1 Instal·la i configura docker al teu ordinador/màquina virtual.

He utilitzat aquestes comandes mentres estaves explicant, son les comandes de instal·lació, i així si utilitzem la comanda docker mostrara la comanda

```
31  sudo apt -y install apt-transport-https ca-certificates curl gnupg2 software-properties-common
32  curl -fsSL https://download.docker.com/linux/debian/gpg | sudo apt-key add -
33  sudo add-apt-repository    "deb [arch=amd64] https://download.docker.com/linux/debian \
34    $(lsb_release -cs) \
35    stable"
36  cat /etc/apt/sources.list
37  apt update
38  sudo apt -y install docker-ce docker-ce-cli containerd.io
39  docker
```

Ara ens indicara que falta arguments, però podrem comprovar que funciona correctament:

```
root@server:/home/daw# docker

Usage:  docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Options:
      --config string         Location of clie
```

2 Comprova que funciona amb l'Hola Món de docker escrivint al terminal:

2.1 `sudo docker run hello-world`

```
root@server:/home/daw# docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:97a379f4f88575512824f3b352bc03cd75e239179eea0fecc38e597b2209f49a
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
```

2.2 Revisa https://docs.docker.com/get-started/ per entendre la comanda

Basicament crea un contenedor anomenat hello-world

3 Executa un interpret de comandes Bash de Linux:

sudo docker run -it ubuntu bash

```
root@server:/home/daw# docker run -it ubuntu bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
7c3b88808835: Downloading [=======================>                ]  13.55MB/28.57MB
```

4 Creació d'un docker amb un Hola Món llençat des de la comanda echo de l'interpret de comandes d'Ubuntu

mkdir hellodocker
cd hellodocker/
A dins crea un dockerfile amb el contingut:
FROM ubuntu:latest
CMD ["/bin/echo", "hola mon"]
cd ..
docker image build . -t holamon
docker container run helloworld

```
root@server:/home/daw/hellodocker# ls
dockerfile  holajava
root@server:/home/daw/hellodocker# cat dockerfile
FROM ubuntu:latest
CMD ["/bin/echo", "hola mon"]
root@server:/home/daw/hellodocker# cd ..
root@server:/home/daw# sudo docker image build . -t holamon
Sending build context to Docker daemon  11.78kB
Step 1/2 : FROM ubuntu:latest
 ---> 2b4cba85892a
Step 2/2 : CMD ["/bin/echo", "hola mon"]
 ---> Using cache
 ---> 46aefa43e473
Successfully built 46aefa43e473
Successfully tagged holamon:latest
root@server:/home/daw# sudo docker container run holamon
hola mon
root@server:/home/daw# 
```

5 Executa un interpret de comandes openjdk de Java:

5.1 `sudo docker container run -it openjdk`

```
root@server:/home/daw/hellodocker# docker container run -it openjdk
Unable to find image 'openjdk:latest' locally
latest: Pulling from library/openjdk
a2af32d411b4: Pull complete
ab18c3f6beb4: Pull complete
ef920c90c154: Pull complete
Digest: sha256:2c3ffd17a95947a187cbf00b454656566063cc53f21ea419929c0d23f3347572
Status: Downloaded newer image for openjdk:latest
Mar 04, 2022 11:16:11 AM java.util.prefs.FileSystemPreferences$1 run
INFO: Created user preferences directory.
|  Welcome to JShell -- Version 17.0.2
|  For an introduction type: /help intro

jshell> ▮
```

5.2 Pots sortir amb la comanda /exit

```
jshell> /exit
|  Goodbye
root@server:/home/daw/hellodocker# ▮
```

6 Creació d'un docker amb un Hola Món en Java llençat des de l'openjdk

6.1 `mkdir holajava`

```
root@server:/home/daw# ls
dockerfile  holajava  HolaMon.jar
root@server:/home/daw# ▮
```

6.2 Crea un fitxer java amb (per exemple un HolaMon.java) o qualsevol altre

```
  GNU nano 5.4                          holajava/HolaMon.java
public class HolaMon {
public static void main(String[] args){
System.out.println(" Hola mon en Java ManelCC▮");
}
}
```

6.3 Compila i genera el .jar

Pero per utilitzar el java s'ha de instalar:

```
root@server:/home/daw/holajava# apt install default-jre
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  alsa-topology-conf alsa-ucm-conf at-spi2-core ca-certificates-java
```

```
root@server:/home/daw/holajava# apt install default-jdk
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  default-jdk-headless libice-dev libpthread-stubs0-dev libsm-dev
  libx11-dev libxau-dev libxcb1-dev libxdmcp-dev libxt-dev
  openjdk-11-jdk openjdk-11-jdk-headless x11proto-dev
  xorg-sgml-doctools xtrans-dev
Paquetes sugeridos:
```

I ara per compilar executarem aquestes comandes:

```
root@server:/home/daw/holajava# javac -d . HolaMon.java
root@server:/home/daw/holajava# ls
HolaMon.class  HolaMon.java
root@server:/home/daw/holajava# jar cvf HolaMon.jar HolaMon.class
manifiesto agregado
agregando: HolaMon.class(entrada = 444) (salida = 305)(desinflado 31%)
root@server:/home/daw/holajava# ls
HolaMon.class  HolaMon.jar  HolaMon.java
root@server:/home/daw/holajava#
```

6.4 Genera el docker file amb l'estructura semblant a:
```
FROM openjdk:latest
COPY HolaJava.jar HolaMon.jar
CMD java -jar /usr/src/HolaMon.jar
```

el copy esta mal y necesitaremos un archivo que especifique la clase:

```
  GNU nano 5.4                          dockerfile
FROM openjdk:latest
COPY HolaMon.jar HolaMon.jar
CMD java -jar /usr/src/HolaMon.jar
```

```
  GNU nano 5.4                          manifest.txt
Main-Class: HolaMon
```

6.5 Inicializar el java:

```
cd ..
docker image build -t holajava:l
docker container run holajava:1
```

```
root@server:/home/daw/holajava# docker image build . -t holajava:1
Sending build context to Docker daemon  6.656kB
Step 1/3 : FROM openjdk:latest
 ---> 2f765403878f
Step 2/3 : COPY HolaMon.jar HolaMon.jar
 ---> 00ea18105345
Step 3/3 : CMD java -jar /usr/src/HolaMon.jar
 ---> Running in 46ecdb9d73e1
Removing intermediate container 46ecdb9d73e1
 ---> 640f24666830
Successfully built 640f24666830
Successfully tagged holajava:1
root@server:/home/daw/holajava#
```

```
root@server:/home/daw# docker container run holajava:1
Hola mon en Java ManelCC
root@server:/home/daw#
```

7 Llista els diferents contenidors, espai, que utilitzen...

(Hi ha el de holajava en separat ja que he tingut que reinicialitzar la maquina de 0)

```
root@server:/home/daw/hellodocker# docker ps -a
CONTAINER ID   IMAGE         COMMAND                  CREATED          STATUS                     PORTS
56e2f7b2870e   openjdk       "jshell"                 2 minutes ago    Exited (0) About a minute ago
041c94ea246e   ubuntu        "bash"                   24 minutes ago   Exited (127) 8 minutes ago
7a4e9201a099   hello-world   "/hello"                 49 minutes ago   Exited (0) 49 minutes ago
9b9f0e9fd9a2   wordpress     "docker-entrypoint.s…"   About an hour ago Up About an hour          0.0.0.0:8087->80/tcp, :::8087->80/tcp
2d651262fad1   nginx         "/docker-entrypoint.…"   About an hour ago Up About an hour          0.0.0.0:8086->80/tcp, :::8086->80/tcp
58d2d07eace5   httpd         "httpd-foreground"       2 hours ago      Up 2 hours                 0.0.0.0:8085->80/tcp, :::8085->80/tcp
root@server:/home/daw/hellodocker#
```

```
e00b66495a11   holajava:1    "/bin/sh -c 'java -j…"   12 minutes ago   Exited (1) 12 minutes ago                                         musing_
```

8 Afegeix un mínim de3 contenidors diferents als creats anteriorment (altres llenguatges, tipus...)

(ja estava fet previament en classe)

```
root@server:/home/daw/hellodocker# docker ps -a
CONTAINER ID   IMAGE         COMMAND                  CREATED          STATUS                     PORTS
56e2f7b2870e   openjdk       "jshell"                 2 minutes ago    Exited (0) About a minute ago
041c94ea246e   ubuntu        "bash"                   24 minutes ago   Exited (127) 8 minutes ago
7a4e9201a099   hello-world   "/hello"                 49 minutes ago   Exited (0) 49 minutes ago
9b9f0e9fd9a2   wordpress     "docker-entrypoint.s…"   About an hour ago Up About an hour          0.0.0.0:8087->80/tcp, :::8087->80/tcp
2d651262fad1   nginx         "/docker-entrypoint.…"   About an hour ago Up About an hour          0.0.0.0:8086->80/tcp, :::8086->80/tcp
58d2d07eace5   httpd         "httpd-foreground"       2 hours ago      Up 2 hours                 0.0.0.0:8085->80/tcp, :::8085->80/tcp
root@server:/home/daw/hellodocker#
```

```
e00b66495a11   holajava:1    "/bin/sh -c 'java -j…"   12 minutes ago   Exited (1) 12 minutes ago                                         musing_
```

Afegits worpress, nginx i httpd