

Ejercicio: Creación de una imagen personalizada y despliegue en Docker Hub

1. Arranca un contenedor basado en Ubuntu 22.04

- Ejecuta el siguiente comando para arrancar un contenedor interactivo basado en `ubuntu:22.04`:
- `docker run -it --name ubuntu_redes ubuntu:22.04 /bin/bash`

```
PS C:\Users\alcar> docker run -it --name ubuntu_redes ubuntu:22.04 /bin/bash
root@f934498af563:/# |
```

2. Instala los paquetes de red

- Una vez dentro del contenedor, actualiza los paquetes e instala las herramientas de red necesarias:
- `apt update && apt install -y inetutils-ping iproute2 dnsutils`

```
root@f934498af563:/# apt update && apt install -y inetutils-ping iproute2 dnsutils
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy InRelease [270 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [1162 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [2374 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [3200 kB]
Get:6 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [44.7 kB]
Get:7 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:8 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:9 http://archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [17.5 MB]
Get:10 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [266 kB]
Get:11 http://archive.ubuntu.com/ubuntu jammy/main amd64 Packages [1792 kB]
Get:12 http://archive.ubuntu.com/ubuntu jammy/restricted amd64 Packages [164 kB]
Get:13 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2650 kB]
```

3. Crea una imagen personalizada

- Tras realizar las instalaciones, abre una nueva terminal y ejecuta el siguiente comando para crear una imagen a partir del contenedor `ubuntu_redes`:
- `docker commit ubuntu_redes <tu_usuario_docker_hub>/ubuntu_redes`

```
PS C:\Users\alcar> docker commit ubuntu_redes cperazu/ubuntu_redes
sha256:27e58ade1f7e809595fedfb5feb71feffa848b34bdc2f39f3b1bf9ea98b7f35f
PS C:\Users\alcar> |
```

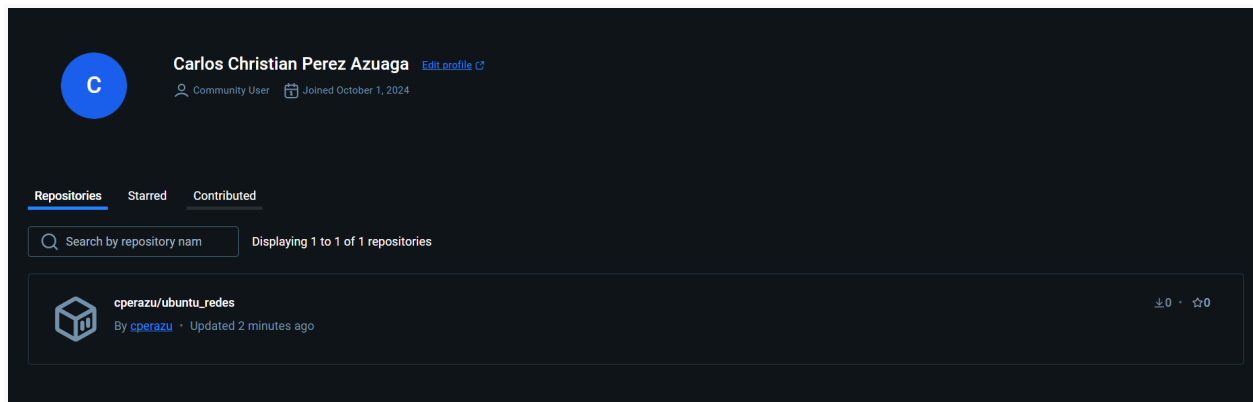
4. Subir la imagen a Docker Hub

- Primero, asegúrate de haber iniciado sesión en Docker Hub con:
- `docker login`

```
PS C:\Users\alcar> docker login
Authenticating with existing credentials...
Login Succeeded
PS C:\Users\alcar> |
```

- Luego, sube la imagen creada al Docker Hub con:
- `docker push <tu_usuario_docker_hub>/ubuntu_redes`

```
PS C:\Users\alcar> docker push cperazu/ubuntu_redes
Using default tag: latest
The push refers to repository [docker.io/cperazu/ubuntu_redes]
4c10b51deb0c: Pushed
6414378b6477: Mounted from library/ubuntu
latest: digest: sha256:27e58ade1f7e809595fedfb5feb71feffa848b34bdc2f39f3b1bf9ea98b7f35f size: 751
PS C:\Users\alcar> |
```



- Ayuda adicional:
 - Si es la primera vez que usas Docker Hub, debes crear una cuenta en hub.docker.com.
 - El nombre de la imagen debe seguir el formato `usuario/nombre_imagen`, donde `usuario` es tu nombre de usuario en Docker Hub.
 - Asegúrate de etiquetar bien la imagen para que pueda ser identificada correctamente.

Entrega

- Captura y entrega pantallazos en un documento PDF con los siguientes elementos:
 - La creación del contenedor `ubuntu_redes`.
 - El comando `docker commit` utilizado para crear la imagen personalizada.
 - La confirmación de que la imagen ha sido subida a Docker Hub (puedes mostrar la imagen en tu cuenta de Docker Hub).