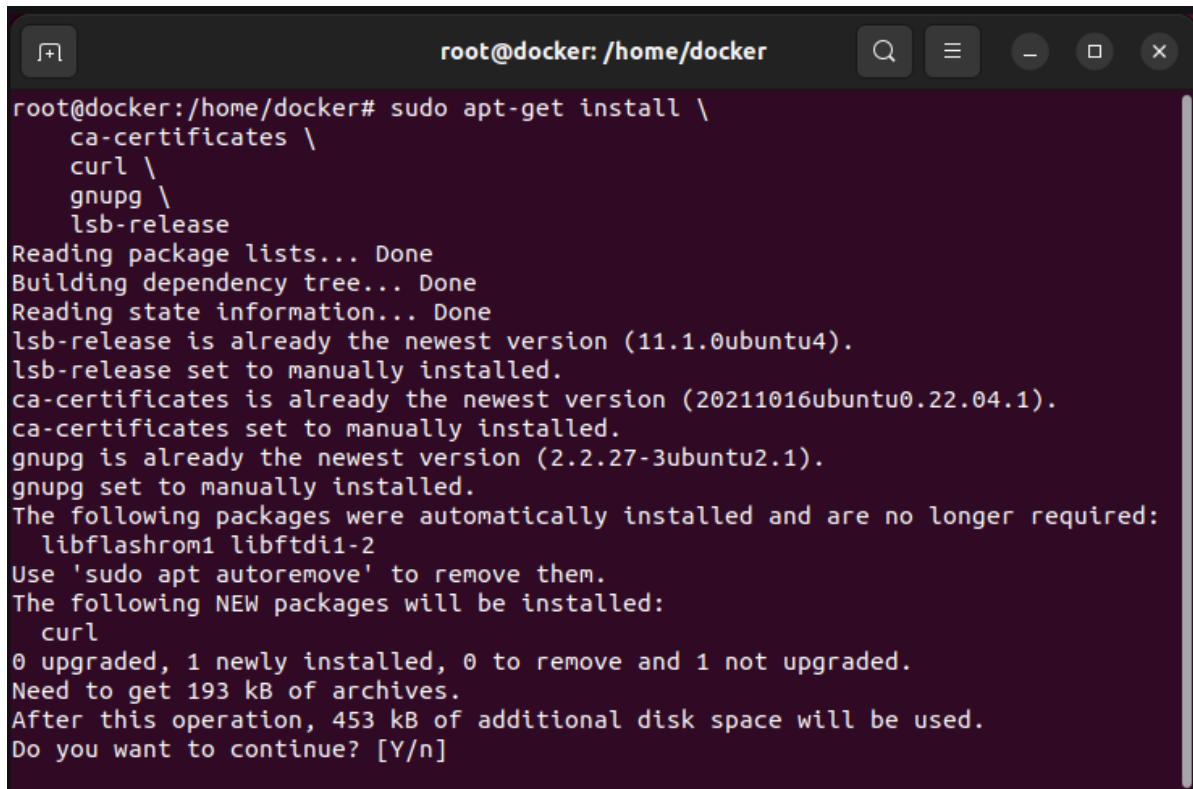


Docker

1. Instala docker en una máquina y configúralo para que se pueda con un usuario sin privilegios.

```
apt install ca-certificates curl gnupg lsb-release
```

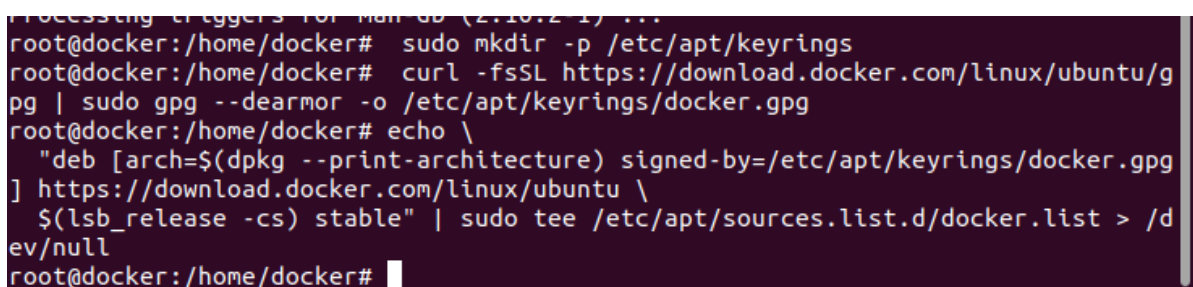


```
root@docker: /home/docker
root@docker:/home/docker# sudo apt-get install \
    ca-certificates \
    curl \
    gnupg \
    lsb-release
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
lsb-release is already the newest version (11.1.0ubuntu4).
lsb-release set to manually installed.
ca-certificates is already the newest version (20211016ubuntu0.22.04.1).
ca-certificates set to manually installed.
gnupg is already the newest version (2.2.27-3ubuntu2.1).
gnupg set to manually installed.
The following packages were automatically installed and are no longer required:
  libflashrom1 libftdi1-2
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  curl
0 upgraded, 1 newly installed, 0 to remove and 1 not upgraded.
Need to get 193 kB of archives.
After this operation, 453 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

```
sudo mkdir -p /etc/apt/keyrings
```

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o
/etc/apt/keyrings/docker.gpg
```

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



```
Processing triggers for man-db (2.10.2-1) ...
root@docker:/home/docker# sudo mkdir -p /etc/apt/keyrings
root@docker:/home/docker# curl -fsSL https://download.docker.com/linux/ubuntu/g
pg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
root@docker:/home/docker# echo \
    "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg
] https://download.docker.com/linux/ubuntu \
    $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /d
ev/null
root@docker:/home/docker#
```

```
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
```

```

root@docker:/home/docker# sudo apt-get install docker-ce docker-ce-cli container
d.io docker-compose-plugin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libflashrom1 libftdi1-2
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  docker-ce-rootless-extras docker-scan-plugin git git-man liberror-perl
  libslirp0 pigz slirp4netns
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite git-daemon-run | git-daemon-sysvinit
  git-doc git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  containerd.io docker-ce docker-ce-cli docker-ce-rootless-extras
  docker-compose-plugin docker-scan-plugin git git-man liberror-perl libslirp0
  pigz slirp4netns
0 upgraded, 12 newly installed, 0 to remove and 1 not upgraded.
Need to get 116 MB of archives.
After this operation, 449 MB of additional disk space will be used.
Do you want to continue? [Y/n]

```

2. Ejecuta un contenedor a partir de la imagen hello-world. Comprueba que nos devuelve la salida adecuada. Comprueba que no se está ejecutando. Lista los contenedores que están parado. Borra el contenedor.

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

```

Processing triggers for libc-bin (2.33-0ubuntu3.1) ...
root@docker:/home/docker# sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:aa0cc8055b82dc2509bed2e19b275c8f463506616377219d9642221ab53cf9fe
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/

root@docker:/home/docker#

```

3. Crea un contenedor interactivo desde una imagen debian. Instala un paquete (por ejemplo nano). Sal de la terminal, ¿sigue el contenedor corriendo? ¿Por qué?. Vuelve a iniciar el contenedor y accede de nuevo a él de forma interactiva. ¿Sigue instalado el nano?. Sal del contenedor, y bórralo. Crea un nuevo contenedor interactivo desde la misma imagen. ¿Tiene el nano instalado?

```
root@docker: /home/docker
root@docker:/home/docker# docker ps
CONTAINER ID   IMAGE          COMMAND         CREATED        STATUS        PORTS        NAMES
root@docker:/home/docker# docker ps -a
CONTAINER ID   IMAGE          COMMAND         CREATED        STATUS        PORTS        NAMES
44a59df8bc42   hello-world    "/hello"       3 minutes ago  Exited (0)    3 minutes ago  flamboyant_murdock
root@docker:/home/docker# docker run debian
root@docker:/home/docker# docker ps -a
CONTAINER ID   IMAGE          COMMAND         CREATED        STATUS        PORTS        NAMES
a425f18bd5e9   debian        "bash"         2 seconds ago  Exited (0)    2 seconds ago  peaceful_wing
44a59df8bc42   hello-world    "/hello"       4 minutes ago  Exited (0)    4 minutes ago  flamboyant_murdock
root@docker:/home/docker#
```

```
flamboyant_murdock
root@docker:/home/docker# docker run -it debian bash
root@c442019da111:/# apt install nano
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
E: Unable to locate package nano
root@c442019da111:/# apt update
Get:1 http://deb.debian.org/debian bullseye InRelease [116 kB]
Get:2 http://deb.debian.org/debian-security bullseye-security InRelease [48.4 kB]
Get:3 http://deb.debian.org/debian bullseye-updates InRelease [44.1 kB]
Get:4 http://deb.debian.org/debian bullseye/main amd64 Packages [8183 kB]
Get:5 http://deb.debian.org/debian-security bullseye-security/main amd64 Packages [214 kB]
Get:6 http://deb.debian.org/debian bullseye-updates/main amd64 Packages [14.6 kB]
Fetched 8620 kB in 1s (8903 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
root@c442019da111:/# apt install nano
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libgpm2 libncursesw6
Suggested packages:
  gpm hunspell
The following NEW packages will be installed:
  libgpm2 libncursesw6 nano
0 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 825 kB of archives.
After this operation, 3087 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

```
Processing triggers for libc-bin (2.31-13+deb11u5) ...
root@c442019da111:/# exit
exit
root@docker:/home/docker# docker ps
CONTAINER ID   IMAGE          COMMAND         CREATED        STATUS        PORTS        NAMES
root@docker:/home/docker#
```

El contenedor no sigue corriendo, ya que con el run -it al salir, se cierra directamente. Al volver a ejecutarlo, el paquete nano no esta instalado ya que se crea un nuevo contenedor.

```

root@docker:/home/docker# docker ps -a
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS
PORTS         NAMES
827065284b32   debian    "bash"                  About a minute ago    Exited (127) 1 seco
nd ago
c442019da111   debian    "bash"                  4 minutes ago        Exited (0) 2 minute
s ago
a425f18bd5e9   debian    "bash"                  5 minutes ago        Exited (0) 5 minute
s ago
44a59df8bc42   hello-world "/hello"               10 minutes ago       Exited (0) 10 minut
es ago
root@docker:/home/docker# docker rm 827065284b32 c442019da111 a425f18bd5e9
827065284b32
c442019da111
a425f18bd5e9
root@docker:/home/docker#

```

4. Crea un contenedor demonio con un servidor nginx, usando la imagen oficial de nginx. Al crear el contenedor, ¿has tenido que indicar algún comando para que lo ejecute? Accede al navegador web y comprueba que el servidor esta funcionando. Muestra los logs del contenedor.

```
docker pull nginx
```

```

root@docker:/home/docker# docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
8740c948ffd4: Pull complete
d2c0556a17c5: Pull complete
c8b9881f2c6a: Pull complete
693c3ffa8f43: Pull complete
8316c5e80e6d: Pull complete
b2fe3577faa4: Pull complete
Digest: sha256:b8f2383a95879e1ae064940d9a200f67a6c79e710ed82ac42263397367e7cc4e
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
root@docker:/home/docker#

```

Para ejecutarlo, hay que ejecutar el siguiente comando:

```
docker run nginx
```

```
root@docker:/home/docker# docker run nginx
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2023/01/16 11:29:38 [notice] 1#1: using the "epoll" event method
2023/01/16 11:29:38 [notice] 1#1: nginx/1.23.3
2023/01/16 11:29:38 [notice] 1#1: built by gcc 10.2.1 20210110 (Debian 10.2.1-6)

2023/01/16 11:29:38 [notice] 1#1: OS: Linux 5.15.0-58-generic
2023/01/16 11:29:38 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2023/01/16 11:29:38 [notice] 1#1: start worker processes
2023/01/16 11:29:38 [notice] 1#1: start worker process 29
2023/01/16 11:29:38 [notice] 1#1: start worker process 30
2023/01/16 11:29:38 [notice] 1#1: start worker process 31
2023/01/16 11:29:38 [notice] 1#1: start worker process 32
2023/01/16 11:29:38 [notice] 1#1: start worker process 33
2023/01/16 11:29:38 [notice] 1#1: start worker process 34
2023/01/16 11:29:38 [notice] 1#1: start worker process 35
2023/01/16 11:29:38 [notice] 1#1: start worker process 36
2023/01/16 11:29:38 [notice] 1#1: start worker process 37
2023/01/16 11:29:38 [notice] 1#1: start worker process 38
2023/01/16 11:29:38 [notice] 1#1: start worker process 39
2023/01/16 11:29:38 [notice] 1#1: start worker process 40
2023/01/16 11:29:38 [notice] 1#1: start worker process 41
2023/01/16 11:29:38 [notice] 1#1: start worker process 42
2023/01/16 11:29:38 [notice] 1#1: start worker process 43
2023/01/16 11:29:38 [notice] 1#1: start worker process 44
```

Para ver la IP del docker, se hace el siguiente comando:

```
docker inspect id_contenedor
```



```

      "80/tcp": null
    },
    "SandboxKey": "/var/run/docker/netns/9e11d7b11945",
    "SecondaryIPAddresses": null,
    "SecondaryIPv6Addresses": null,
    "EndpointID": "7b345411a300b1762f90c24d324ec5d02fe3d6565dcd7c1152fa3
0dec8442582",
    "Gateway": "172.17.0.1",
    "GlobalIPv6Address": "",
    "GlobalIPv6PrefixLen": 0,
    "IPAddress": "172.17.0.2",
    "IPPrefixLen": 16,
    "IPv6Gateway": "",
    "MacAddress": "02:42:ac:11:00:02",
    "Networks": {
      "bridge": {
        "IPAMConfig": null,
        "Links": null,
        "Aliases": null,
        "NetworkID": "42e08a97a33435b1522ac18549b7d57edae627f3e89495
298be6415de2ff3aec",
        "EndpointID": "7b345411a300b1762f90c24d324ec5d02fe3d6565dcd7
c1152fa30dec8442582",
        "Gateway": "172.17.0.1",
        "IPAddress": "172.17.0.2",
        "IPPrefixLen": 16,
        "IPv6Gateway": "",
        "GlobalIPv6Address": "",
        "GlobalIPv6PrefixLen": 0,
        "MacAddress": "02:42:ac:11:00:02",
        "DriverOpts": null
      }
    }
  }
}
]
docker@docker:~$

```

Y el resultado final, es este:

