docker-compose

related

- docker-compose casos de uso

comandos

- detached start:

```
docker-compose up -d [service]
```

- escalar servicio:

```
docker-compose up -d --scale <SERVICIO>=<NUM_CONT>
```

- comnprobar sintaxis `docker-compose.yml`:

```
docker compose config
```

estado servicio

```
docker compose ps
docker compose ps -a
docker compose ps --status=exited
```

reiniciar servicio

```
# reiniciar sin cambios:
docker compose restart [nombre_del_servicio]
# reconstruir si detecta cambios (y reiniciar servicios dependientes si no estan en marcha)
docker compose up -d [nombre_del_servicio]
# reconstruir forzando
docker compose up -d --build [nombre_del_servicio]
docker compose up -d --no-deps [nombre_del_servicio] # evitar inicio dependencias
docker compose up -d --force-recreate [nombre_del_servicio]
```

parar servicio

```
docker compose stop [nombre_del_servicio] # para pero no destruye el contenedor
docker compose down [nombre_del_servicio] # para y destruye el contenedor
docker compose down # para y destruye todos los contenedores y la red de la aplicación
```

variables

- maneras de definir variables para usar en el fichero **docker-compose.yml**
- están ordenadas de menos prioritaria a más prioritaria:
  1. se puede definir un fichero **.env** con variables para usar en el **docker-compose**: VARIABLE
  2. se pueden usar variables definidas en el shell dentro del **docker-compose** : `${VARIABLE}`
  3. se pueden pasar variables desde la línea de comandos: `docker run -e <variable>=<valor>` o `docker run --e <variable>=<valor>`
  4. se pueden especificar otros archivos de variables y pasarlas desde la línea de comandos: `docker run --env-file <fichero>`

.env

```
# variables de entorno preceden a estas
# variables pasadas por línea de comando preceden a estas
#   docker run --env <variable>=<valor>
# se pueden especificar otras variables en otros archivos
#   docker run --env-file <fichero>

MYSQL_ROOT_PASSWORD=toor
MYSQL_DATABASE=xxxx
MYSQL_USER=xxxx
MYSQL_PASSWORD=xxxx
MYSQL_STORAGE_PATH=/home/mate/Docker/.storage/mysql/xxxx
```

- para pasar variables a los contenedores, se puede:
  - usar la sección **environment**, donde definimos la variable (y podemos setear o hace que recoga el valor del shell)
  - usar la sección **env_file**, donde carga los valores desde un fichero externo
  - pasarlo por línea de comando con el parámetro **-e <VAR>[=<VALOR>]** y al igual que la seccción environment, podemos setearlo directamente o dejar que coja el valor que haya en el shell
- para ver el resultado de aplicar varias configuraciones de variables y que se obtiene el resultado que se desea, podemos usar: `docker-compose config` y nos mostrará el fichero con el resultado final una vez hecha la conversión de las diferentes variables
- https://docs.docker.com/compose/environment-variables/

From:
https://matebcn.eu/ - **miguel angel torres egea**

Permanent link:
https://matebcn.eu/tech:docker:docker-compose:start

Last update: **13/07/2026 12:00**