

Evidencias taller volúmenes

1.

```
(julian@Julian)-[~]
$ mkdir web

(julian@Julian)-[~]
$ cd web

(julian@Julian)-[~/web]
$ echo "<h1>Hola desde bind mount</h1>" > ~/web/index.html

(julian@Julian)-[~/web]
$ docker run -d -name web-ro -p 8080:80 \ -v
~/web:/usr/share/nginx/html:ro \ nginx:alpine
unknown shorthand flag: 'n' in -name
See 'docker run --help'.
bash: /home/julian/web:/usr/share/nginx/html:ro: No such file or directory

(julian@Julian)-[~/web]
$ docker run -d --name web-ro -p 8080:80 \
-v /home/julian/web:/usr/share/nginx/html:ro \
nginx:alpine

Unable to find image 'nginx:alpine' locally
alpine: Pulling from library/nginx
9824c27679d3: Already exists
6bc572a340ec: Pull complete
403e3f251637: Pull complete
9adfbac99cb7: Pull complete
7a8a46741e18: Pull complete
c9ebe2ff2d2c: Pull complete
a992fbc61ecc: Pull complete
cb1ff4086f82: Pull complete
Digest: sha256:42a516af16b852e33b7682d5ef8acbd5d13fe08fecadc7ed98605ba5e3b26ab8
Status: Downloaded newer image for nginx:alpine
90c3885bd4b4c14ddd047cc7c0bfcd09fe125ce270344fb18c02875dd8ea2ba

(julian@Julian)-[~/web]
$
```



Hola desde bind mount

```
(julian@Julian)-[~/web]
$ echo "<h1>Cambiado desde host</h1>" > ~/web/index.html

(julian@Julian)-[~/web]
$
```

localhost:8080

Cambiado desde host

```
(julian@Julian)-[~/web]
$ docker exec -it web-ro sh -lc 'echo test > /usr/share/nginx/html/test.txt'
sh: can't create /usr/share/nginx/html/test.txt: Read-only file system

(julian@Julian)-[~/web]
$
```

2.

```
(julian@Julian)-[~]
$ mkdir postgres
(julian@Julian)-[~]
$ cd postgres
(julian@Julian)-[~/postgres]
$ docker volume create pgdata
pgdata
(julian@Julian)-[~/postgres]
$ docker run -d --name pg -e POSTGRES_PASSWORD=postgres -p 5432:5432 \
-v pgdata:/var/lib/postgresql/data \
postgres:16-alpine

Unable to find image 'postgres:16-alpine' locally
16-alpine: Pulling from library/postgres
9824c27679d3: Already exists
01ef787617d5: Pull complete
d444581c5dc1: Pull complete
127625cab66d: Pull complete
7f8bf47818a2: Pull complete
0951477387e1: Pull complete
878e28e3ecd5: Pull complete
d079e32a74cc: Pull complete
cb87d3c01966: Pull complete
40af0ccd9733: Pull complete
0b003ba20c51: Pull complete
Digest: sha256:8ffca822c1933bdc8be7dbbe9c2330974bdb43f5027f47717772fa35925412b0
Status: Downloaded newer image for postgres:16-alpine
61952a40f644019d923b199672818abb61af0cbbaaf2d54af7bc6db960c785775
docker: Error response from daemon: driver failed programming external connectivity on endpoint pg (a62e41fd5448e866e0bbaf8aa7d66dad48c04b59090a4f900a88ba987035b967): Bind for 0.0.0.0:5432 failed: port is already allocated.
```

```

(julian@Julian)-[~/postgre]
$ docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                               NAMES
90c3885bd4b4   nginx:alpine  "/docker-entrypoint..."  13 minutes ago  Up 13 minutes  0.0.0.0:8080->80/tcp, :::8080->80/tcp  web-ro
cb11b3b884c0   mi-postgres  "docker-entrypoint.s..."  9 days ago     Up 17 minutes  0.0.0.0:5432->5432/tcp, :::5432->5432/tcp  contenedor-postgres

(julian@Julian)-[~/postgre]
$ docker stop cb11b3b884c0

(julian@Julian)-[~/postgre]
$ docker run -d --name pg -e POSTGRES_PASSWORD=postgres -p 5432:5432 -v pgdata:/var/lib/postgresql/data postgres:16-alpine
docker: Error response from daemon: Conflict. The container name "/pg" is already in use by container "61952a40f644019d923b199672818abb61af0cbaaf2d54af7bc6db960c785775". You have to remove (or rename) that container to be able to reuse that name.
See 'docker run --help'.

(julian@Julian)-[~/postgre]
$ docker run -d --name pg1 -e POSTGRES_PASSWORD=postgres -p 5432:5432 -v pgdata:/var/lib/postgresql/data postgres:16-alpine
9fb097477f79eb401a2599e93d35d24f85e29cddd58eed8b634d39da9934c262

(julian@Julian)-[~/postgre]
$ docker exec -it pg1 psql -U postgres -c "CREATE TABLE
test(id serial, nombre text);"
docker exec -it pg1 psql -U postgres -c "INSERT INTO
test(nombre) VALUES ('Ada'),('Linus');"
docker exec -it pg1 psql -U postgres -c "SELECT * FROM
test;"
CREATE TABLE
INSERT 0 2
 id | nombre
----+-----
  1 | Ada
  2 | Linus
(2 rows)

(julian@Julian)-[~/postgre]
$ docker rm -f pg
pg

(julian@Julian)-[~/postgre]
$ docker rm -f pg1
pg1

(julian@Julian)-[~/postgre]
$ docker run -d --name pg1 -e POSTGRES_PASSWORD=postgres -p 5432:5432 \
-v pgdata:/var/lib/postgresql/data \
postgres:16-alpine
f684996d9ab61b73a745cbaf2ae67de3e017714f360167b098eda481b927167b

(julian@Julian)-[~/postgre]
$ docker exec -it pg1 psql -U postgres -c "SELECT * FROM test;"
 id | nombre
----+-----
  1 | Ada
  2 | Linus
(2 rows)

```

3.

```
(julian@Julian)-[~]
$ mkdir shared_logs

(julian@Julian)-[~]
$ cd shared_logs/

(julian@Julian)-[~/shared_logs]
$ docker volume create sharedlogs
sharedlogs

(julian@Julian)-[~/shared_logs]
$ docker run -d --name writer -v sharedlogs:/data \
  alpine:3.20 sh -c 'while true; do date > /data/log.txt;
  sleep 1; done'
unknown shorthand flag: 'n' in -name
See 'docker run --help'.

(julian@Julian)-[~/shared_logs]
$ docker run -d --name writer -v sharedlogs:/data \
  alpine:3.20 sh -c 'while true; do date > /data/log.txt; sleep 1; done'
Unable to find image 'alpine:3.20' locally
3.20: Pulling from library/alpine
01d036902a3c: Pull complete
Digest: sha256:b3119ef930faabb6b7b976780c0c7a9c1aa24d0c75e9179ac10e6bc9ac080d0d
Status: Downloaded newer image for alpine:3.20
9d9384a626a7a05d2e980ca6b0416688ff7f00301307ae518296e28355666948

(julian@Julian)-[~/shared_logs]
$ docker run -it --rm --name reader -v sharedlogs:/data \
  alpine:3.20 tail -f /data/log.txt
Wed Sep  3 21:37:11 UTC 2025
^C^C

(julian@Julian)-[~/shared_logs]
$ docker rm -f writer
writer

(julian@Julian)-[~/shared_logs]
$ docker run -d --name writer -v sharedlogs:/data \
  alpine:3.20 sh -c 'while true; do date > /data/log.txt; sleep 1; done'
9cfff715dc2fc7f241df7299c34d181fba4d4d774b3b63966a785c13e218d96e3

(julian@Julian)-[~/shared_logs]
$ docker run --rm -v sharedlogs:/data alpine:3.20 sh -lc 'tail -n 3 /data/log.txt'
Wed Sep  3 21:39:30 UTC 2025
```

4.

```
(julian@Julian)-[~/backup]
$ docker volume create appdata
appdata

(julian@Julian)-[~/backup]
$ docker run --rm -v appdata:/data alpine:3.20 sh -lc 'echo "backup-$(date +%F)" > /data/info.txt'

(julian@Julian)-[~/backup]
$ mkdir -p /home/julian/backups

(julian@Julian)-[~/backup]
$ docker run --rm -v appdata:/data:ro -v /home/julian/backups:/backup \
  alpine:3.20 sh -lc 'cd /data && tar czf /backup/appdata.tar.gz .'

(julian@Julian)-[~/backup]
$ docker volume create appdata_restored
appdata_restored

(julian@Julian)-[~/backup]
$ docker run --rm -v appdata_restored:/data -v /home/julian/backups:/backup \
  alpine:3.20 sh -lc 'cd /data && tar xzf /backup/appdata.tar.gz'

(julian@Julian)-[~/backup]
$ docker run --rm -v appdata_restored:/data alpine:3.20 cat /data/info.txt
backup-2025-09-03

(julian@Julian)-[~/backup]
$
```

Reflexión

Los ejercicios propuestos ayudaron a tener un concepto más claro de volúmenes en Docker, mostrando que incluso si se borra un contenedor, siempre que se levante sobre la misma imagen la persistencia estará ahí, esto es lo que más me importó aprender, pues aún no me quedaba muy claro la función exacta de los volúmenes, pero ya lo comprendo gracias al taller

Tuve problemas en 2 de los ejercicios con las rutas relativas, ya que simplemente intenté copiar los comandos proporcionados por el profesor y ver que ejecutaban, pero lógicamente no iban a funcionar, estos problemas fueron fáciles de solucionar en cuanto a que solo tuve que modificar la ruta del comando por la de mi maquina y ejecutarlo de nuevo.

Así mismo, en el ejercicio 2, tuve problemas con el puerto, pues ya estaba ocupado por un contenedor previo que había creado de Postgre para otro ejercicio, al cual le había indicado siempre estar corriendo a no ser de que se le indicara lo contrario, el problema fue fácil de solucionar simplemente parando este contenedor previo, pero al intentar levantar el contenedor de nuevo se presentó un nuevo problema ya que había usado el nombre “pg” para el primer intento, la solución fue muy simple, solo levanté un nuevo contenedor con nombre “pg1” y al momento de borrar el contenedor para probar el funcionamiento del volumen se borró ambos

Podría mencionar un pequeño problema con el ejercicio 3, y es que la ejecución del consumidor no se detuvo con “CTRL+C” a lo cual sencillamente abrí otra consola y seguí ejecutando los comandos posteriores