Practica 7:

1. Crear los directorios y ficheros necesarios para las imagenes:

root@docker:/# mkdir /var/lib/mysql
root@docker:/# mkdir /var/www/html/wp-content
root@docker:/# mkdir /usr/local/etc/haproxy

Una vez creados, creamos tambien el fichero de configuracion para el proxy y lo configuramos de la siguiente manera:

```
  GNU nano 5.4                    haproxy.cfg
global
        daemon
        maxconn 256
        user haproxy
        group haproxy
defaults
        mode http
        log global
        timeout connect 5000ms
        timeout client 5000ms
        timeout server 5000ms
fronted http-in
        bind *:80
        mode http
        stats enable
        default_backend contenedores_Wordpress
backend contenedores_Wordpress
        balance roundrobin
        server wp1
        server wp2
                          [ 20 líneas leídas ]
^G Ayuda       ^O Guardar     ^W Buscar      ^K Cortar      ^T Ejecutar
```

2. Creación de las redes necesarias:

```
root@docker:~# docker network ls
NETWORK ID     NAME      DRIVER    SCOPE
0eb0edce2841   bridge    bridge    local
4bae296a7a1b   host      host      local
463e4894981b   none      null      local
root@docker:~#
```

3. Crear el contenedor de la base de datos MariaDB root

```
root@docker:~# docker run -d --name mariaBD --network bridge -v /var/lib/mysql
 -e MARIADB_ROOT_PASSWORD=root -e MARIADB_USER=wordpress_container1 -e MARIADB
_PASSWORD=wp1 -e MARIADB_USER=wordpress_container2 -e MARIADB_PASSWORD=wp2 -e
WORDPRESS_DB_NAME=mariaBD wordpress
c6ea53fb0de72c655d1b2d23a222d4886bd8216063230411e2e5891d21d5810f
```

4. Creamos los contenedores usando las imagenes wordpress:

```
root@docker:~# docker run --name wordpress_c1 -v /var/html/wp-content --networ
k bridge -p 8081:80 -e WORDPRESS_DB_HOST=mariaDB -e WORDPRESS_DB_USER=wordpres
s_c1 -e WORDPRESS_DB_PASSWORD=wp1 -e WORDPRESS_DB_NAME=mariaDB wordpress
WordPress not found in /var/www/html - copying now...
Complete! WordPress has been successfully copied to /var/www/html
No 'wp-config.php' found in /var/www/html, but 'WORDPRESS_...' variables suppl
ied; copying 'wp-config-docker.php' (WORDPRESS_DB_HOST WORDPRESS_DB_NAME WORDP
RESS_DB_PASSWORD WORDPRESS_DB_USER)
AH00558: apache2: Could not reliably determine the server's fully qualified do
main name, using 172.17.0.4. Set the 'ServerName' directive globally to suppre
ss this message
AH00558: apache2: Could not reliably determine the server's fully qualified do
main name, using 172.17.0.4. Set the 'ServerName' directive globally to suppre
ss this message
[Sun Dec 24 11:17:30.393669 2023] [mpm_prefork:notice] [pid 1] AH00163: Apache
/2.4.57 (Debian) PHP/8.2.13 configured -- resuming normal operations
[Sun Dec 24 11:17:30.393734 2023] [core:notice] [pid 1] AH00094: Command line:
 'apache2 -D FOREGROUND'
```

```
root@docker:~# docker run --name wordpress_c2 -v /var/html/wp-content --networ
k bridge -p 8081:80 -e WORDPRESS_DB_HOST=mariaDB -e WORDPRESS_DB_USER=wordpres
s_c2 -e WORDPRESS_DB_PASSWORD=wp2 -e WORDPRESS_DB_NAME=mariaDB wordpress
WordPress not found in /var/www/html - copying now...
Complete! WordPress has been successfully copied to /var/www/html
No 'wp-config.php' found in /var/www/html, but 'WORDPRESS_...' variables suppl
ied; copying 'wp-config-docker.php' (WORDPRESS_DB_HOST WORDPRESS_DB_NAME WORDP
RESS_DB_PASSWORD WORDPRESS_DB_USER)
AH00558: apache2: Could not reliably determine the server's fully qualified do
main name, using 172.17.0.4. Set the 'ServerName' directive globally to suppre
ss this message
AH00558: apache2: Could not reliably determine the server's fully qualified do
main name, using 172.17.0.4. Set the 'ServerName' directive globally to suppre
ss this message
[Sun Dec 24 11:19:39.970680 2023] [mpm_prefork:notice] [pid 1] AH00163: Apache
/2.4.57 (Debian) PHP/8.2.13 configured -- resuming normal operations
[Sun Dec 24 11:19:39.972166 2023] [core:notice] [pid 1] AH00094: Command line:
 'apache2 -D FOREGROUND'
```

5. Creamos el proxy con la imagen HAPROXY

```
root@docker:~# docker run --name haProxy --network host -v /usr/local/etc/hapr
oxy/haproxy.cfg --sysctl net.ipv4.ip_unprivileged_port_start=0 haproxy
Unable to find image 'haproxy:latest' locally
latest: Pulling from library/haproxy
af107e978371: Already exists
a1f064ee9317: Pull complete
a689dfa22b58: Pull complete
f9b41e6a2385: Pull complete
8f081e03ba21: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:c915699e39d92dbb50d7f8e1267122406ffe4e74fe421d6aeee4b078b8367e7
6
```

6. Conectamos las redes que faltan

```
root@docker:~# docker network connect bridge wordpress_c1
root@docker:~# docker network connect bridge wordpress_c2
```