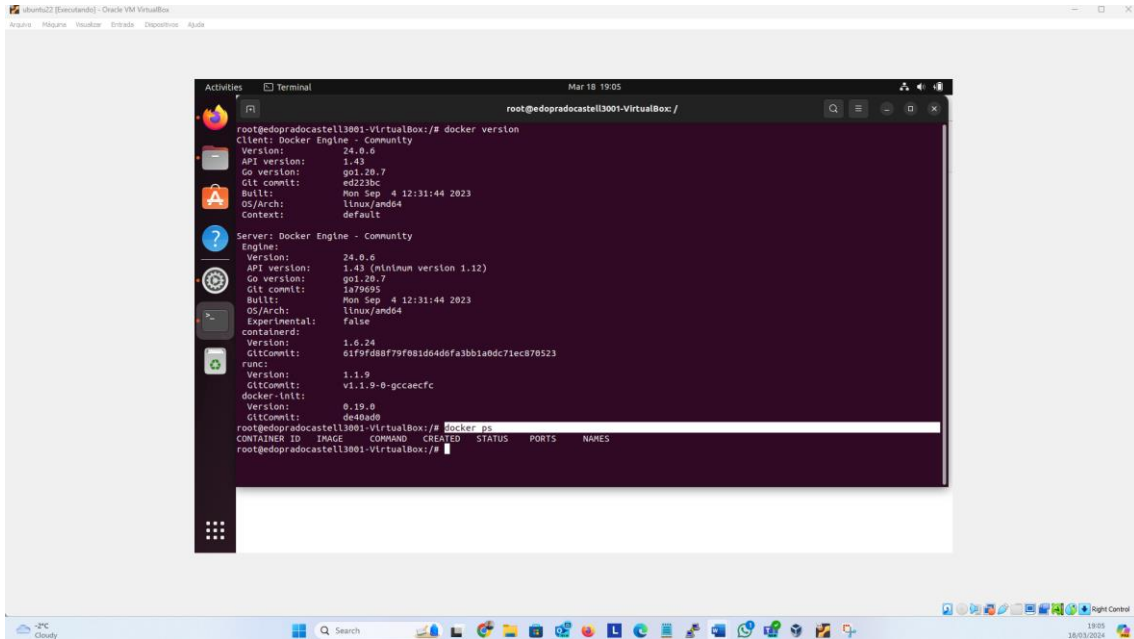
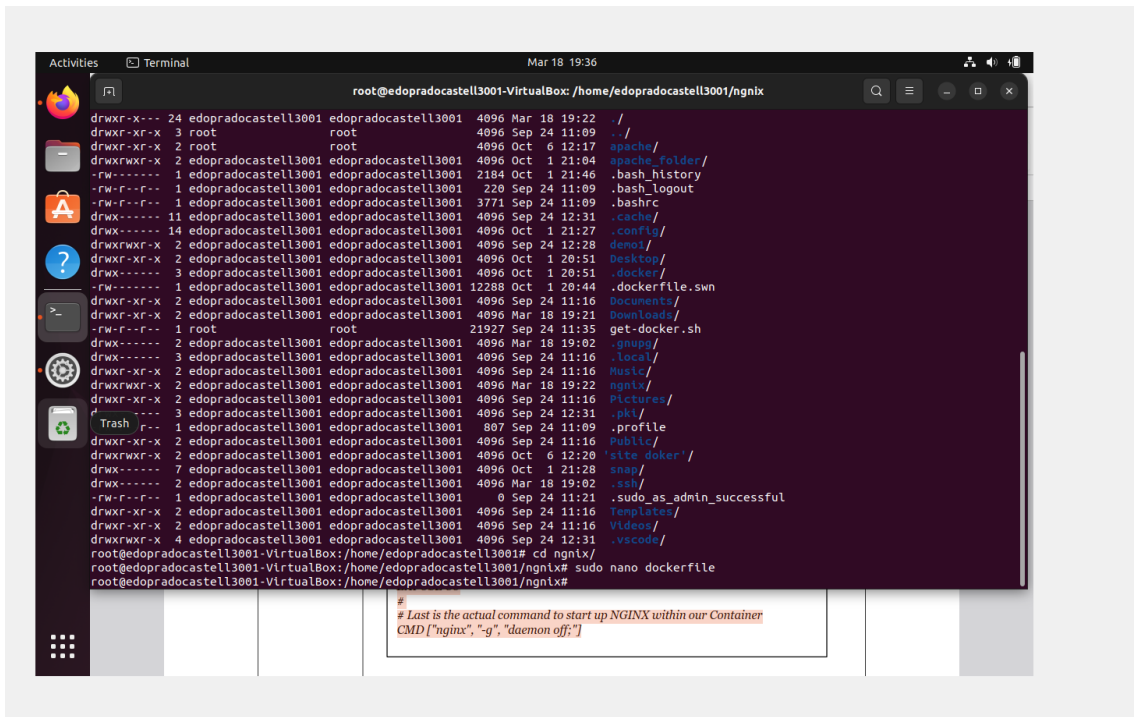


So, I have Docker installed on my virtual machine (VirtualBox), so I can see my name and Conestoga id.

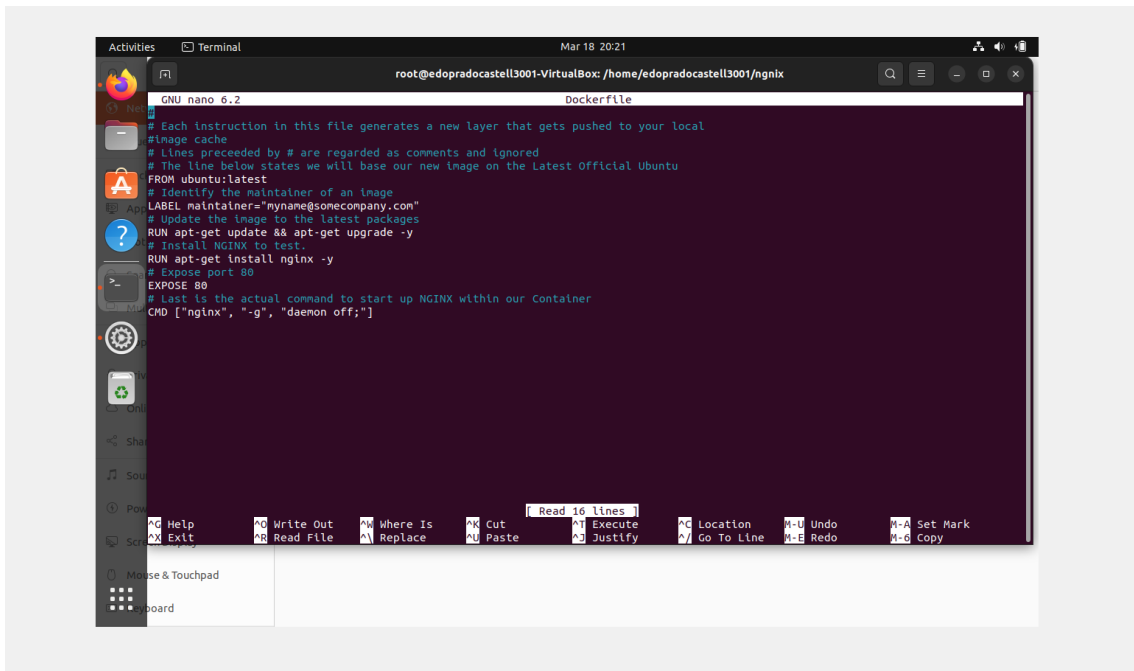
In this screan shot, show the docker version and docker process (docker ps)



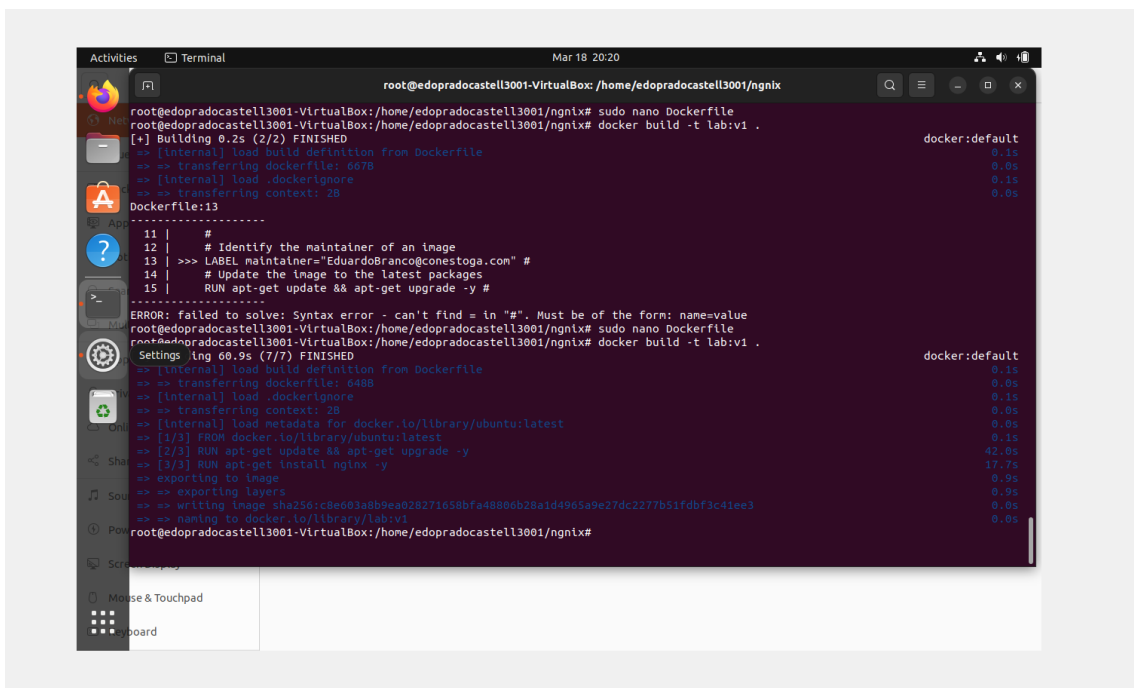
Sudo nano vi Dockerfile



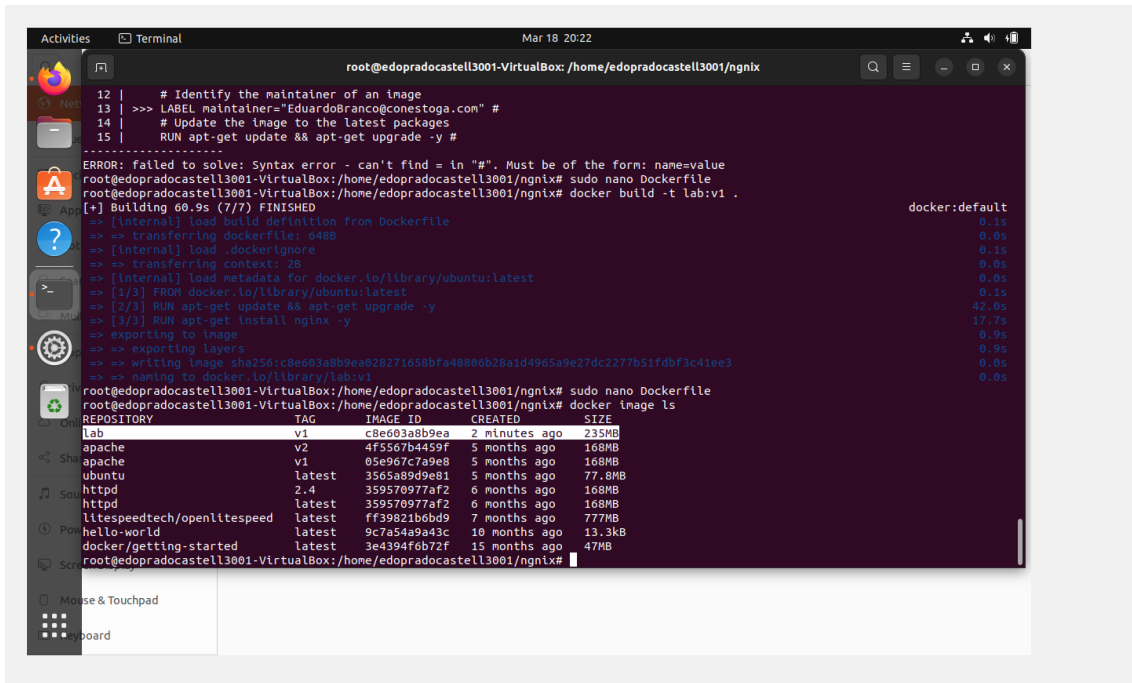
## Edit file Dockerfile



## Build image



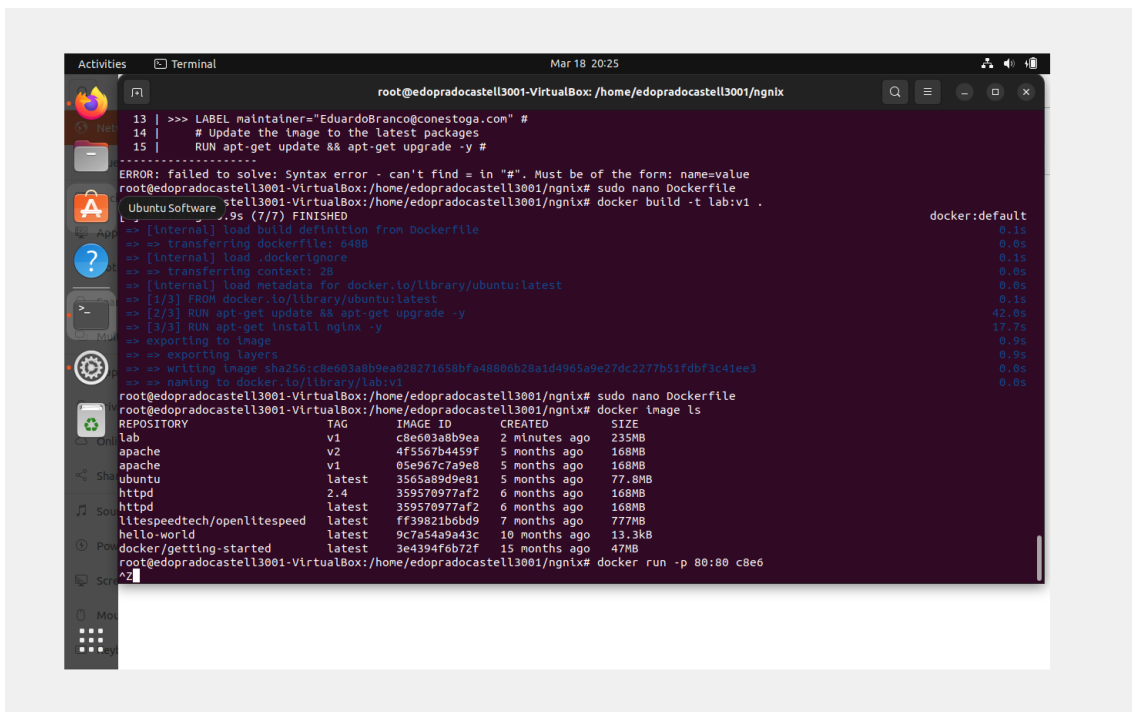
## Docker image List



A terminal window showing the process of building a Docker image. The user is in a virtual machine environment. The terminal output shows the Dockerfile being processed, including instructions to update packages and install nginx. The build process is successful, and the image is named 'lab:v1'. The terminal also shows the output of the 'docker image ls' command, listing the newly built image along with other existing images.

```
12 | # Identify the maintainer of an image
13 | >>> LABEL maintainer="EduardoBranco@conestoga.com" #
14 | # Update the image to the latest packages
15 | RUN apt-get update && apt-get upgrade -y #
-----
ERROR: failed to solve: Syntax error - can't find = in "#". Must be of the form: name=value
root@edopradocastell3001-VirtualBox: /home/edopradocastell3001/nginx# sudo nano Dockerfile
root@edopradocastell3001-VirtualBox: /home/edopradocastell3001/nginx# docker build -t lab:v1 .
[*] Building 60.9s (7/7) FINISHED
=> [Internal] load build definition from Dockerfile
=> == transferring dockerfile: 648B
=> [Internal] load .dockerignore
=> == transferring context: 2B
=> [Internal] load metadata for docker.io/library/ubuntu:latest
=> [1/3] FROM docker.io/library/ubuntu:latest
=> [2/3] RUN apt-get update && apt-get upgrade -y
=> [3/3] RUN apt-get install nginx -y
=> exporting to image
=> == exporting layers
=> == writing image sha256:c8e603a8b9ea028271658bfa48806b28a1d4965a9e27dc2277b51fdbf3c41ee3
=> == naming to docker.io/library/lab:v1
root@edopradocastell3001-VirtualBox: /home/edopradocastell3001/nginx# sudo nano Dockerfile
root@edopradocastell3001-VirtualBox: /home/edopradocastell3001/nginx# docker image ls
REPOSITORY          TAG          IMAGE ID       CREATED        SIZE
lab                  v1          c8e603a8b9ea  2 minutes ago 235MB
apache              v2          4f5567b4459f  5 months ago  168MB
apache              v1          05e967c7a9e8  5 months ago  168MB
ubuntu              latest      3565a89d9e81  5 months ago  77.8MB
httpd               2.4         359570977af2  6 months ago  168MB
httpd               latest      359570977af2  6 months ago  168MB
litespeedtech/openlitespeed latest      ff39821b6bd9  7 months ago  777MB
hello-world         latest      9c7a54a9a43c  10 months ago 13.3kB
docker/getting-started latest      3e4394f6b72f  15 months ago  47MB
root@edopradocastell3001-VirtualBox: /home/edopradocastell3001/nginx#
```

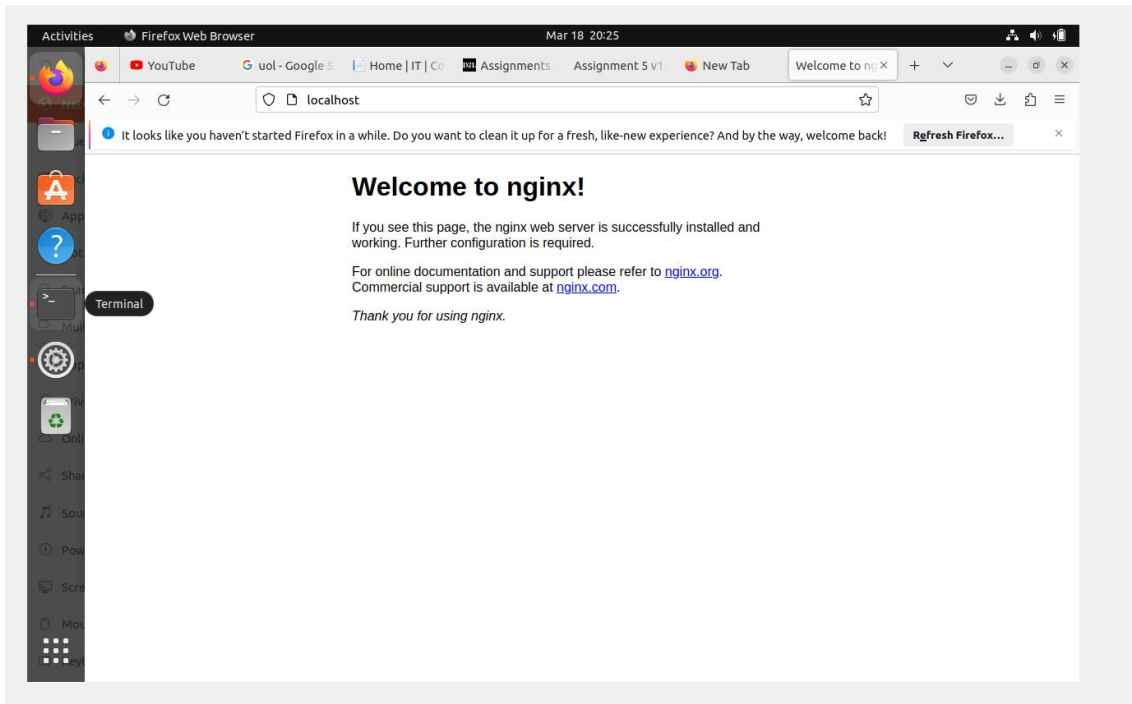
## Run image



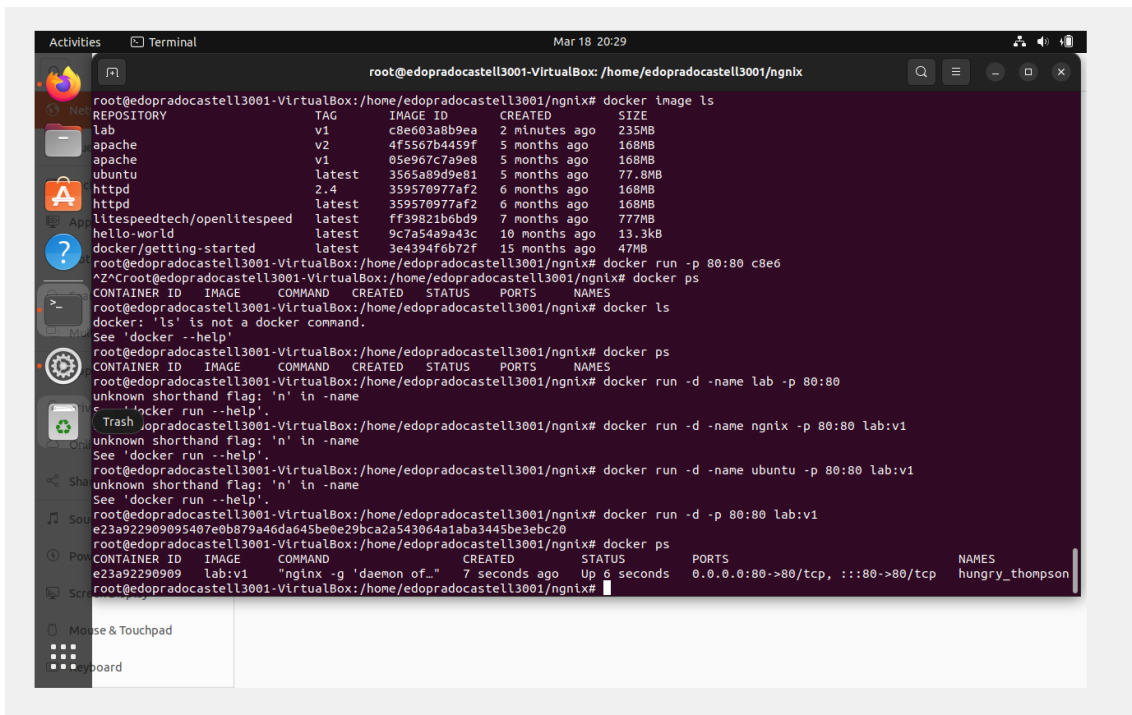
A terminal window showing the execution of the 'docker run' command. The user is in the same virtual machine environment. The terminal output shows the Dockerfile being processed, including instructions to update packages and install nginx. The build process is successful, and the image is named 'lab:v1'. The terminal also shows the output of the 'docker image ls' command, listing the newly built image along with other existing images. The user then runs the command 'docker run -p 80:80 c8e6'.

```
13 | >>> LABEL maintainer="EduardoBranco@conestoga.com" #
14 | # Update the image to the latest packages
15 | RUN apt-get update && apt-get upgrade -y #
-----
ERROR: failed to solve: Syntax error - can't find = in "#". Must be of the form: name=value
root@edopradocastell3001-VirtualBox: /home/edopradocastell3001/nginx# sudo nano Dockerfile
root@edopradocastell3001-VirtualBox: /home/edopradocastell3001/nginx# docker build -t lab:v1 .
[*] Building 60.9s (7/7) FINISHED
=> [Internal] load build definition from Dockerfile
=> == transferring dockerfile: 648B
=> [Internal] load .dockerignore
=> == transferring context: 2B
=> [Internal] load metadata for docker.io/library/ubuntu:latest
=> [1/3] FROM docker.io/library/ubuntu:latest
=> [2/3] RUN apt-get update && apt-get upgrade -y
=> [3/3] RUN apt-get install nginx -y
=> exporting to image
=> == exporting layers
=> == writing image sha256:c8e603a8b9ea028271658bfa48806b28a1d4965a9e27dc2277b51fdbf3c41ee3
=> == naming to docker.io/library/lab:v1
root@edopradocastell3001-VirtualBox: /home/edopradocastell3001/nginx# sudo nano Dockerfile
root@edopradocastell3001-VirtualBox: /home/edopradocastell3001/nginx# docker image ls
REPOSITORY          TAG          IMAGE ID       CREATED        SIZE
lab                  v1          c8e603a8b9ea  2 minutes ago 235MB
apache              v2          4f5567b4459f  5 months ago  168MB
apache              v1          05e967c7a9e8  5 months ago  168MB
ubuntu              latest      3565a89d9e81  5 months ago  77.8MB
httpd               2.4         359570977af2  6 months ago  168MB
httpd               latest      359570977af2  6 months ago  168MB
litespeedtech/openlitespeed latest      ff39821b6bd9  7 months ago  777MB
hello-world         latest      9c7a54a9a43c  10 months ago 13.3kB
docker/getting-started latest      3e4394f6b72f  15 months ago  47MB
root@edopradocastell3001-VirtualBox: /home/edopradocastell3001/nginx# docker run -p 80:80 c8e6
^Z
```

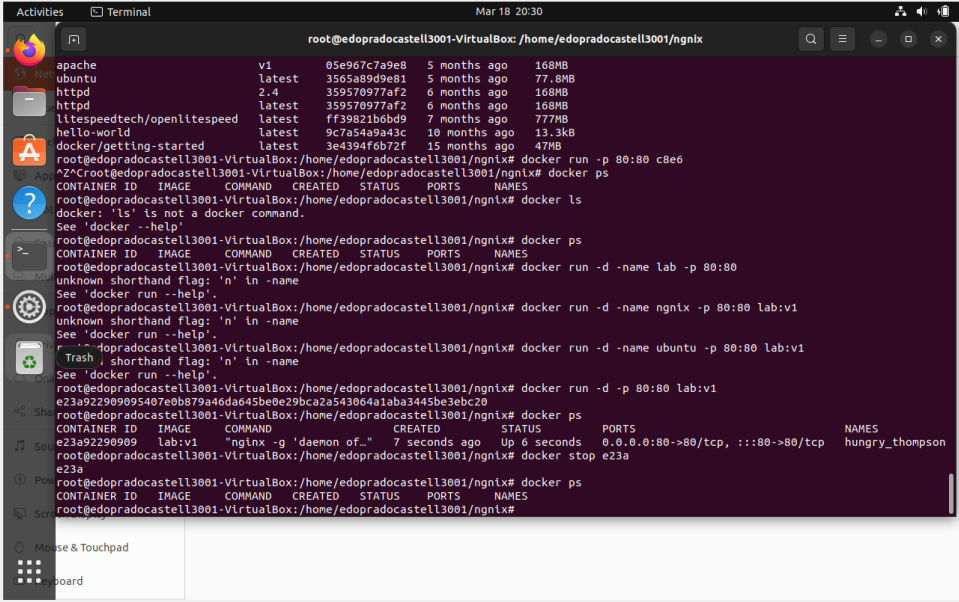
## Nginx working!



## Docker ps .. as can be seen container up



## Stop container



The terminal window shows a series of Docker commands and their outputs. The user is in a shell with the prompt `root@edopradocastell3001-VirtualBox: /home/edopradocastell3001/nginx`. The terminal displays the following sequence of actions:

- A list of Docker images is shown, including `apache`, `ubuntu`, `httpd`, `litespeedtech/openlitespeed`, `hello-world`, and `docker/getting-started`.
- The user runs `docker run -p 80:80 c8e6`, which fails with a "permission denied" error.
- The user runs `docker ps`, showing no containers are running.
- The user runs `docker ls`, receiving an error: `docker: 'ls' is not a docker command.`
- The user runs `docker --help`, displaying the help text for the Docker CLI.
- The user runs `docker ps` again, showing no containers.
- The user runs `docker run -d -name lab -p 80:80`, which fails with the error: `unknown shorthand flag: 'n' in -name`.
- The user runs `docker run -d -name nginx -p 80:80 lab:v1`, which fails with the error: `unknown shorthand flag: 'n' in -name`.
- The user runs `docker run -d -name ubuntu -p 80:80 lab:v1`, which fails with the error: `unknown shorthand flag: 'n' in -name`.
- The user runs `docker run -p 80:80 lab:v1`, which fails with the error: `unknown shorthand flag: 'n' in -name`.
- The user runs `docker ps`, showing a container named `hungry_thompson` with ID `e23a92290909` running `nginx -g 'daemon off;'` on port `80:80->80/tcp`.
- The user runs `docker stop e23a`, which successfully stops the container.
- The user runs `docker ps` again, showing no containers are running.
- The user runs `docker ps` one final time, showing no containers.