Job 1

docker run hello-world

```
user@SOUFIANE:~$ docker run hello-world

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 Do
cker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image wh
ich runs the
    executable that produces the output you are currently readin
g.
 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
:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/
```

Job 2

Créez un répertoire projet :

```
mkdir helloworld-ubuntu
cd helloworld-ubuntu
```

Créer le Dockerfile :

Créez un fichier nommé `Dockerfile` dans ce répertoire :

```
touch Dockerfile
```

Nano :

```
# Utiliser une image Ubuntu comme base
FROM ubuntu:22.04

# Mettre à jour les paquets et nettoyer
RUN apt-get update && apt-get install -y \
    ca-certificates \
    && apt-get clean \
    && rm -rf /var/lib/apt/lists/*

# Créer un script avec un shebang
RUN echo '#!/bin/sh' > /usr/local/bin/helloworld && \
    echo 'echo "Hello from Docker Container using Ubuntu!"' >>
/usr/local/bin/helloworld && \
    chmod +x /usr/local/bin/helloworld

# Commande par défaut à exécuter
CMD ["/usr/local/bin/helloworld"]
```

Construire l'image

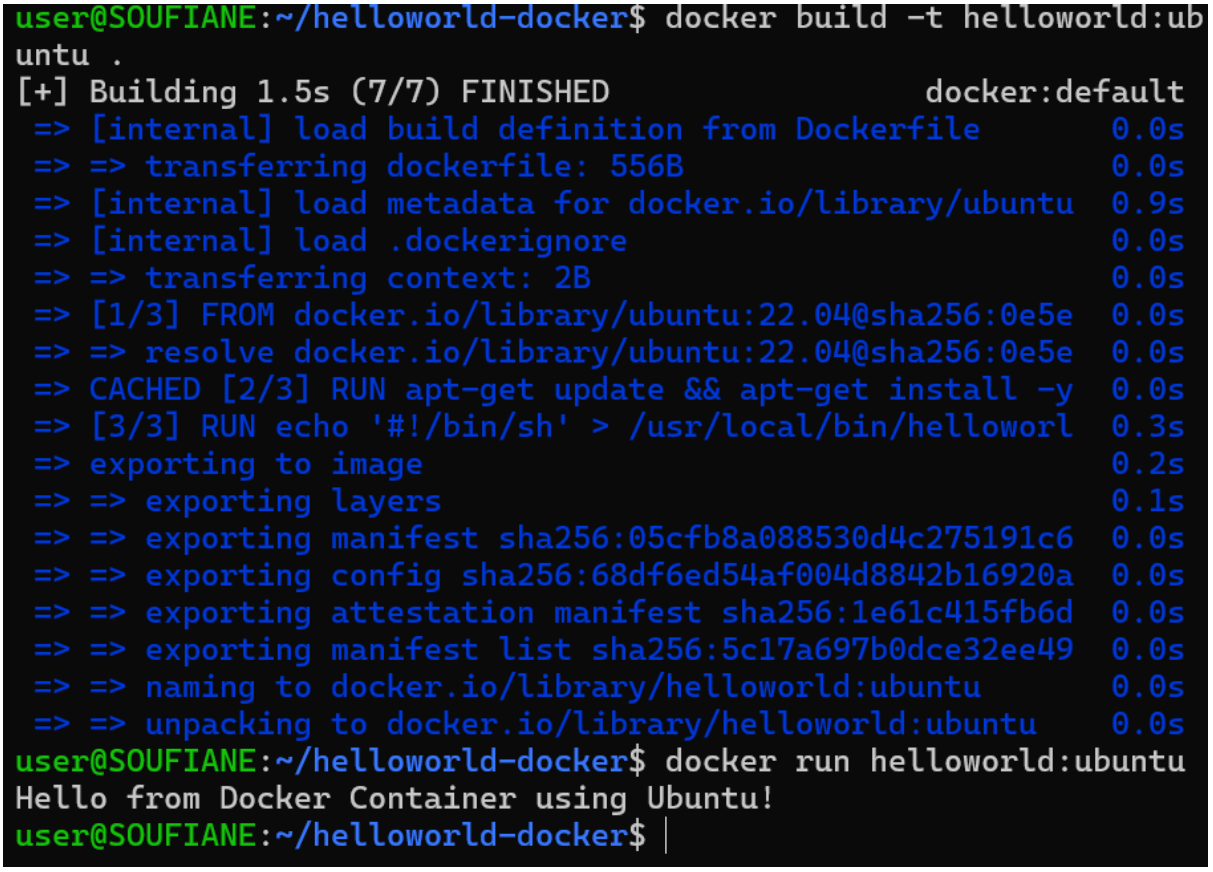
À partir du répertoire contenant le `Dockerfile`, utilisez la commande suivante pour construire l'image Docker :

```
docker build -t helloworld:ubuntu .
```

Exécuter le conteneur

Une fois l'image construite, lancez un conteneur basé sur cette image :

docker run helloworld:ubuntu



JOB 4

```
user@SOUFIANE:~$ cd ssh-docker/
user@SOUFIANE:~/ssh-docker$ ls
Dockerfile  ssh-servertest
user@SOUFIANE:~/ssh-docker$ cd ssh-servertest/
user@SOUFIANE:~/ssh-docker/ssh-servertest$ docker ps
CONTAINER ID   IMAGE           COMMAND               CREATED
       STATUS          PORTS                    NAMES
4c007cd85697   busybox         "sh"                  41 minutes
 ago   Up 41 minutes                            reader
1e5e970d8a30   busybox         "sh"                  41 minutes
 ago   Up 41 minutes                            writer
cd9361def1b0   ssh-servertest   "/usr/sbin/sshd -D"   5 hours ag
o      Up 5 hours      0.0.0.0:3333->2222/tcp   ssh-container_te
st
user@SOUFIANE:~/ssh-docker/ssh-servertest$ ssh root@localhost -p
 3333
root@localhost's password:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 5.15.167.4-microsoft-st
andard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

This system has been minimized by removing packages and content
that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Mon Dec  9 21:12:44 2024 from 172.17.0.1
root@cd9361def1b0:~#
```

Job 5

```
  GNU nano 7.2           /home/user/.bashrc
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

# Docker command aliases
alias dps="docker ps"                          # Liste les cont>
alias dpsa="docker ps -a"                      # Liste tous les>
alias drm="docker rm"                          # Supprime un co>
alias drma="docker rm $(docker ps -aq)"        # Supprime tous >
alias dimg="docker images"                     # Liste toutes l>
alias drmi="docker rmi"                        # Supprime une i>
alias drmia="docker rmi $(docker images -q)"   # Supprime toute>
alias dexec="docker exec -it"                  # Exécute une co>
alias dstart="docker start"                    # Démarre un con>
alias dstop="docker stop"                      # Arrête un cont>
alias dbuild="docker build -t"                 # Construit une >
alias drun="docker run -it"                    # Lance un conte>
alias drund="docker run -d"                    # Lance un conte>
alias dlogs="docker logs"                      # Affiche les lo>
alias dlogsf="docker logs -f"                  # Suivi en temps>
alias dinsp="docker inspect"                   # Affiche les dé>
alias dstopall="docker stop $(docker ps -q)"   # Arrête tous le>
alias dclean="docker system prune -f"          # Nettoie Docker>

^G Help      ^O Write Out^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File^\ Replace   ^U Paste     ^J Justify
```

Job 6

```
user@SOUFIANE:~$ docker volume ls
DRIVER    VOLUME NAME
local     partage
user@SOUFIANE:~$ docker volume inspect partage
[
    {
        "CreatedAt": "2024-12-09T21:19:00Z",
        "Driver": "local",
        "Labels": null,
        "Mountpoint": "/var/lib/docker/volumes/partage/_data",
        "Name": "partage",
        "Options": null,
        "Scope": "local"
    }
]
user@SOUFIANE:~$
```