

GENERAR UNA IMAGEN DE CONTENEDOR

FROM: Especifica la imagen base en la que se basará tu imagen personalizada.

LABEL: Agrega metadatos a la imagen, como autor, descripción, versión, etc.

ENV: Establece variables de entorno que estarán disponibles en el contenedor en tiempo de ejecución.

RUN: Ejecuta comandos durante la construcción de la imagen. Se usa para instalar paquetes, configurar el entorno, etc.

COPY y ADD: Copian archivos y directorios desde el sistema de archivos local al sistema de archivos de la imagen.

WORKDIR: Establece el directorio de trabajo (directorio actual) para comandos posteriores en el Dockerfile.

EXPOSE: Expone puertos que el contenedor utilizará para la comunicación con el exterior.

VOLUME: Crea un punto de montaje (volumen) que permite persistir datos fuera del contenedor.

CMD y ENTRYPOINT: Especifican el comando predeterminado que se ejecutará cuando se inicie un contenedor basado en la imagen. **CMD** es reemplazable, mientras que **ENTRYPOINT** no lo es.

USER: Establece el usuario que se utilizará al ejecutar los comandos del contenedor.

HEALTHCHECK: Agrega un comando de verificación de salud para determinar si el contenedor está funcionando correctamente.

ARG: Define argumentos que pueden ser pasados en tiempo de compilación (por ejemplo, para personalizar la imagen).

ONBUILD: Define comandos que se ejecutarán en imágenes derivadas cuando se use esta imagen como base.

STOPSIGNAL: Especifica la señal que se utilizará para detener el contenedor.

SHELL: Configura el shell predeterminado utilizado por los comandos **RUN**.

DOCKERFILE

```
# Establece la imagen base
FROM ubuntu:20.04
# Agrega metadatos a la imagen
LABEL version="1.0" \
    description="Ejemplo de Dockerfile con varias instrucciones" \
    maintainer="Tu Nombre <tunombre@example.com>"
# Establece variables de entorno
ENV MI_VARIABLE="Valor por defecto"
# Ejecuta comandos durante la construcción
RUN apt-get update && apt-get install -y \
    paquete1 \
    paquete2 \
    && apt-get clean \
    && rm -rf /var/lib/apt/lists/*
# Copia archivos locales al sistema de archivos de la imagen
COPY archivo1.txt /ruta/en/el/contenedor/
COPY directorio_local/ /directorio/en/el/contenedor/
# Establece el directorio de trabajo
WORKDIR /directorio/trabajo/en/el/contenedor
# Expone un puerto
EXPOSE 8080
# Crea un punto de montaje (volumen)
VOLUME /datos
# Comando predeterminado
CMD ["mi_comando"]
# Agrega un comando de verificación de salud
HEALTHCHECK --interval=5s --timeout=3s \
    CMD curl -f http://localhost/ || exit 1
# Define un argumento
ARG MI_ARG="Valor predeterminado"
# Define comandos para ejecutar en imágenes derivadas
ONBUILD RUN echo "Comando en imagen derivada"
# Especifica el usuario para los comandos
USER usuario
# Establece la señal para detener el contenedor
STOPSIGNAL SIGTERM
# Configura el shell predeterminado
SHELL ["/bin/bash", "-c"]
# Comando final (útil solo en la última instrucción del Dockerfile)
CMD ["mi_comando_final"]
```