EJERCICIOS DOCKERS - 3

Ejercicios para entregar

- **Trabajar con redes docker**

1. Vamos a crear dos redes de ese tipo (BRIDGE) con los siguientes datos:

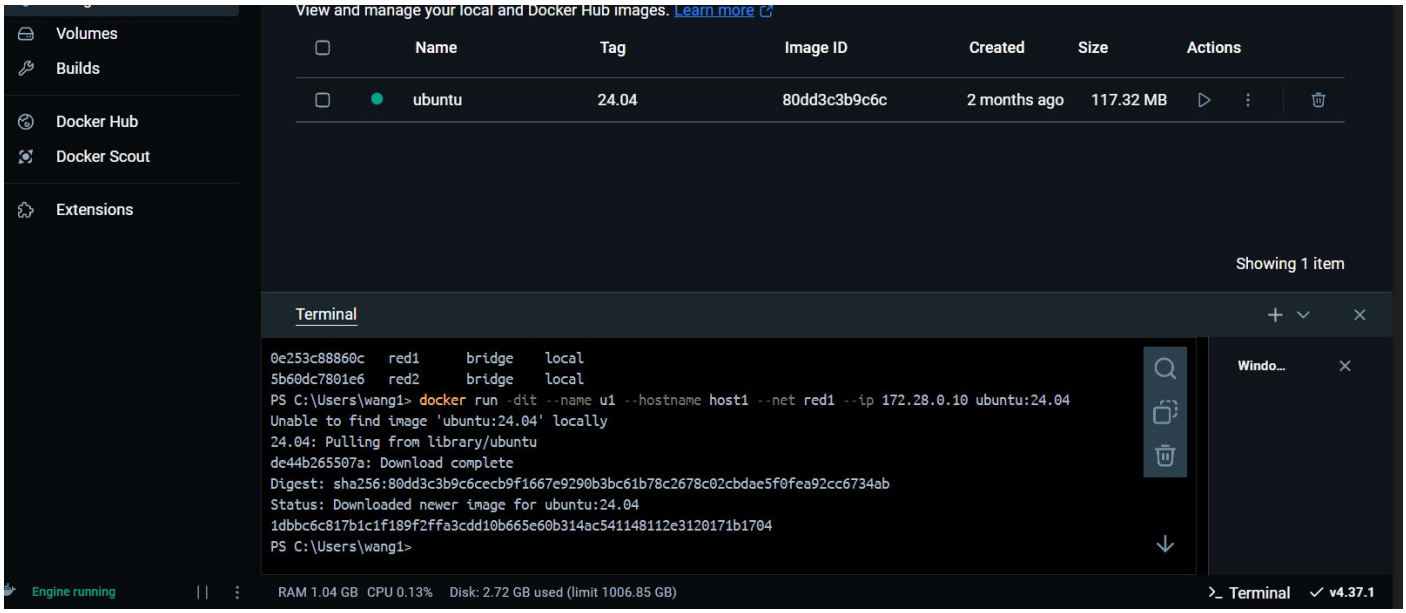
   Red1

   - Nombre: red1
   - Dirección de red: 172.28.0.0
   - Máscara de red: 255.255.0.0
   - Gateway: 172.28.0.1

   Red2

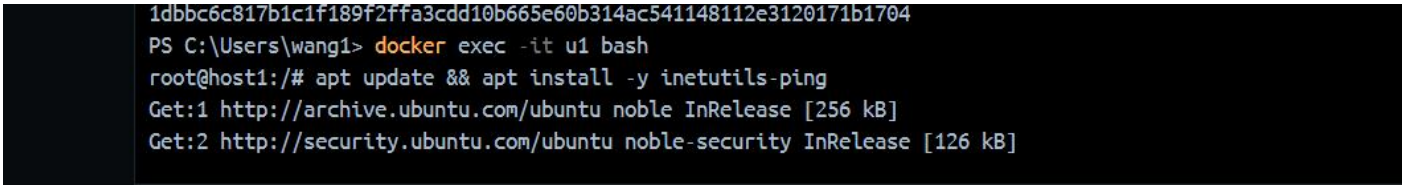
   - Nombre: red2
   - El resto de los datos será proporcionado automáticamente por Docker.

2. Poner en ejecución un contenedor de la imagen `ubuntu:24.04` que tenga como hostname `host1`, como IP `172.28.0.10` y que esté conectado a la red1. Lo llamaremos `u1`.



docker run -dit --name u1 --hostname host1 --net red1 --ip 172.28.0.10 ubuntu:24.04

3. Entrar en ese contenedor e instalar la aplicación ping (`apt update && apt install inetutils-ping`).



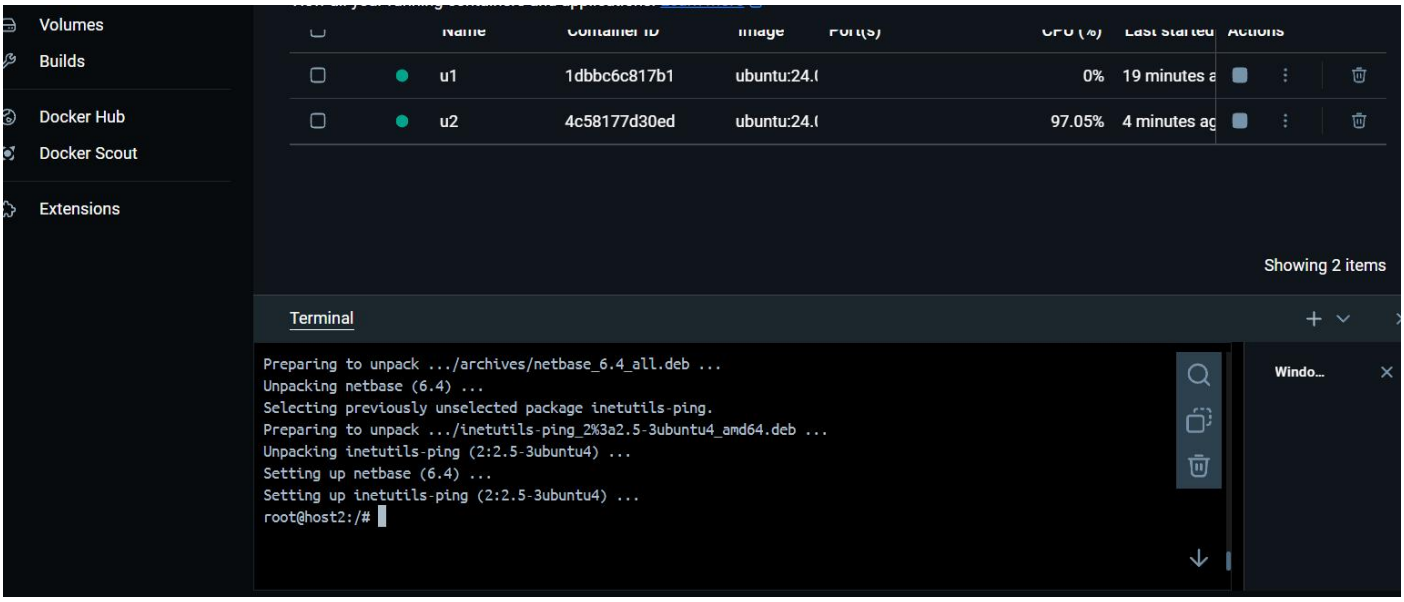

```
docker exec -it u1 bash
apt update && apt install -y inetutils-ping
Exit
```

4. Poner en ejecución un contenedor de la imagen `ubuntu:24.04` que tenga como hostname `host2` y que esté conectado a la red2. En este caso será docker el que le dé una IP correspondiente a esa red. Lo llamaremos `u2`.

```
docker run -dit --name u2 --hostname host2 --net red2 ubuntu:24.04
```

`ping`).

5. Entrar en ese contenedor e instalar la aplicación ping (`apt update && apt install inetutils-`




```
docker exec -it u2 bash

apt update && apt install -y inetutils-ping

exit
```

Deberás entregar los siguientes pantallazos en un documento pdf:

- Pantallazo donde se vea la configuración de red del contenedor u1.
- Pantallazo donde se vea la configuración de red del contenedor u2.
- Pantallazo donde desde cualquiera de los dos contenedores se pueda ver que no podemos hacer ping al otro ni por ip ni por nombre.

```
 in $PATH: unknown
PS C:\Users\wang1> docker exec -it u1 ping host2
ping: unknown host
PS C:\Users\wang1> docker exec -it u1 ping 172.28.0.10
PING 172.28.0.10 (172.28.0.10): 56 data bytes
64 bytes from 172.28.0.10: icmp_seq=0 ttl=64 time=0.104 ms
64 bytes from 172.28.0.10: icmp_seq=1 ttl=64 time=0.107 ms
PS C:\Users\wang1> ^C
PS C:\Users\wang1> docker exec -it u2 ping host1
ping: unknown host
PS C:\Users\wang1>
```

- Pantallazo donde se pueda comprobar que si conectamos el contenedor `u1` a la red2 (con `docker network connect`), desde el contenedor u1, tenemos acceso al contenedor u2 mediante ping, tanto por nombre como por ip.

docker network connect red2 u1

docker exec -it u1 ping host2

```
ping: unknown host
PS C:\Users\wang1> docker network connect red2 u1
PS C:\Users\wang1> docker exec -it u1 ping host2
>>
PING host2 (172.18.0.2): 56 data bytes
64 bytes from 172.18.0.2: icmp_seq=0 ttl=64 time=0.186 ms
64 bytes from 172.18.0.2: icmp_seq=1 ttl=64 time=0.135 ms
64 bytes from 172.18.0.2: icmp_seq=2 ttl=64 time=0.072 ms
```

docker exec -it u1 ping host2

```
^C--- host2 ping statistics ---
10 packets transmitted, 10 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.052/0.099/0.186/0.044 ms
PS C:\Users\wang1> docker exec -it u2 ping host1
PING host1 (172.18.0.3): 56 data bytes
64 bytes from 172.18.0.3: icmp_seq=0 ttl=64 time=0.077 ms
64 bytes from 172.18.0.3: icmp_seq=1 ttl=64 time=0.061 ms
64 bytes from 172.18.0.3: icmp_seq=2 ttl=64 time=0.051 ms
64 bytes from 172.18.0.3: icmp_seq=3 ttl=64 time=0.133 ms
64 bytes from 172.18.0.3: icmp_seq=4 ttl=64 time=0.134 ms
```

- ## Despliegue de Wordpress + Mariadb

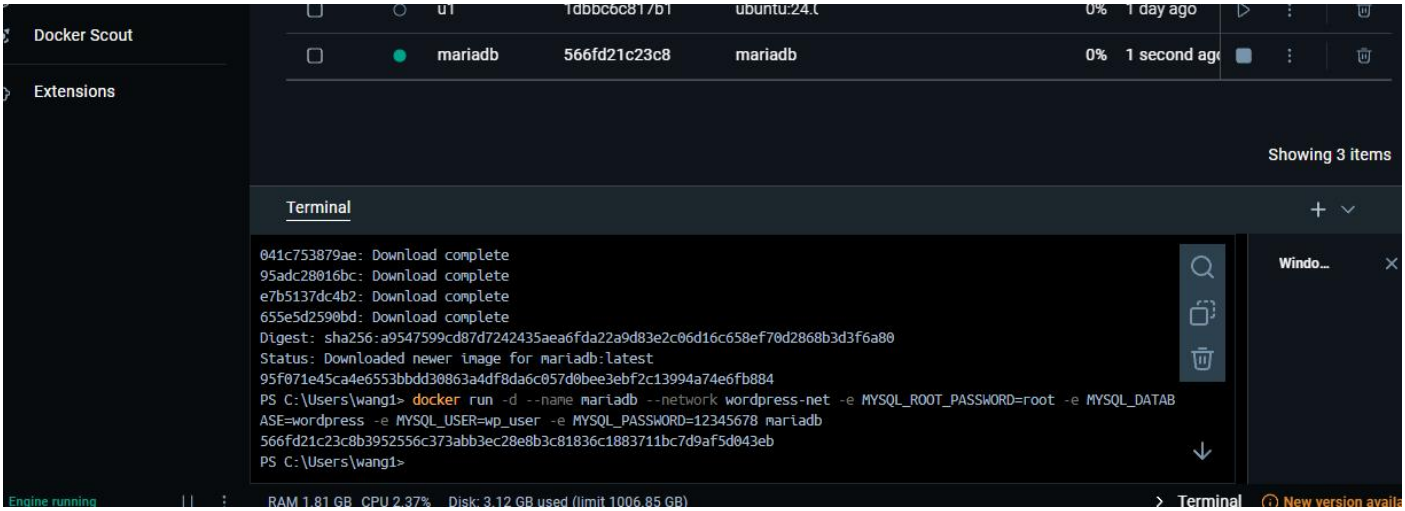
Para la instalación de WordPress necesitamos dos contenedores: la base de datos (imagen `mariadb`) y el servidor web con la aplicación (imagen `wordpress`). Los dos contenedores tienen que estar en la misma red y deben tener acceso por nombres (resolución DNS) ya que de principio no sabemos que ip va a coger cada contenedor. Por lo tanto, vamos a crear los contenedores en la misma red.

Crear una red Docker para ambos contenedores

docker network create wordpress-net

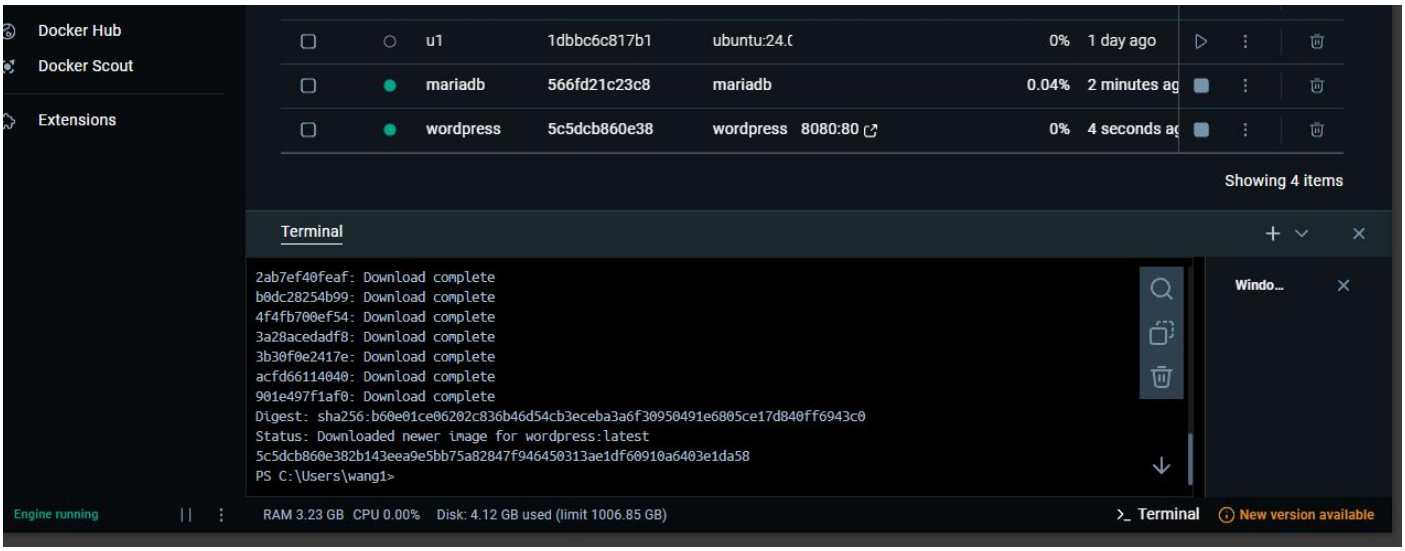
Crear el contenedor de MariaDB

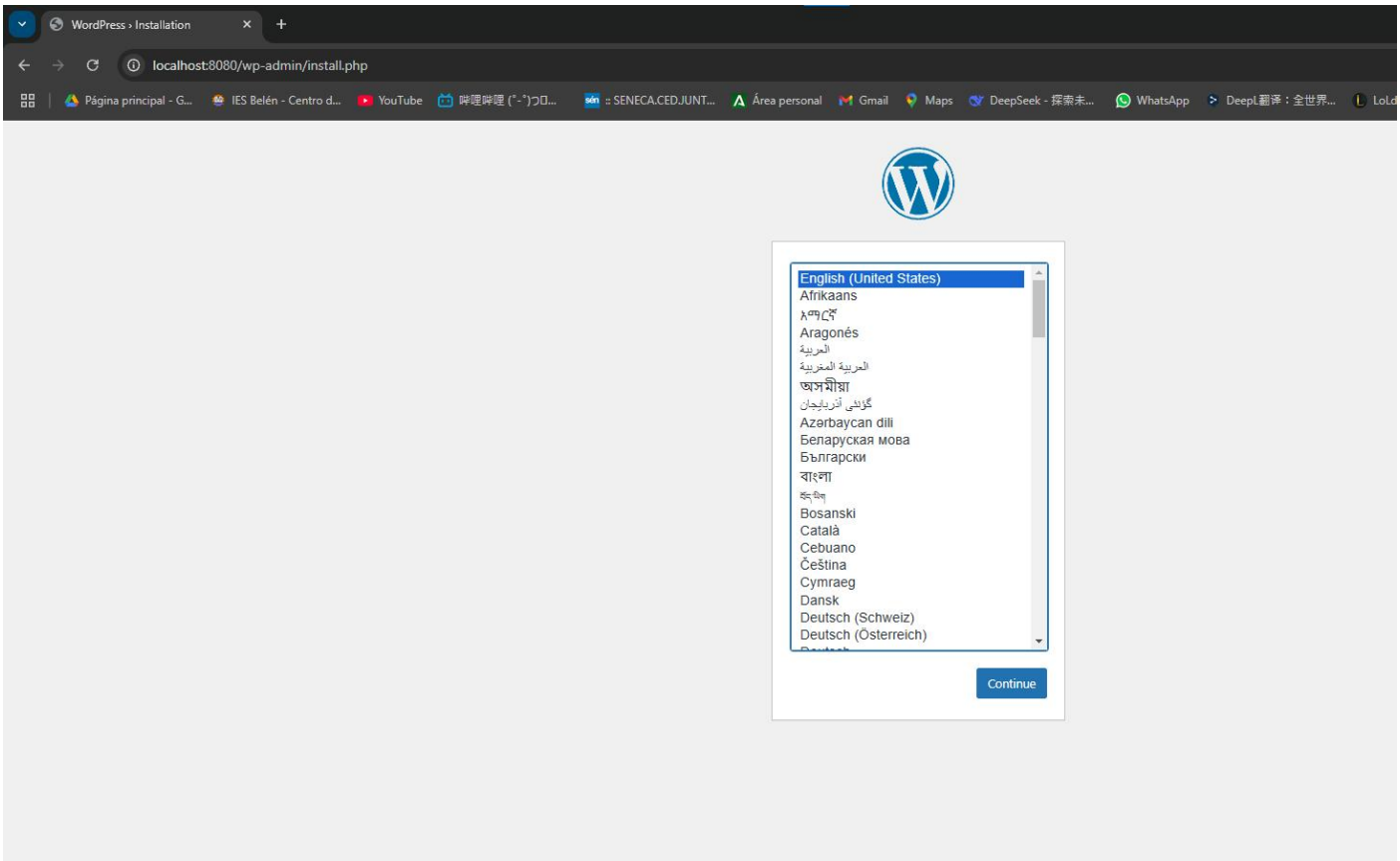
docker run -d --name mariadb --network wordpress-net -e MYSQL_ROOT_PASSWORD=root -e MYSQL_DATABASE=wordpress -e MYSQL_USER=wp_user -e MYSQL_PASSWORD=wp_pass mariadb



Crear el contenedor de WordPress

docker run -d --name wordpress --network wordpress-net -e WORDPRESS_DB_HOST=mariadb -e WORDPRESS_DB_USER=wp_user -e WORDPRESS_DB_PASSWORD=12345678 -e WORDPRESS_DB_NAME=wordpress -p 8080:80 wordpress





Deberás entregar los siguientes pantallazos en un documento pdf:

- Pantallazo donde se vea la creación del servidor de Mariadb
- Pantallazo donde se vea la creación del CMS WordPress con los parámetros de la BBD.
- Pantallazo del acceso al WordPress desde el navegador.