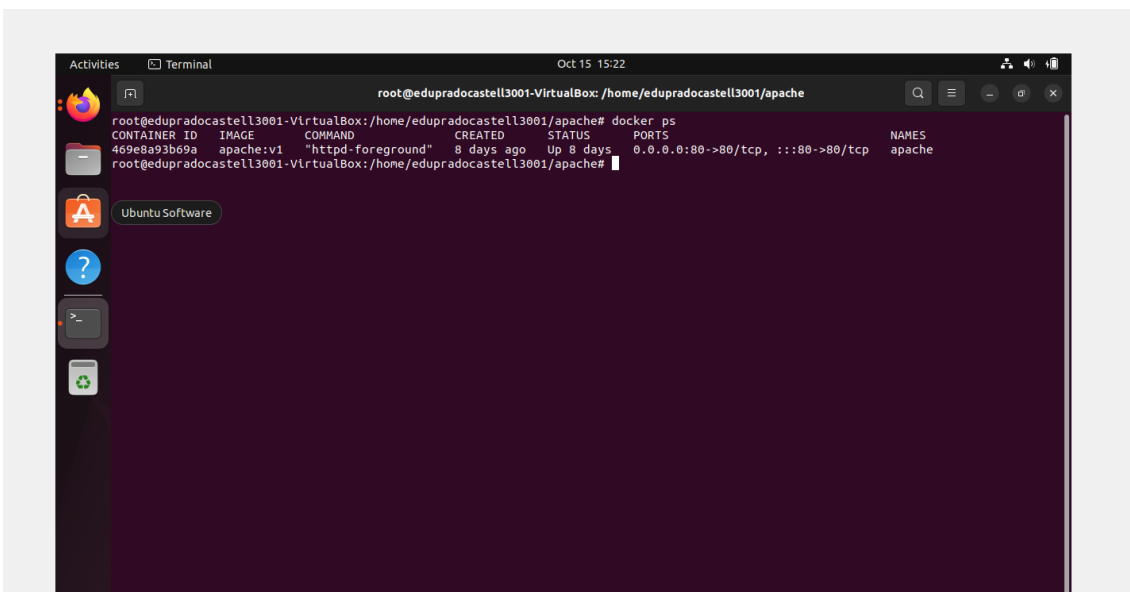
## Docker ps

Container id 469e8a93b69a



```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker build -t apache:v1 .
[+] Building 0.7s (7/7) FINISHED
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] Load build definition from Dockerfile
=> => transferring dockerfile: 107B
=> load metadata for docker.io/library/httpd:latest
=> load build context
=> => transferring context: 149B
=> [1/2] FROM docker.io/library/httpd:latest
=> [2/2] COPY index.html /usr/local/apache2/htdocs
=> exporting to image
=> => exporting layers
=> writing image sha256:9afb7bfb1bb1878f52d17dcd76cc94e7312b629040b8819a3f3936a792f545d0
=> naming to docker.io/library/apache:v1
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker run -d --name apache -p 80:80 apache:v1
docker: Error response from daemon: Conflict. The container name "/apache" is already in use by container "5aaf36f1002b3c5ed0eb13a697566113407edbb440dc0bd693483f5d92e018". You have to remove (or rename) that container to be able to reuse that name.
See 'docker run --help'.
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker rm 5a
5a
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker run -d --name apache -p 80:80 apache:v1
469e8a93b69ab9addd48481880b5a10a36f64d2badeb2f53102e3ac389d7e4
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker container ls
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS    PORTS                               NAMES
469e8a93b69a  apache:v1 "httpd-foreground"      8 seconds ago    Up 7 seconds    0.0.0.0:80->80/tcp, :::80->80/tcp    apache
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS    PORTS                               NAMES
469e8a93b69a  apache:v1 "httpd-foreground"      11 minutes ago    Up 11 minutes    0.0.0.0:80->80/tcp, :::80->80/tcp    apache
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache#
```

In resume after 8 days still up.



```
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache# docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS    PORTS                               NAMES
469e8a93b69a  apache:v1 "httpd-foreground"      8 days ago    Up 8 days    0.0.0.0:80->80/tcp, :::80->80/tcp    apache
root@edupradocastell3001-VirtualBox: /home/edupradocastell3001/apache#
```