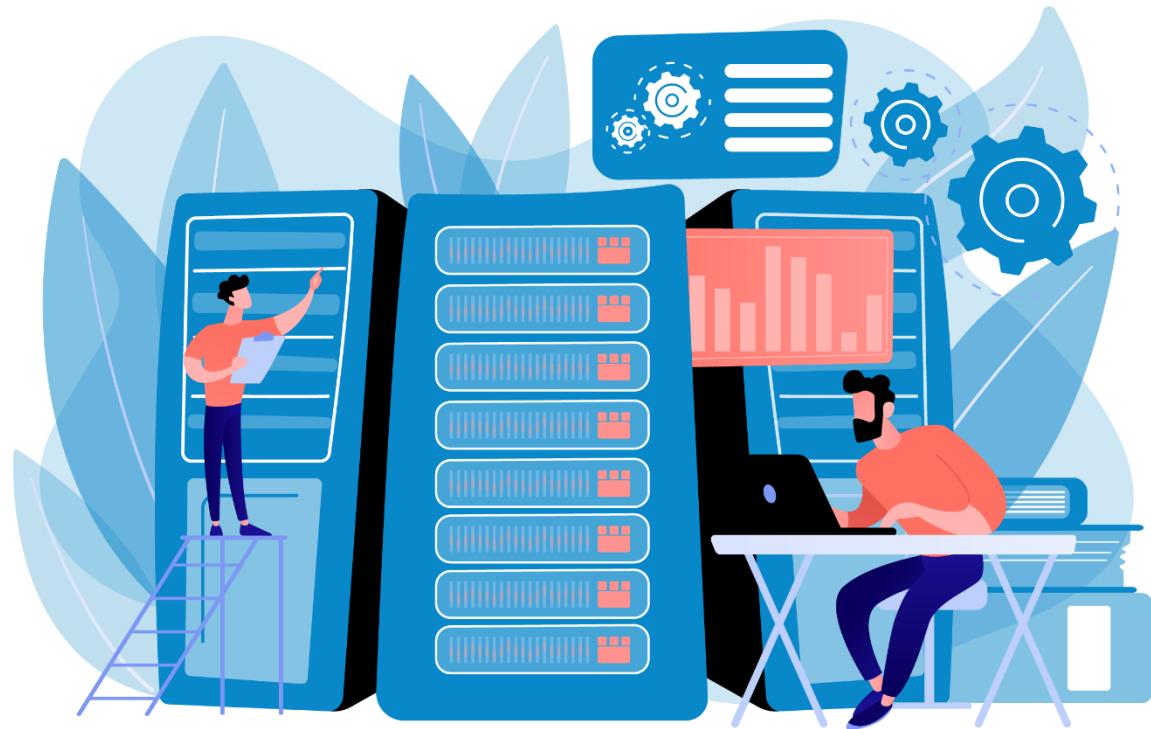


IES MONTE NARANCO

# PRÁCTICAS

DEVOPS Y CLOUD COMPUTING

---



---

# **UT - 3**

## **Ejercicio 1**

---

# Índice

1    *Uso de comandos Docker* \_\_\_\_\_ 4

---

# 1 Uso de comandos Docker

1. Descarga la versión 3.23 de la imagen alpine.

```
docker pull alpine:3.23
```

2. Crea un contenedor que ejecute la versión 3.22 de alpine dejando que Docker le asigne el nombre al contenedor y ejecutando un comando que escriba el mensaje "Ejecuto alpine:3.22"

```
docker run alpine:3.22 echo "Ejecuto alpine:3.22"
```

3. Muestra los contenedores en ejecución

```
docker ps
```

4. Muestra todos los contenedores incluso los parados

```
docker ps -a
```

5. Borra todos los contenedores parados con un solo comando

```
docker container prune -f
```

6. Crea un contenedor que se llame conte001 que use la imagen de alpine 3.23 que ejecute un comando que escriba por pantalla "Hola" y se ejecute en segundo plano.

```
docker run -d --name conte001 alpine:3.23 echo "Hola"
```

7. Borra el contenedor anterior usando su nombre

```
docker rm -f conte001
```

8. Ejecutar el comando **tail -f /dev/null** en un contenedor hace que se quede en estado de ejecución. Crea un contenedor que se llame conte005 que use la imagen alpine 3.23 que se ejecute en segundo plano y que se quede en estado de ejecución.

```
docker run -d --name conte005 alpine:3.23 tail -f /dev/null
```

9. Detén la ejecución del contenedor anterior sin borrar el contenedor.

```
docker stop conte005
```

---

10. Reanuda la ejecución del contenedor conte005

```
docker start conte005
```

11. Podemos borrar el contenedor anterior de dos formas:

Forma 1 (usando dos comandos):

```
docker stop conte005  
docker rm conte005
```

Forma2 (usando un solo comando):

```
docker rm -f conte005
```

12. Lista las imágenes descargadas.

```
docker images
```

13. Elimina la imagen alpine:3.23

```
docker rmi alpine:3.23
```

14. Crea un volumen que se llame miVolumen1

```
docker volume create miVolumen1
```

15. Crea un segundo volumen que se llame miVolumen2

```
docker volume create miVolumen2
```

16. Lista los volúmenes existentes

```
docker volume ls
```

17. Muestra información sobre miVolumen2

```
docker volume inspect miVolumen2
```

18. Borra el volumen miVolumen1

```
docker volume rm miVolumen1
```

- 
19. Crea un contenedor basado en la imagen alpine que se borre automáticamente al terminar de ejecutarse que se llame conte007 y como comando ejecute un echo con el mensaje "Mensaje de prueba"

```
docker run --rm --name conte007 alpine echo "Mensaje de prueba"
```

20. Modifica el comando anterior para que me monte un volumen llamado volumenDatos en la ruta /var/datos y escriba el mensaje "Mensaje de prueba" en el archivo datos.txt dentro del volumen.

```
docker run --rm --name conte007 -v volumenDatos:/var/datos alpine sh -c "echo 'Mensaje de prueba' > /var/datos/datos.txt"
```

21. Modifica el ejercicio anterior para que el comando que se ejecute sea mostrar el contenido del fichero datos.txt que está en el volumen volumenDatos

```
docker run --rm --name conte007 -v volumenDatos:/var/datos alpine cat /var/datos/datos.txt
```

22. Crea un contenedor basado en la imagen alpine que se borre automáticamente al terminar de ejecutarse que se llame conte008 y como comando ejecute un echo con el mensaje "Usando un bind mount"

```
docker run --rm --name conte008 alpine echo "Usando un bind mount"
```

23. Modifica el comando del ejercicio anterior para que usando un bind mount mapee la carpeta de nuestro equipo c:\temp\misDatos con la carpeta del contenedor /var/carpetaexterna y guarde el contenido del echo en un fichero llamado mensaje.txt en la carpeta del bind mount.

```
docker run --rm --name conte008 -v c:/temp/misDatos:/var/carpetaexterna alpine sh -c "echo 'Usando un bind mount' > /var/carpetaexterna/mensaje.txt"
```

24. Modifica el comando del ejercicio anterior para que muestre el contenido del fichero mensaje.txt

```
docker run --rm --name conte008 -v c:/temp/misDatos:/var/carpetaexterna alpine cat /var/carpetaexterna/mensaje.txt
```