

1- Conceptos de Dockerfiles

Describir las instrucciones

FROM Indica la imagen padre sobre la que se está construyendo *"The FROM instruction initializes a new build stage and sets the Base Image for subsequent instructions"*

RUN Crea una capa del contenedor de escritura sobre la imagen especificada, y al iniciarse utiliza estos comandos *"The RUN instruction will execute any commands in a new layer on top of the current image and commit the results. The resulting committed image will be used for the next step in the Dockerfile."*

ADD Copia archivos/directorios al sistema de archivos del contenedor *"The ADD instruction copies new files, directories or remote file URLs from <src> and adds them to the filesystem of the image at the path <dest>."*

COPY A diferencia de add solamente copia archivos locales a la dirección indicada en el contenedor *"The COPY instruction copies new files or directories from <src> and adds them to the filesystem of the container at the path <dest>."*

EXPOSE Informa que la imagen creada por el Dockerfile escuchará en el puerto indicado cuando se corra el contenedor *"the EXPOSE instruction informs Docker that the container listens on the specified network ports at runtime. You can specify whether the port listens on TCP or UDP, and the default is TCP if the protocol is not specified. The EXPOSE instruction does not actually publish the port. It functions as a type of documentation between the person who builds the image and the person who runs the container, about which ports are intended to be published"*

CMD Comando CMD especifica la instrucción que va a ser ejecutada cuando el contenedor se inicialice *"There can only be one CMD instruction in a Dockerfile. If you list more than one CMD then only the last CMD will take effect. The main purpose of a CMD is to provide defaults for an executing container."*

ENTRYPOINT Setea ejecutables que siempre serán corridos cuando se inicialice el contenedor *"An ENTRYPOINT allows you to configure a container that will run as an executable."*

2- Generar imagen de docker

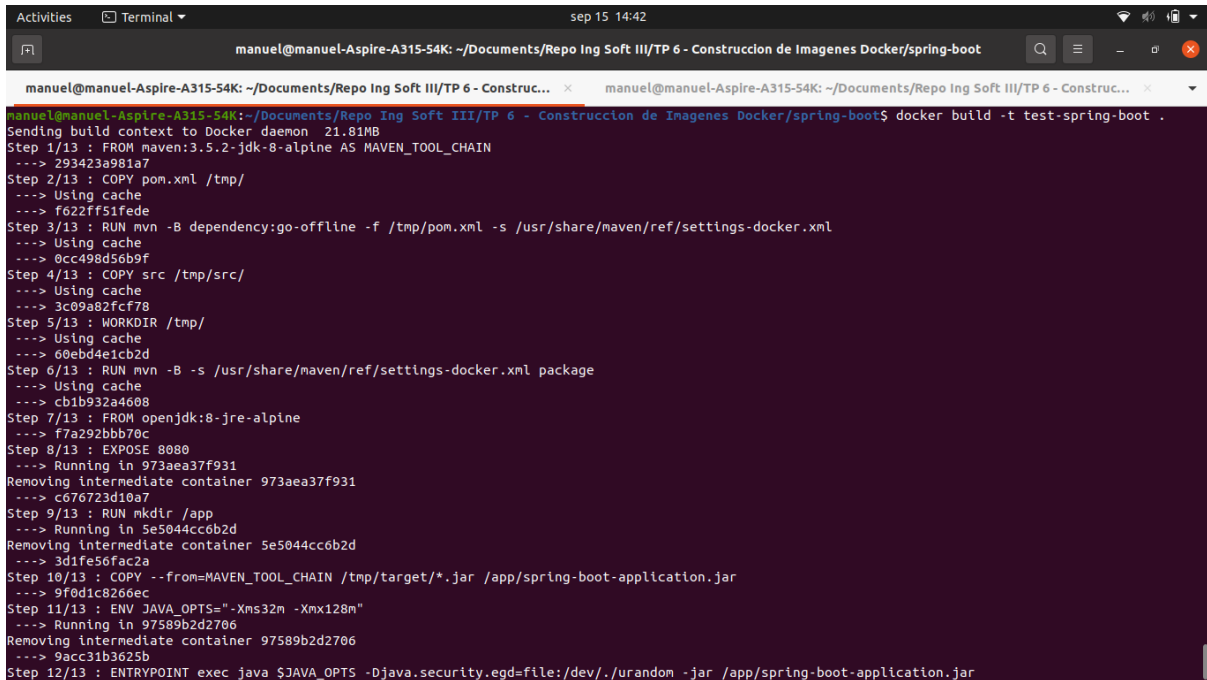
```
Activities Terminal
manuel@manuel-Aspire-A315-54K: ~/Documents/Repo Ing Soft III/TP 6 - Construccion de Imagenes Docker/spring-boot
manuel@manuel-Aspire-A315-54K:~/Documents/Repo Ing Soft III/TP 6 - Construccion de Imagenes Docker/spring-boot$ docker build -t test-spring-boot .
Sending build context to Docker daemon 21.81MB
Step 1/7 : FROM openjdk:8-jre-alpine
----> f7a292bbb70c
Step 2/7 : RUN apk add --no-cache bash
----> Running in d5fc2acd4e4a
fetch http://dl-cdn.alpinelinux.org/alpine/v3.9/main/x86_64/APKINDEX.tar.gz
fetch http://dl-cdn.alpinelinux.org/alpine/v3.9/community/x86_64/APKINDEX.tar.gz
(1/5) Installing ncurses-terminfo-base (6.1_p20190105-r0)
(2/5) Installing ncurses-terminfo (6.1_p20190105-r0)
(3/5) Installing ncurses-libs (6.1_p20190105-r0)
(4/5) Installing readline (7.0.003-r1)
(5/5) Installing bash (4.4.19-r1)
Executing bash-4.4.19-r1.post-install
Executing busybox-1.29.3-r10.trigger
OK: 93 MiB in 58 packages
Removing intermediate container d5fc2acd4e4a
----> d43acde4eca8
Step 3/7 : WORKDIR /app
----> Running in 6e21d1fb147a
Removing intermediate container 6e21d1fb147a
----> fa73a92ae2fc
Step 4/7 : COPY target/*.jar ./spring-boot-application.jar
----> cd3d678094eb
Step 5/7 : ENV JAVA_OPTS="-Xms32m -Xmx128m"
----> Running in e0d8f38ff531
Removing intermediate container e0d8f38ff531
----> 25556cf6056d
Step 6/7 : EXPOSE 8080
----> Running in c10da6069bc9
Removing intermediate container c10da6069bc9
----> 9ca035df0567
Step 7/7 : ENTRYPOINT exec java $JAVA_OPTS -Djava.security.egd=file:/dev/./urandom -jar spring-boot-application.jar
----> Running in dd36a0e20e7e
Removing intermediate container dd36a0e20e7e
----> ad3dd8316dbb
Successfully built ad3dd8316dbb
Successfully tagged test-spring-boot:latest
```

Ejecutar el contenedor

```
Activities Terminal sep 15 / 2027  
manuel@manuel-Aspire-A315-S4K: ~/Documents/Repo Ing Soft III/TP 6 - Construcion de Imagenes Docker/spring-boot  
  
manuel@manuel-Aspire-A315-S4K: ~/Documents/Repo Ing Soft III/TP 6 - Construc... x manuel@manuel-Aspire-A315-S4K: ~/Documents/Repo Ing Soft III/TP 6 - Construc...  
manuel@manuel-Aspire-A315-S4K:~/Documents/Repo Ing Soft III/TP 6 - Construcion de Imagenes Docker/spring-boot$ docker run -p 8080:8080 test-spring-bo  
ot  
===== Spring Boot =====  
:: Spring Boot :: (v2.0.2.RELEASE)  
  
2022-09-15 17:23:43.700 INFO 1 --- [main] s.actuator.SampleActuatorApplication : Starting SampleActuatorApplication v2.0.2 on 1c491c27  
1eF7 with PID 1 (/app/spring-boot-application.jar started by root in /app)  
2022-09-15 17:23:43.714 INFO 1 --- [main] s.actuator.SampleActuatorApplication : No active profile set, falling back to default profil  
es: default  
2022-09-15 17:23:43.844 INFO 1 --- [main] ConfigServletWebServerApplicationContext : Refreshing org.springframework.boot.web.servlet conte  
xt.AnnotationConfigServletWebServerApplicationContext: startup date [Thu Sep 15 17:23:43 GMT 2022]; root of context hierarchy  
2022-09-15 17:23:46.187 INFO 1 --- [main] transactionDelegatesBeanPostProcessorChecker : Bean 'org.springframework.transaction.annotation.Prox  
yTransactionManagementConfiguration' of type [org.springframework.transaction.annotation.ProxyTransactionManagementConfiguration$$EnhancerBySpringCGLI  
B$$5524ade89] is not eligible for getting processed by all BeanPostProcessors (for example: not eligible for auto-proxying)  
2022-09-15 17:23:46.811 INFO 1 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8080 (http)  
2022-09-15 17:23:46.854 INFO 1 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]  
2022-09-15 17:23:46.854 INFO 1 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet Engine: Apache Tomcat/9.5.31  
2022-09-15 17:23:46.869 INFO 1 --- [ost-startStop-1] o.a.catalina.core.AprLifecycleListener : The APR based Apache Tomcat Native library which allo  
ws optimal performance in production environments was not found on the java.library.path: [/usr/lib/jvm/java-1.8-openjdk/jre/lib/amd64/server:/usr/lib  
/jvm/java-1.8-openjdk/jre/lib/amd64:/usr/lib/amd64:/usr/packages/lib/amd64:/usr/lib64:/lib64:/lib:/usr/lib]  
2022-09-15 17:23:46.983 INFO 1 --- [ost-startStop-1] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext  
2022-09-15 17:23:46.984 INFO 1 --- [ost-startStop-1] o.s.web.context.ContextLoader : Root WebApplicationContext: initialization completed  
in 3140 ms  
2022-09-15 17:23:48.266 INFO 1 --- [ost-startStop-1] o.s.b.w.servlet.ServletRegistrationBean : Servlet dispatcherServlet mapped to [/]  
2022-09-15 17:23:48.273 INFO 1 --- [ost-startStop-1] o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'characterEncodingFilter' to: [*]  
2022-09-15 17:23:48.274 INFO 1 --- [ost-startStop-1] o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'hiddenHttpMethodFilter' to: [*]  
2022-09-15 17:23:48.274 INFO 1 --- [ost-startStop-1] o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'httpPutFormContentFilter' to: [*]  
2022-09-15 17:23:48.274 INFO 1 --- [ost-startStop-1] o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'requestContextFilter' to: [*]  
2022-09-15 17:23:48.275 INFO 1 --- [ost-startStop-1] o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'httpTraceFilter' to: [*]  
2022-09-15 17:23:48.275 INFO 1 --- [ost-startStop-1] o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'webMvcMetricsFilter' to: [/*]
```

```
Activities Terminal
sep 15 14:27
manuel@manuel-Aspire-A315-54K: ~/Documents/Repo Ing Soft III/TP 6 - Construcccion de Imagenes Docker/spring-boot
manuel@manuel-Aspire-A315-54K: ~/Documents/Repo Ing Soft III/TP 6 - Construcccion de Imagenes Docker/spring-boot$ curl -v localhost:8080
* Trying 127.0.0.1:8080...
* TCP_NODELAY set
* Connected to localhost (127.0.0.1) port 8080 (#0)
> GET / HTTP/1.1
> Host: localhost:8080
> User-Agent: curl/7.68.0
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200
< Content-Type: application/json;charset=UTF-8
< Transfer-Encoding: chunked
< Date: Thu, 15 Sep 2022 17:25:48 GMT
<
* Connection #0 to host localhost left intact
{"message":"Spring boot says hello from a Docker container"}manuel@manuel-Aspire-A315-54K:~/Documents/Repo Ing Soft III/TP 6 - Construcccion de Imagenes Docker/spring-boot$
```

3- Dockerfiles Multi Etapas



```
manuel@manuel-Aspire-A315-54K: ~/Documents/Repo Ing Soft III/TP 6 - Construcción de Imagenes Docker/spring-boot$ docker build -t test-spring-boot .
Sending build context to Docker daemon 21.81MB
Step 1/13 : FROM maven:3.5.2-jdk-8-alpine AS MAVEN_TOOL_CHAIN
--> 293423a981a7
Step 2/13 : COPY pom.xml /tmp/
--> Using cache
--> f622ff51fede
Step 3/13 : RUN mvn -B dependency:go-offline -f /tmp/pom.xml -s /usr/share/maven/ref/settings-docker.xml
--> Using cache
--> 0cc498d56b9f
Step 4/13 : COPY src /tmp/src/
--> Using cache
--> 3c09a82fcf78
Step 5/13 : WORKDIR /tmp/
--> Using cache
--> 60ebd4e1cb2d
Step 6/13 : RUN mvn -B -s /usr/share/maven/ref/settings-docker.xml package
--> Using cache
--> cb1b932a4608
Step 7/13 : FROM openjdk:8-jre-alpine
--> f7a292bbb70c
Step 8/13 : EXPOSE 8080
--> Running in 973aea37f931
Removing intermediate container 973aea37f931
--> c676723d10a7
Step 9/13 : RUN mkdir /app
--> Running in 5e5044cc6b2d
Removing intermediate container 5e5044cc6b2d
--> 3d1fe56fac2a
Step 10/13 : COPY --from=MAVEN_TOOL_CHAIN /tmp/target/*.jar /app/spring-boot-application.jar
--> 9f0d1c8266ec
Step 11/13 : ENV JAVA_OPTS="-Xms32m -Xmx128m"
--> Running in 97589b2d2706
Removing intermediate container 97589b2d2706
--> 9acc31b3625b
Step 12/13 : ENTRYPOINT exec java $JAVA_OPTS -Djava.security.egd=file:/dev/./urandom -jar /app/spring-boot-application.jar
```

Analizar y explicar el nuevo Dockerfile, incluyendo las nuevas instrucciones.

FROM maven:3.5.2-jdk-8-alpine AS MAVEN_TOOL_CHAIN

Trae una imagen de maven para trabajar sobre ella

COPY pom.xml /tmp/

Copia el archivo pom.xml a /tmp/

RUN mvn -B dependency:go-offline -f /tmp/pom.xml -s /usr/share/maven/ref/settings-docker.xml

Corre el comando mvn -B dependency:go-offline -f /tmp/pom.xml -s /usr/share/maven/ref/settings-docker.xml

COPY src /tmp/src/

Copia el archivo src a /tmp/src

WORKDIR /tmp/

Setea /tmp/ como directorio de trabajo del contenedor “The WORKDIR command is used to define the working directory of a Docker container at any given time. The

command is specified in the Dockerfile. Any RUN, CMD, ADD, COPY, or ENTRYPOINT command will be executed in the specified working directory."

RUN mvn -B -s /usr/share/maven/ref/settings-docker.xml package

Corre el comando mvn -B -s /usr/share/maven/ref/settings-docker.xml package

FROM java:8-jre-alpine

Trae una imagen de java para trabajar sobre ella

EXPOSE 8080

Indica que hay que exponer el puerto 8080

RUN mkdir /app

Corre el comando mkdir /app

COPY --from=MAVEN_TOOL_CHAIN /tmp/target/*.jar /app/spring-boot-application.jar

Copia --from=MAVEN_TOOL_CHAIN /tmp/target/*.jar a /app/spring-boot-application.jar

ENV JAVA_OPTS="-Xms32m -Xmx128m"

Setea variable de entorno JAVA_OPTS="-Xms32m -Xmx128m"

ENTRYPOINT exec java \$JAVA_OPTS -Djava.security.egd=file:/dev/./urandom -jar /app/spring-boot-application.jar

Cada vez que se corra el contenedor se ejecuta el comando java \$JAVA_OPTS -Djava.security.egd=file:/dev/./urandom -jar /app/spring-boot-application.jar

HEALTHCHECK --interval=1m --timeout=3s CMD wget -q -T 3 -s http://localhost:8080/actuator/health/ || exit 1

Realiza periódicamente chequeos de la salud del contenedor

4- Python Flask

¡Explicar que sucedió!

Al correr el contenedor con docker-compose up -d con la opción -d corre el proceso en el background, al llamar el docker compose, también se llama al dockerfile, que mediante las líneas

```
COPY ./requirements.txt /requirements.txt
```

```
RUN pip install -r /requirements.txt
```

Instala las dependencias mediante el archivo requirements.txt

¿Para qué está la key build.context en el docker-compose.yml?

El context define el path/directorio que contiene el dockerfile

5- Imagen para aplicación web en Nodejs

Archivo Dockerfile creado:

```
FROM node:16.17.0-alpine
RUN mkdir -p /usr/src/app
WORKDIR /usr/src/app
RUN npx create-react-app my-app
RUN cd my-app
WORKDIR /usr/src/app/my-app
RUN npm install
CMD ["npm","start"]
EXPOSE 3000
```

Comando para la creación de la imagen

```
docker build -t node-test .
```

```
docker run -p 3000:3000 node-test
```

6- Publicar la imagen en Docker Hub.

```
manuel@manuel-Aspire-A315-54K:~$ docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.
Username: manuelbobadilla
Password:
WARNING! Your password will be stored unencrypted in /home/manuel/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
manuel@manuel-Aspire-A315-54K:~$ docker tag node-test manuelbobadilla/node-test:latest
manuel@manuel-Aspire-A315-54K:~$ docker push manuelbobadilla/node-test:latest
The push refers to repository [docker.io/manuelbobadilla/node-test]
e686a063df71: Pushed
6bc31c2eb673: Pushed
9b4e76917ace: Pushed
f1ed0bba6314: Mounted from library/node
2808ff9120f2: Mounted from library/node
cb6eda6d73f0: Mounted from library/node
994393dc58e7: Mounted from library/node
latest: digest: sha256:4b353a65688372d80d6a4edf578009e8dc9d79b0b30ad1fac0c1dd89e5e92736 size: 1788
```


Activities Google Chrome sep 20 14:33

hub.docker.com/r/manuelbobadilla/node-test


Analisis PEST... Kali Linux Hist...

dockerhub Search for great content (e.g., mysql) Explore Repositories Organizations Help Upgrade manuelbobadilla

Explore manuelbobadilla/node-test

 **manuelbobadilla/node-test** ☆
By [manuelbobadilla](#) • Updated 2 minutes ago Manage Repository
Pulls 0

Overview Tags


No overview available
This repository doesn't have an overview

Docker Pull Command

```
docker pull manuelbobadilla/node-test
```