

## Ejercicios Docker DOC 3

### Vamos a trabajar con volúmenes docker:

1. Crea un volumen docker que se llame miweb.

```
docker volume create miweb
```

□

2. Crea un contenedor desde la imagen php:7.4-apache donde montes en el directorio/var/www/html (que sabemos que es el DocumentRoot del servidor que nos ofrece esaimagen) el volumen docker que has creado.

```
docker run -d --name miphp --mount type=volume,src=miweb,dst=/var/www/html -p 8080:80 php:7.4-apache
```

3. Utiliza el comando docker cp para copiar un fichero index.html en el directorio/var/www/html. □

```
docker cp ./index.html miphp:/var/www/html
```

🔗

4. Accede al contenedor desde el navegador para ver la información ofrecida por el fichero index.html.
5. Borra el contenedor

```
docker stop miphp
```

□

```
docker rm miphp
```

□

6. Crea un nuevo contenedor y monta el mismo volumen como en el ejercicio anterior.
7. Accede al contenedor desde el navegador para ver la información ofrecida por el ficheroindex.html. ¿Seguía existiendo ese fichero?

En principio sigue existiendo no habría ninguna variación o cambio.

### Vamos a trabajar con bind mount:

1. Crea un directorio en tu host y dentro crea un fichero index.html. □ □
2. Crea un contenedor desde la imagen php:7.4-apache donde montes en el directorio/var/www/html el directorio que has creado por medio de bind mount.
3. Accede al contenedor desde el navegador para ver la información ofrecida por el ficheroindex.html.

4. Modifica el contenido del fichero index.html en tu host y comprueba que al refrescar la página ofrecida por el contenedor, el contenido ha cambiado.

```
nano ejmiweb/index.html
```

□ □

5. Borra el contenedor □

```
Resultado: localhost:8080/index.html
```

□

6. Crea un nuevo contenedor y monta el mismo directorio como en el ejercicio anterior.

```
docker run -d --name miphpEjemplo2 -v /home/$USER/ejmiweb:/var/html -p 8080.80 php:7.4-apache
```

7. Accede al contenedor desde el navegador para ver la información ofrecida por el fichero index.html. ¿Se sigue viendo el mismo contenido?

Si, se seguirá viendo el mismo contenido

