

Dockerizando la aplicación

Paso 1.- Descargar e instalar docker desde la pagina web: www.docker.com

Paso 2.- Comprobamos en la consola CMD si se ha instalado correctamente.

```
docker –version
```

Paso 3.- Generamos el .jar del microservicio que vamos a desplegar en Docker.
En la consola MS-DOS nos posicionamos dentro del proyecto y ejecutamos el siguiente comando:

```
.\mvnw clean package
```

Paso 4.- En la raíz del proyecto creamos el archivo Dockerfile (sin extensión y debe llamarse así) y ponemos el siguiente contenido:

```
FROM openjdk:8-jdk-alpine
VOLUME /tmp
EXPOSE 8002 (el puerto donde actualmente esta desplegado el microservicio)
ADD ./target/micro-servicio-pedidos-0.0.1-SNAPSHOT.jar nombre-abreviado.jar
ENTRYPOINT ["java", "-Djava.security.egd=file:/dev/./urandom", "-jar", "/nombre-abreviado.jar"]
```

Nombre del archivo resultante al crear el jar
nombre-abreviado.jar nombre alternativo, puede ser cualquiera

Paso 5.- Arrancamos Docker desktop para que el Daemon este escuchando y Creamos la imagen de Docker desde la raíz del proyecto, hay que estar ubicado en la raíz del proyecto:

```
docker build -t nombre-abreviado:v1 .
```

Ese punto final que no se olvide.
nombre-abreviado sin .jar
v1 es un sobrenombre que puede ser cualquier cosa

Paso 6.- Confirmamos que se ha generado la imagen correctamente

```
docker images
```

Paso 7.- Para que todos los microservicios se puedan comunicar entre si deben estar en la misma red, así que vamos a crear una red que llamaremos mired.

```
docker network create mired
```

Paso 8.- Generamos el contenedor a partir de la imagen:

```
docker run -p 8002:8002 –name nombre-abreviado –network mired nombre-abreviado:v1 (todo seguido)
```

--name es con doble guion
--network también con doble guion

Paso 9.- Se puede probar en el navegador:

Localhost:8002/.....