

# Docker Básico

---

**Arturo Silvelo**

Try New Roads

# Ejercicio: Volúmenes

## Ejercicio 1: Anonymous

1. Crea una nueva instancia de postgres con un volumen anónimo e inspecciona la información de los volúmenes.

Creamos el contenedor de postgres y lo asignamos a la red

```
docker run --name cdb -e POSTGRES_PASSWORD=12345678 --hostname course-database --network course-network -d postgres
```

Verificar la creación

```
docker ps  
docker inspect course-network
```

## 2. Inserta algunas entradas de ejemplo en la base de datos de postgres.

Comprobar que los datos se han guardado en la base de datos

```
docker exec -it cdb psql -U postgres -d postgres -c "SELECT * FROM task;"
```

### 3. Elimina el contenedor de postgres y crea uno nuevo.

Eliminamos la máquina en caliente, este paso puede hacer que el backend deje de funcionar, por lo que tenemos que reiniciar.

```
docker rm -f cdb  
docker restart cb
```

Creamos de nuevo la máquina

```
docker run --name cdb -e POSTGRES_PASSWORD=12345678 --hostname course-database --network course-network -d postgres
```

Repetimos el punto 2 para verificar que están los datos.

## Ejercicio 2: Named Volumes

---

1. Crea un nuevo contenedor de postgres usando un volumen de nombrado.

Creamos el volume

```
docker volume create postgres-data
```

Comprobar que los datos se han guardado en la base de datos

```
docker run --name cdb -e POSTGRES_PASSWORD=12345678 --hostname course-database --network course-network -d -v postgres-data:/var/lib/postgresql/data postgres
```

## 2. Inserta algunas entradas de ejemplo en la base de datos de postgres.

Comprobar que los datos se han guardado en la base de datos

```
docker exec -it cdb psql -U postgres -d postgres -c "SELECT * FROM task;"
```

### 3. Elimina el contenedor de postgres y crea uno nuevo.

Eliminamos el contenedor

```
docker rm -f cdb
```

Creamos otro de nuevo pero usando el mismo volume

```
docker run --name cdb -e POSTGRES_PASSWORD=12345678 --hostname course-database --network course-network -d -v postgres-data:/var/lib/postgresql/data postgres
```

Comprobar que los datos siguen

```
docker exec -it cdb psql -U postgres -d postgres -c "SELECT * FROM task;"
```



## Ejercicio 3: Bind Mounts

1. Haz una copia del volumen en un directorio local.

Copiamos el contenido de la base de datos a local.

```
docker cp cdb:/var/lib/postgresql/data .
```

Esto copia la carpeta de data del contenedor a una carpeta data en el host

## 2. Creamos una nueva máquina de postgres con un volumen de tipo `bind`.

Eliminamos el contenedor de la base de datos.

```
docker rm -f cdb
```

Creamos uno nuevo pero usando `bind` para montar el directorio copiado en el paso anterior

```
docker run --name cdb -e POSTGRES_PASSWORD=12345678 --hostname course-database --network course-network -d -v ./data:/var/lib/postgresql/data postgres
```

### 3. Verifica que ambas bases de datos tengan los mismos datos.

Es posible que necesitemos reiniciar el backend

```
docker restart cb
```

Comprobamos que los datos de la nueva base de datos sean los mismos.

```
docker exec -it cdb psql -U postgres -d postgres -c "SELECT \* FROM task;"
```

También podemos crear nuevas tareas y comprobar que se añaden