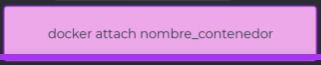
8. Docker: Comandos para manejar imágenes y contenedores

Introducción

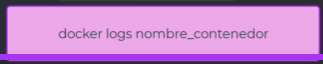


El -d asignamos el contendor en segundo plano en estado DETACHED y no bloquea el terminal

Para sacarlos de segundo plano seria con un attach



Y validar los logs de un contenedor



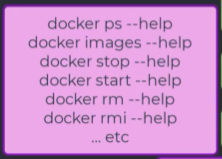
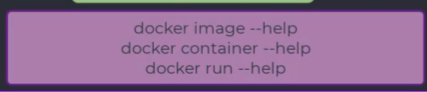
Ver el log y atacherlo para futuros logs



Ver todos los comandos Docker



Comando de ayuda y muy útiles



Reiniciando contenedores

Listar con ps y arrancar con star el -d sanemos que para dos estados DTACHED Y ATACH

Deteachet levanta los contendores por detrás sin necsita de usar el terminal principal

ATAch levanta el contenedor en el terminal principal

Si al levanta un contendor no lleva la condición -d en automatico se toca que es ATACH

 Conceptos y comandos para adjuntar (attach) o separar contenedores del terminal

Con el commando attach --> Attach local standard input, output, and error streams to a running container, detener el contendor con el stop y el nombre. El docker logs para validar todos los procesos que se han llevado a cabo en el container

Eliminar contenedores e imágenes

Con el comando stop detenemos el contendor que se esta ejecutando y para eliminarl0o sera con el Docker rm y agregando su id o el nombre del contendor, para eliminar todos los contenedores detenidos los podemos realizar con el comando container prune y pasamos la advertencia con Yes.

Eliminado imágenes, las listo primero Docker images y veo todo el detalle de las imágenes. Para elminar la imagen sera con el comando rmi y el id o el nombre de la imagen que se quiere eliminar.

Claramente para eliminar una imagen esta no debe estar en ejecución o tenga un container activoy y para eliminar todas la imágenes que no tiene contendores lo realizamos con el comando image prune

Eliminar contenedores detenidos automáticamente

Con el comando Docker run podemos habilitar una bandera en caso de eliminar uyn coantainer en cuanto se detenga su ejecución, para ello se realiza con el comando –rm

Ingresando en modo interactivo en contenedores

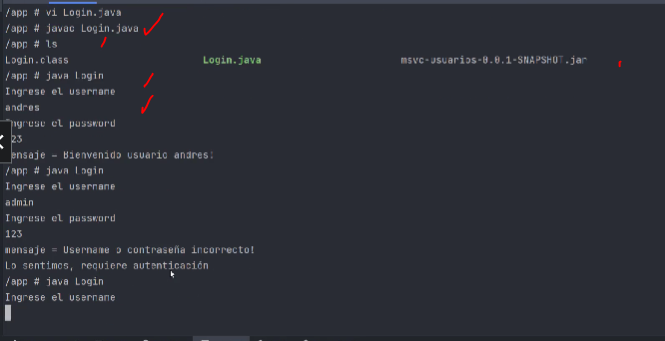
Podemos iniciar en el modo interactivo con el comando -it (interactive mode con el terminal) lo que hacemos entra al Shell del contendor. Para el Docker file el CMD nos permite sobre escribir la inrtruccion a ejecutar a diferencia del entrypoint, de este modo podemos administrar el SO de la terminal ejemplo en un container de DB puedo ingresar al Shell para validar las tablas y realizar las diferentes funciones de DB

 Copiando archivos hacia/desde el contenedor en ejecución

Permite copiar archivos de configuración o logs del contenedor a la maquina o local o viceversa, para ello entramos a modo interactivo, para ello vamos agregar un nuevo archivo (clase main LOGIN) al contendor, y lo realizamos desde la maquina local con el comando **Docker cp .\login.java nombreOIdContenedor:/rutacarpetaacopiar/login.java**



Y listo se copia el archivo al container y por medio del bin o el bash del container podemos validar y administrar dicho archivo copiado. En este ejemplo luego de coíad el archivo lo compilamos y podemos hacer uso de el.



* Copiando del contenedor al LOCAL, solo que renombramos el archivo a copiar



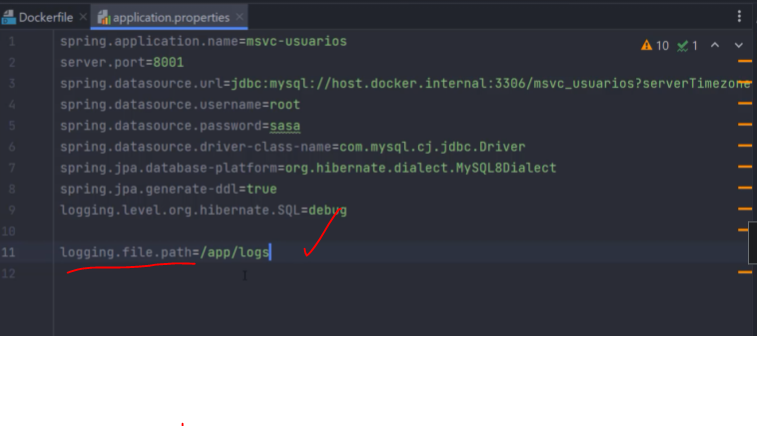
* Copiando toda la carpta APP en una nueva carpeta local llamada TEST



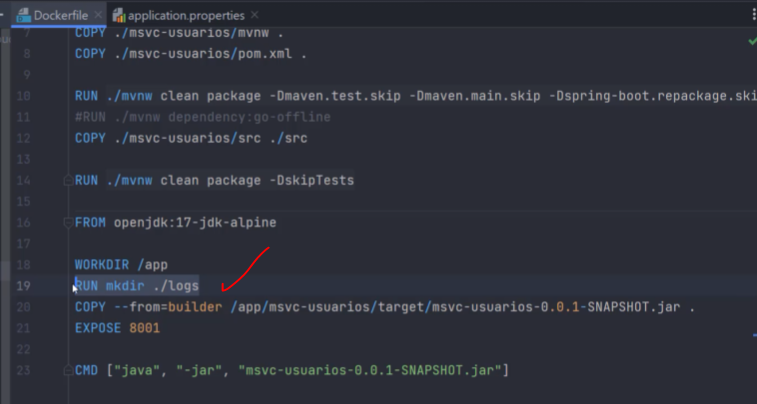
Copiando archivos logs de spring desde el contenedor

Copiando la traza de logs del contenedor al local

* Configurar la ruta de login.logs en el properties del proyecto spingBoot



* Ajustar el dockerfile para que ene automatico cuando se creem las capas de la imagen de una vez agregue la carpeta de logs



Al levantar nuevamente la imagen esta carpeta de logs ya debe tener su primera traza. Es muy importante y de buena practica administrar la traza de levantamiento de una imagen para cualquier proyecto, de ese modo ya podemos administrar los logs del container y copiarlos al local



Ya el archuvo de lopgs se puede validaren el bash por medio de una vim de consola que este se descarga previamente o en el local por medio del archivo y ya se abre con NotePaht

Más detalles sobre las imagenes y contenedores con el comando inspect

El comando inspect permite ver a detalle una image o un container seguido del id a verificar entre ellas el id completa su configuracion, variables de ambiente paht, arquitectura, las capas creadas en general.

Nombrando y etiquetando imagenes & contenedores

Nombre y etiqueta o version de una imagen, lo puedo validar desde dockerHub y toda la informacion de la imagen que se desea instalar, podemos ver la version el peso y los requisitos necesarios.

En este caso crear una imagen con su nombre y version



Esto es muy util en el caso de ir versionando nuestra dockerizacion según los avances de la aplicaacioin en los diferentes entornos ejemplo usuarios:desarrollo --- y asi susevivamente para tener un control de las version según el ambiente.

Para darrle nomre al conetedor lo podemos realizar por medio del la etiqueta —name