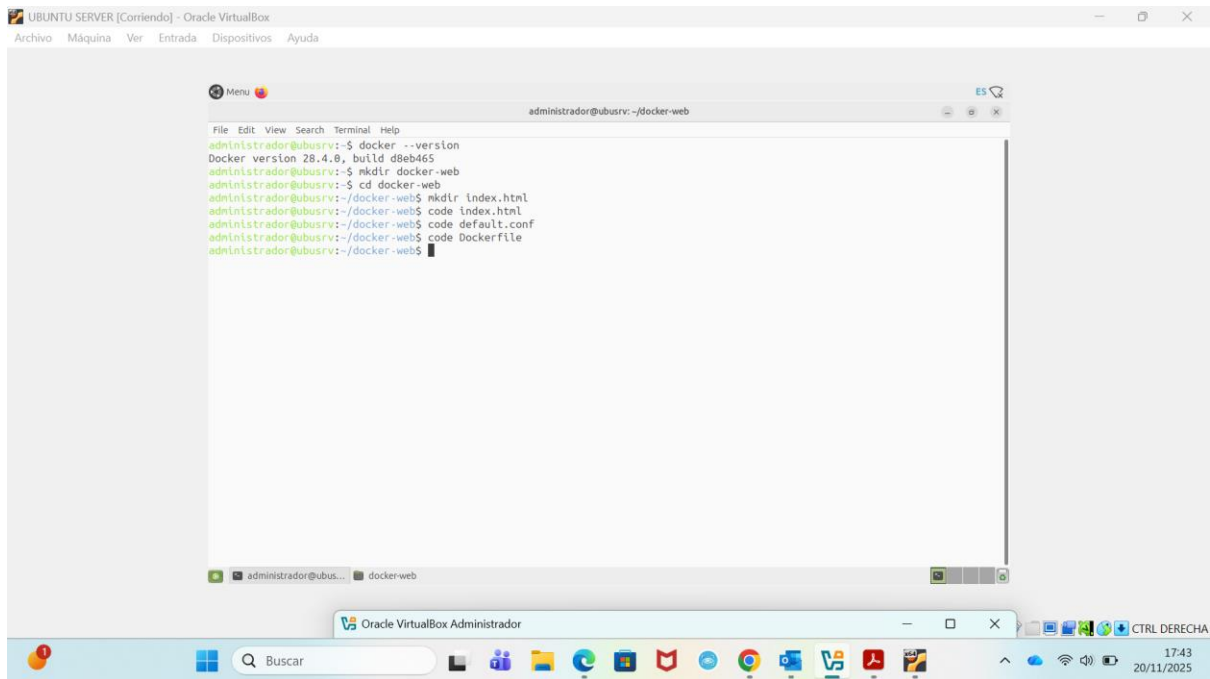


Despliegue de una Página Web con Docker

Parte 1: Configuración del entorno:

1. Crear un directorio para el proyecto

En tu terminal, crea un directorio llamado docker-web y accede a él:



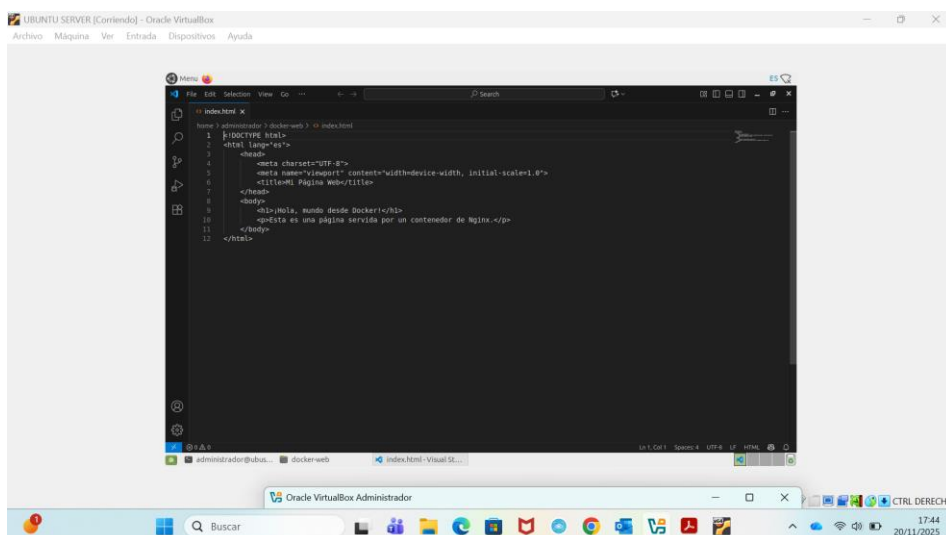
```
UBUNTU SERVER [Corriendo] - Oracle VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

administrador@ubusrv: ~/docker-web

File Edit View Search Terminal Help
administrador@ubusrv:~$ docker --version
Docker version 28.4.0, build d8eb465
administrador@ubusrv:~$ mkdir docker-web
administrador@ubusrv:~$ cd docker-web
administrador@ubusrv:~/docker-web$ mkdir index.html
administrador@ubusrv:~/docker-web$ code index.html
administrador@ubusrv:~/docker-web$ code default.conf
administrador@ubusrv:~/docker-web$ code Dockerfile
administrador@ubusrv:~/docker-web$
```

2. Crear una página web personalizada

Dentro del directorio docker-web, crea un archivo llamado index.html con el siguiente contenido:

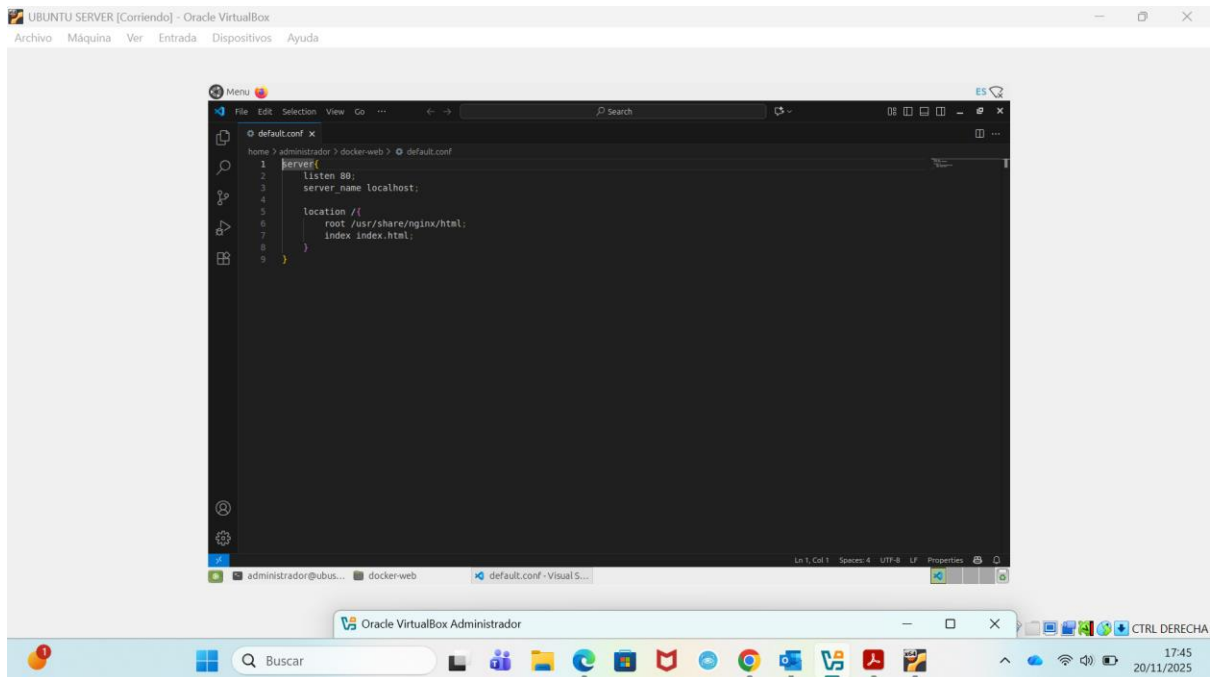


```
UBUNTU SERVER [Corriendo] - Oracle VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

index.html x
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>Mi Página Web</title>
7 </head>
8 <body>
9 <p>¡Hola, mundo desde Docker!</p>
10 <p>Esta es una página servida por un contenedor de Nginx.</p>
11 </body>
12 </html>
```

3. Crear un archivo de configuración para Nginx

Crea un archivo con en editor de texto llamado default.conf en el mismo directorio con la siguiente configuración básica para Nginx:

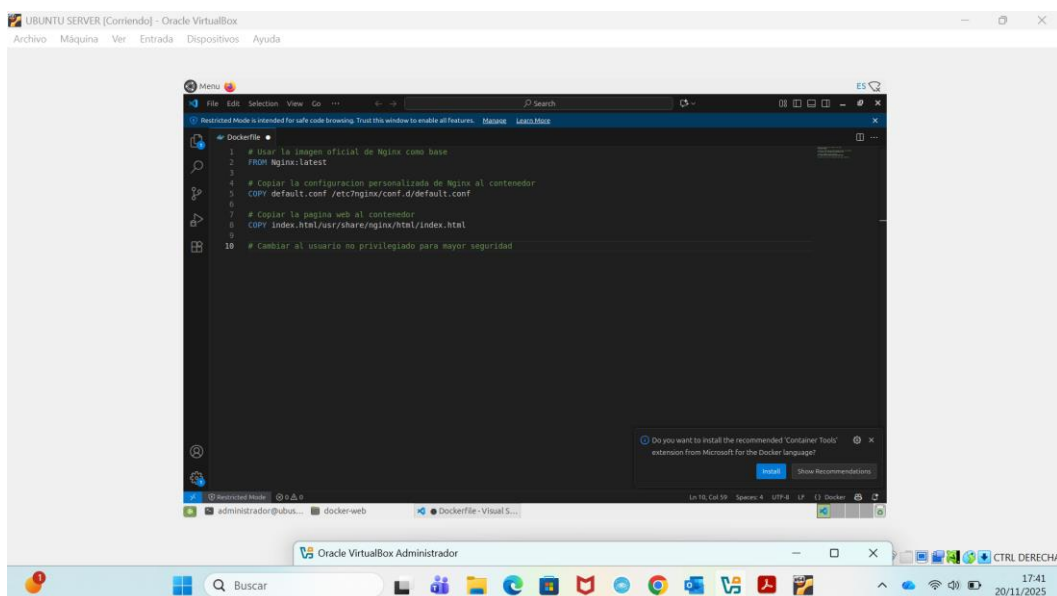


```
1 server {
2   listen 80;
3   server_name localhost;
4
5   location / {
6     root /usr/share/nginx/html;
7     index index.html;
8   }
9 }
```

Parte 2: Construcción del contenedor

1. Crear un archivo Dockerfile

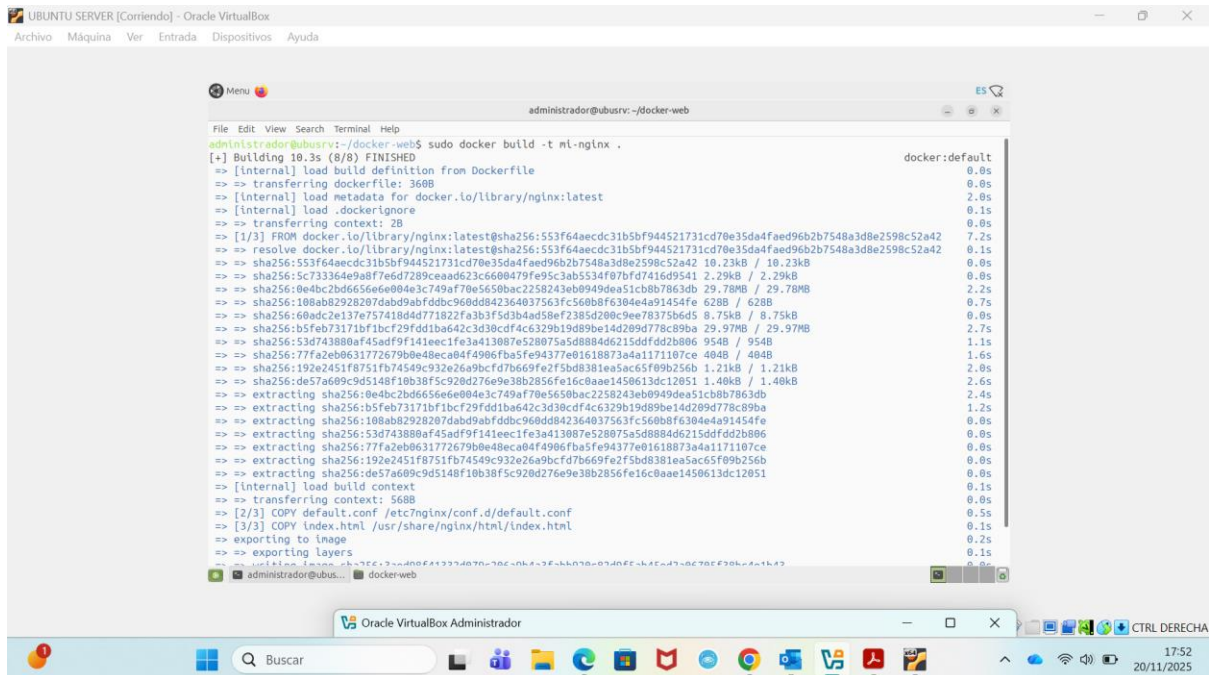
En el directorio docker-web, crea un archivo llamado Dockerfile sin extensión con el siguiente contenido:



```
1 # Usar la imagen oficial de Nginx como base
2 FROM nginx:latest
3
4 # Copiar la configuración personalizada de Nginx al contenedor
5 COPY default.conf /etc/nginx/conf.d/default.conf
6
7 # Copiar la pagina web al contenedor
8 COPY index.html /usr/share/nginx/html/index.html
9
10 # Cambiar al usuario no privilegiado para mayor seguridad
```

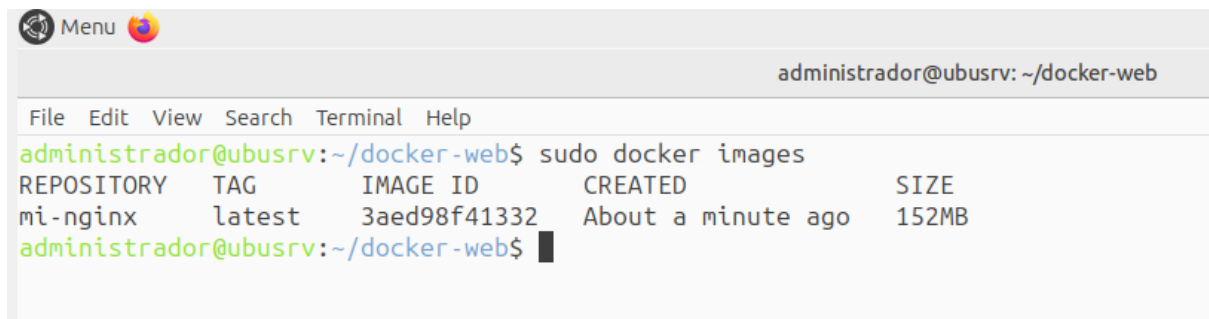
2. Construir la imagen personalizada

Construye la imagen utilizando el siguiente comando:



3. Verificar que la imagen se creó correctamente

Lista las imágenes disponibles en tu sistema:



Parte 3: Despliegue del servidor web 1.

1. Ejecutar el contenedor

Inicia un contenedor basado en la imagen que acabas de crear:

```
administrador@ubusrv:~/docker-web$ sudo docker run -d --name servidor-web -p 8080:80 mi-nginx
f8e908dea869177ada978200d623b6080c1077a015a80ce6b6ed302f8c7e373d
administrador@ubusrv:~/docker-web$
```

2. Verificar que el contenedor está corriendo

Comprueba los contenedores en ejecución:

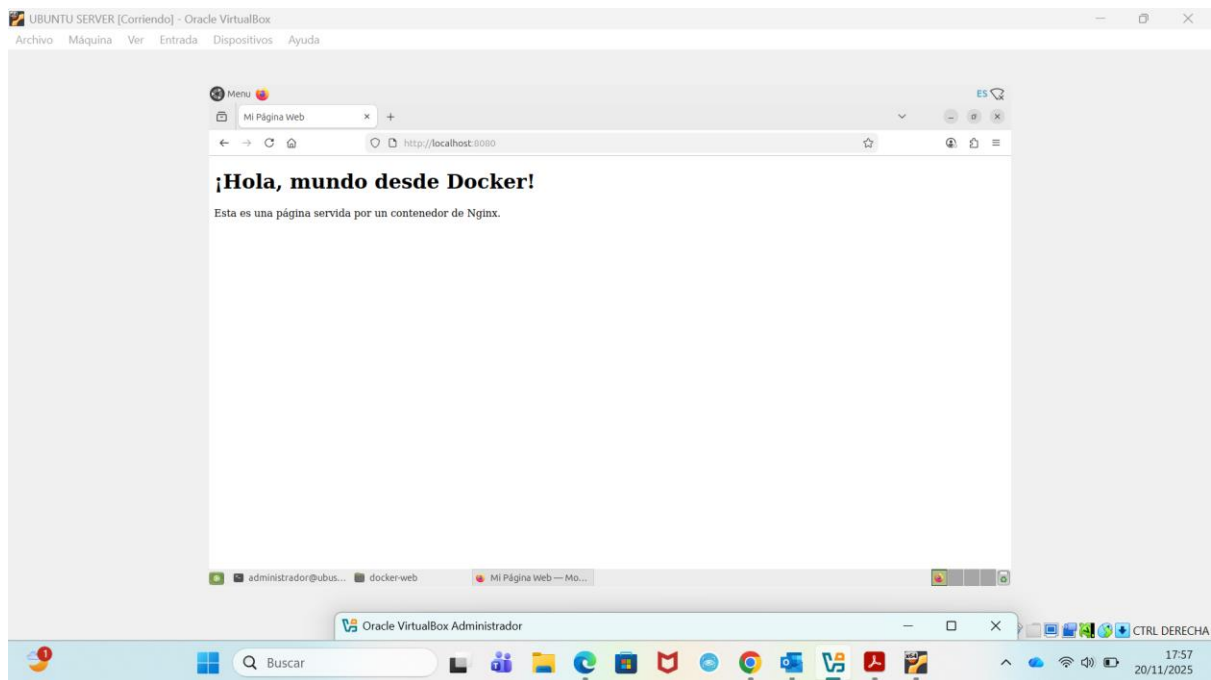
```
administrador@ubusrv:~/docker-web$ sudo docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
f8e908dea869	mi-nginx	"/docker-entrypoint..."	About a minute ago	Up About a minute	0.0.0.0:8080->80/tcp, [::]:8080->80/tcp

```
administrador@ubusrv:~/docker-web$
```

3. Probar la página web

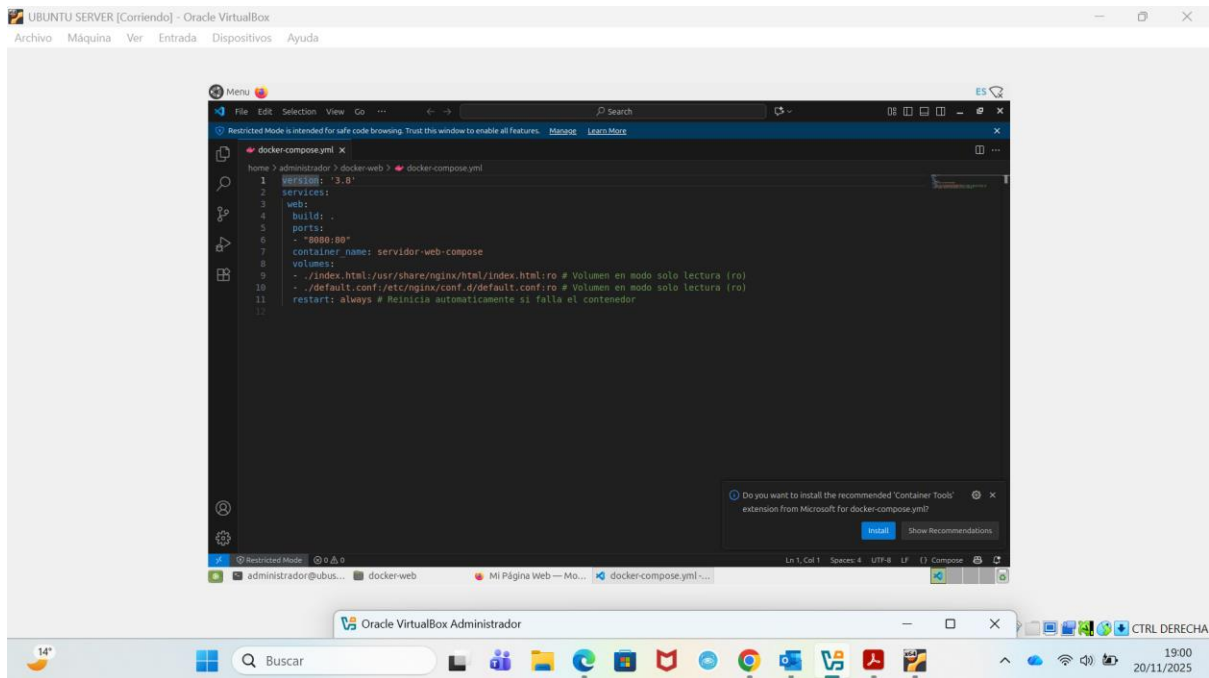
Abre tu navegador y visita <http://localhost:8080>. Deberías ver la página personalizada que creaste.



Parte 4: Uso de Docker Compose

1. Crear un archivo docker-compose.yml

En el directorio docker-web, crea un archivo llamado docker-compose.yml con el siguiente contenido:



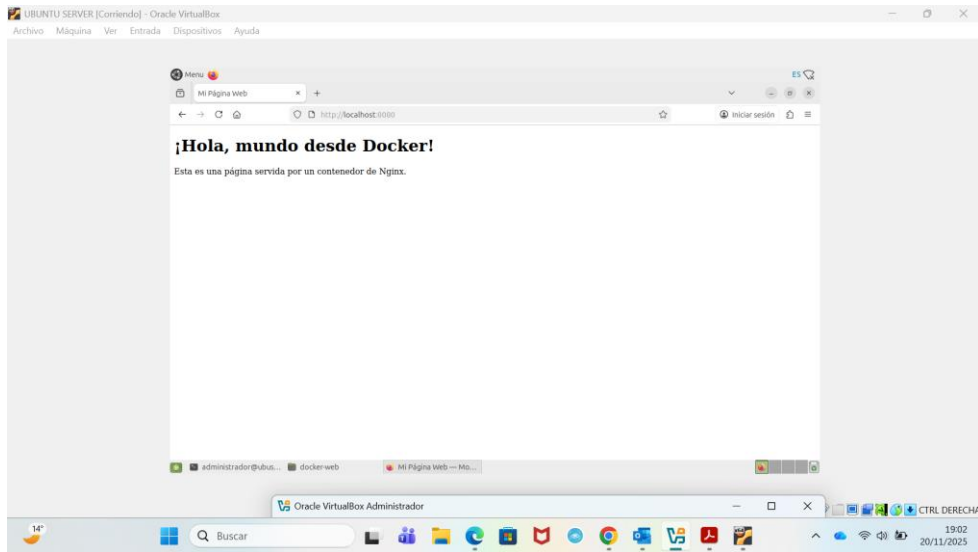
2. Iniciar los servicios con Docker Compose

Ejecuta el siguiente comando para iniciar el servicio definido en segundo plano (modo "detached"):

```
administrador@ubusrv:~/docker-web$ sudo docker-compose up -d
WARN[0000] /home/administrador/docker-web/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please
remove it to avoid potential confusion
[+] Running 1/1
✔ Container servidor-web-compose Started                                0.2s
administrador@ubusrv:~/docker-web$
```

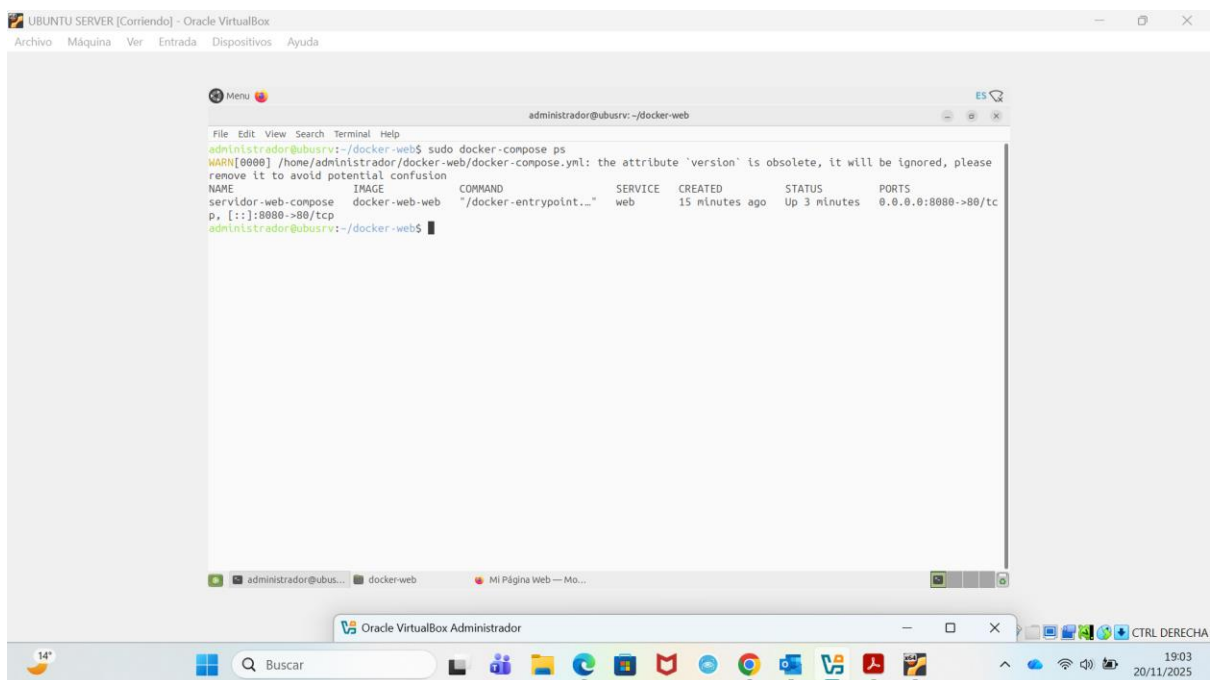
3. Comprobar que funciona

Visita nuevamente <http://localhost:8080> en tu navegador.



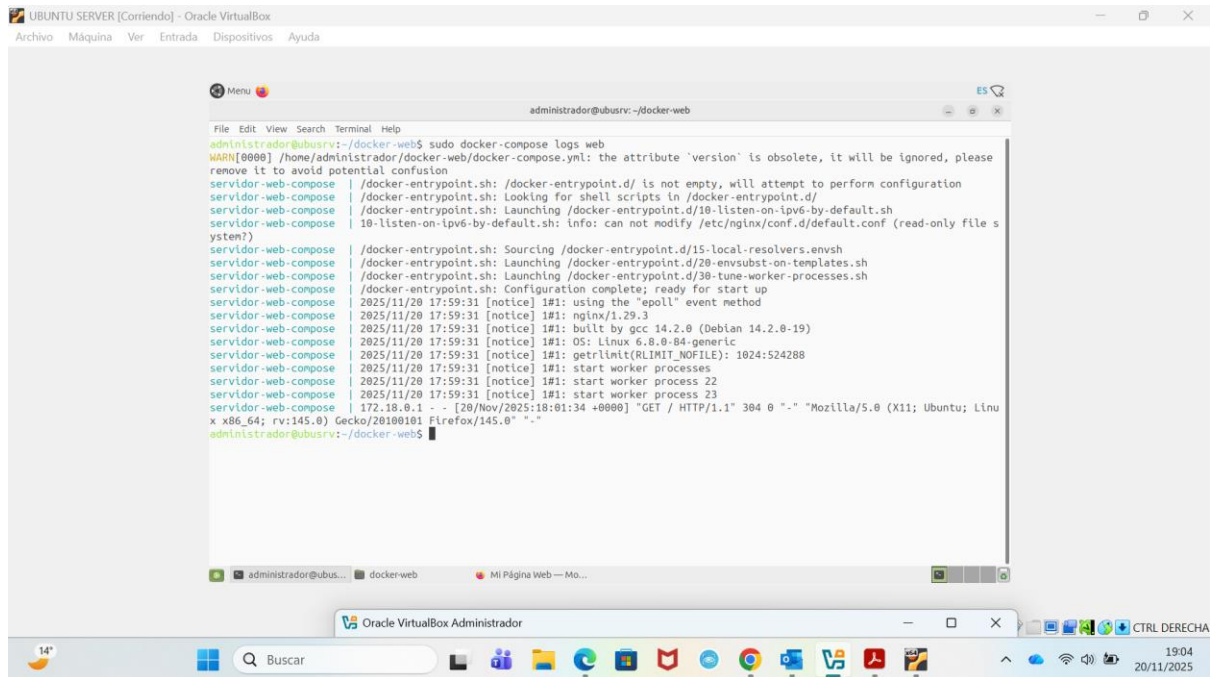
4. Verificar los servicios levantados

Comprueba los servicios activos con:



5. Ver los logs del servicio

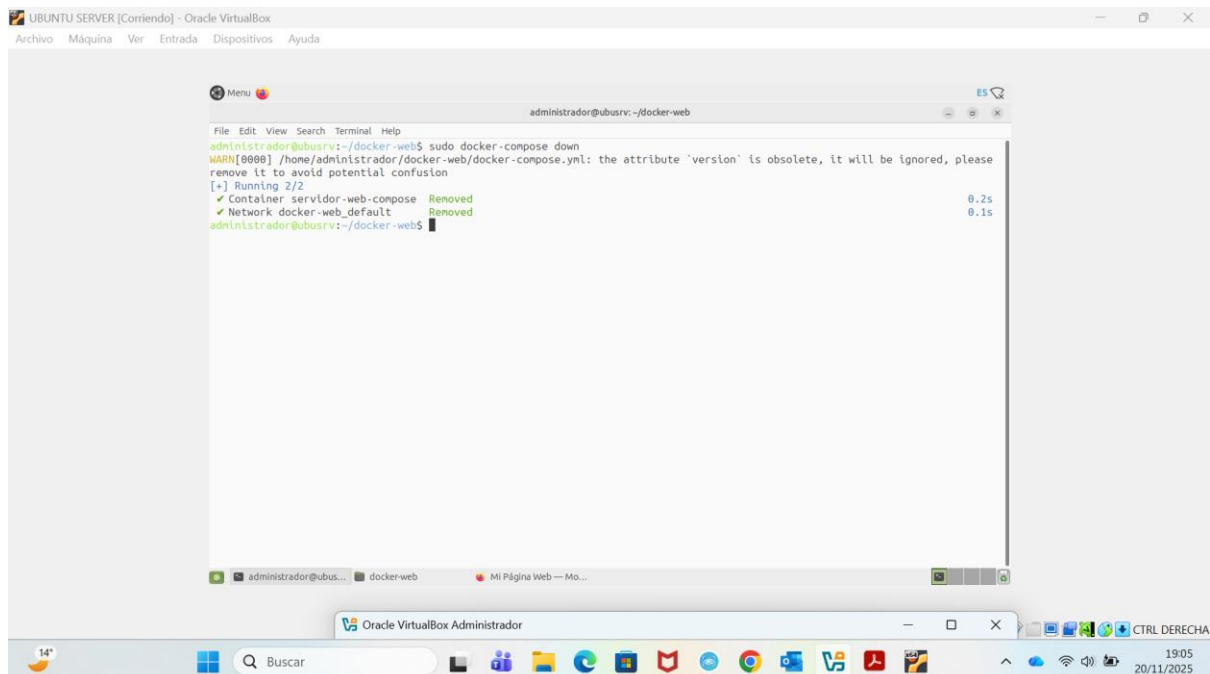
Muestra los registros generados por Nginx:



```
administrador@ubusrv:~/docker-web$ sudo docker-compose logs web
WARN[0000] /home/administrador/docker-web/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please
remove it to avoid potential confusion
servidor-web-compose | /docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
servidor-web-compose | /docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
servidor-web-compose | /docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
servidor-web-compose | 10-listen-on-ipv6-by-default.sh: info: can not modify /etc/nginx/conf.d/default.conf (read-only file s
ystem?)
servidor-web-compose | /docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
servidor-web-compose | /docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
servidor-web-compose | /docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
servidor-web-compose | /docker-entrypoint.sh: Configuration complete; ready for start up
servidor-web-compose | 2025/11/20 17:59:31 [notice] 1#1: using the "epoll" event method
servidor-web-compose | 2025/11/20 17:59:31 [notice] 1#1: nginx/1.29.3
servidor-web-compose | 2025/11/20 17:59:31 [notice] 1#1: built by gcc 14.2.0 (Debian 14.2.0-19)
servidor-web-compose | 2025/11/20 17:59:31 [notice] 1#1: OS: Linux 6.8.0-84-generic
servidor-web-compose | 2025/11/20 17:59:31 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1024:524288
servidor-web-compose | 2025/11/20 17:59:31 [notice] 1#1: start worker processes
servidor-web-compose | 2025/11/20 17:59:31 [notice] 1#1: start worker process 22
servidor-web-compose | 2025/11/20 17:59:31 [notice] 1#1: start worker process 23
servidor-web-compose | 172.18.0.1 - - [20/Nov/2025:18:01:34 +0000] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X11; Ubuntu; Linu
x x86_64; rv:145.0) Gecko/20100101 Firefox/145.0" "-"
administrador@ubusrv:~/docker-web$
```

6. Detener y eliminar los servicios

Cuando termines, detén y elimina los servicios creados con este comando:

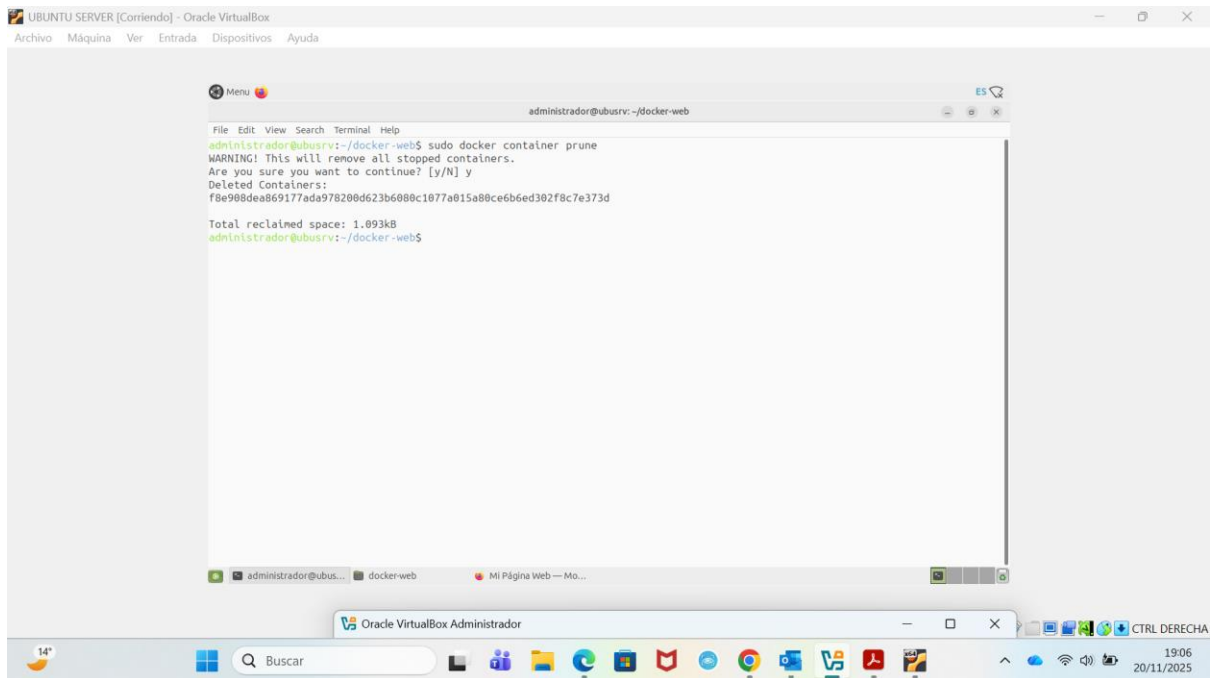


```
administrador@ubusrv:~/docker-web$ sudo docker-compose down
WARN[0000] /home/administrador/docker-web/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please
remove it to avoid potential confusion
[+] Running 2/2
  ✓ Container servidor-web-compose Removed 0.2s
  ✓ Network docker-web_default         Removed 0.1s
administrador@ubusrv:~/docker-web$
```

Parte 5: Limpieza

1. Eliminar todos los contenedores detenidos

Elimina todos los contenedores detenidos para liberar espacio: ¡OJO SI TIENES MAS CONTENEDORES NO LO HAGAS!



```
UBUNTU SERVER [Corriendo] - Oracle VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda

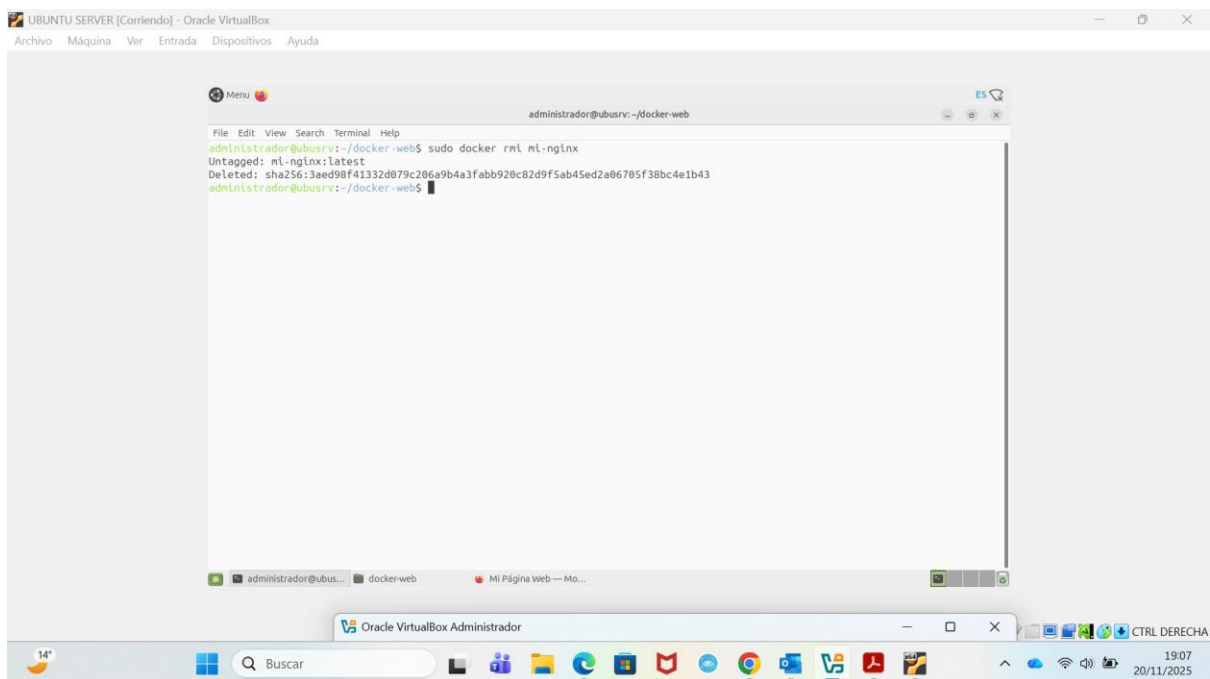
Menu
administrador@ubusrv: ~/docker-web

File Edit View Search Terminal Help
administrador@ubusrv:~/docker-web$ sudo docker container prune
WARNING! This will remove all stopped containers.
Are you sure you want to continue? [y/N] y
Deleted Containers:
f8e908dea869177ada978208d623b6080c1077a015a88ce6b6ed302f8c7e373d

Total reclaimed space: 1.093kB
administrador@ubusrv:~/docker-web$
```

2. Eliminar la imagen creada

Elimina la imagen personalizada si ya no es necesaria:



```
UBUNTU SERVER [Corriendo] - Oracle VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda

Menu
administrador@ubusrv: ~/docker-web

File Edit View Search Terminal Help
administrador@ubusrv:~/docker-web$ sudo docker rmi mi-nginx
Untagged: mi-nginx:latest
Deleted: sha256:3aed98f41332d079c206a9b4a3fabb920c82d9f5ab45ed2a06705f38bc4e1b43
administrador@ubusrv:~/docker-web$
```