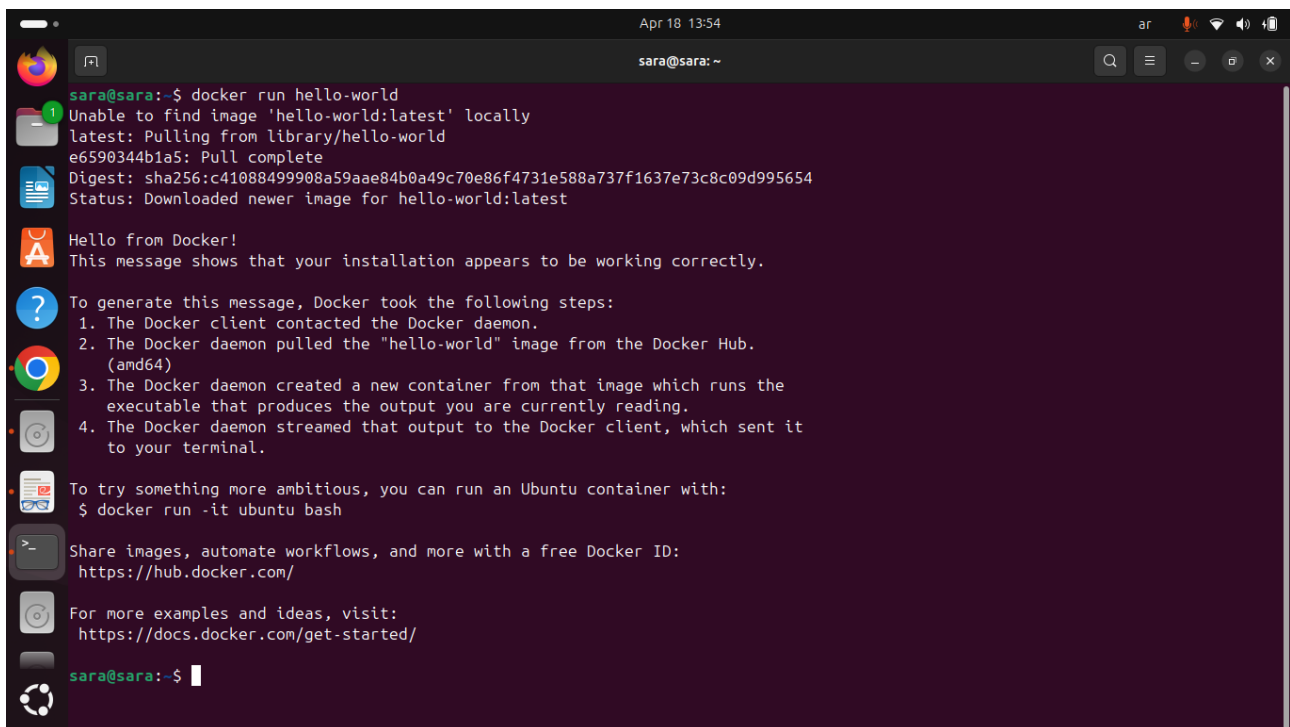


Problem 1:

- Run the container hello-world
- Check the container status
- Start the stopped container
- Remove the container
- Remove the image

1-

A terminal window titled 'sara@sara: ~' with a dark purple background. The terminal shows the command 'docker run hello-world' being executed. The output indicates that the 'hello-world:latest' image was not found locally and was pulled from the Docker Hub. The digest is 'sha256:c41088499908a59aae84b0a49c70e86f4731e588a737f1637e73c8c09d995654'. The status is 'Downloaded newer image for hello-world:latest'. The output then shows a 'Hello from Docker!' message, followed by a list of steps Docker took to generate the message. The steps are: 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. Below the steps, there is a suggestion to try something more ambitious by running an Ubuntu container with the command '\$ docker run -it ubuntu bash'. At the bottom, there are links to 'https://hub.docker.com/' and 'https://docs.docker.com/get-started/'. The terminal prompt 'sara@sara:~\$' is visible at the bottom.

```
sara@sara:~$ docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
e6590344b1a5: Pull complete
Digest: sha256:c41088499908a59aae84b0a49c70e86f4731e588a737f1637e73c8c09d995654
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:
https://hub.docker.com/

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

sara@sara:~$
```

```
Apr 18 13:54
sara@sara: ~
latest: Pulling from library/hello-world
e6590344b1a5: Pull complete
Digest: sha256:c41088499908a59aae84b0a49c70e86f4731e588a737f1637e73c8c09d995654
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:
https://hub.docker.com/

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

sara@sara:~$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
6593c239285d   hello-world    "/hello"                2 minutes ago   Exited (0)    2 minutes ago   dazzling_boyd
046b0b5022e6   nginx         "/docker-entrypoint..." 3 hours ago     Exited (0)    2 hours ago     musing_noether
sara@sara:~$
```

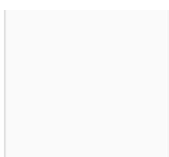
```
Apr 18 13:59
sara@sara: ~
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

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:
https://hub.docker.com/

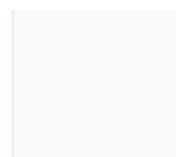
For more examples and ideas, visit:
https://docs.docker.com/get-started/

sara@sara:~$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
6593c239285d   hello-world    "/hello"                2 minutes ago   Exited (0)    2 minutes ago   dazzling_boyd
046b0b5022e6   nginx         "/docker-entrypoint..." 3 hours ago     Exited (0)    2 hours ago     musing_noether
sara@sara:~$ docker start 6593c239285d
6593c239285d
sara@sara:~$ docker rm 6593c239285d
6593c239285d
sara@sara:~$ docker rmi hello-world
Untagged: hello-world:latest
Untagged: hello-world@sha256:c41088499908a59aae84b0a49c70e86f4731e588a737f1637e73c8c09d995654
Deleted: sha256:74cc54e27dc41bb10dc4b2226072d469509f2f22f1a3ce74f4a59661a1d44602
Deleted: sha256:63a41026379f4391a306242eb0b9f26dc3550d863b7fdbb97d899f6eb89efe72
sara@sara:~$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
046b0b5022e6   nginx         "/docker-entrypoint..." 3 hours ago     Exited (0)    2 hours ago     musing_noether
sara@sara:~$
```



Problem 2:

- Run container centos or ubuntu in an interactive mode
- Run the following command in the container "echo docker"
- Open a bash shell in the container and touch a file named hello-docker
- Stop the container and remove it. Write your comment about the file hello-docker
- Remove all stopped containers



```
Apr 18 14:19
root@5b371d37c292: /

sara@sara:~$ docker run -it ubuntu
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
2726e237d1a3: Pull complete
Digest: sha256:1e622c5f073b4f6bfad6632f2616c7f59ef256e96fe78bf6a595d1dc4376ac02
Status: Downloaded newer image for ubuntu:latest
root@5b371d37c292:/#
```

```
Apr 18 14:20
root@5b371d37c292: /

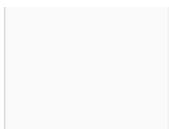
sara@sara:~$ docker run -it ubuntu
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
2726e237d1a3: Pull complete
Digest: sha256:1e622c5f073b4f6bfad6632f2616c7f59ef256e96fe78bf6a595d1dc4376ac02
Status: Downloaded newer image for ubuntu:latest
root@5b371d37c292:/# echo docker
docker
root@5b371d37c292:/# touch hello-docker
root@5b371d37c292:/#
```

```
Apr 18 14:31
sara@sara: ~

sara@sara:~$ docker stop 5b371d37c292
docker rm 5b371d37c292
5b371d37c292
5b371d37c292
sara@sara:~$
```

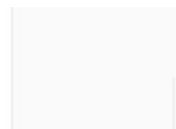
```
Apr 18 14:33
sara@sara: ~

sara@sara:~$ docker stop 5b371d37c292
docker rm 5b371d37c292
5b371d37c292
5b371d37c292
sara@sara:~$ docker container prune -f
Deleted Containers:
046b0b5022e607e48ef51eb4ad787cdbc069fceca03bd47352fbe16416b66032
Total reclaimed space: 1.093kB
sara@sara:~$ docker ps -a
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS        NAMES
sara@sara:~$
```



Problem 3:

- Deploy a MySQL database called app-database. Use the mysql latest image, and use the -e flag to set MYSQL_ROOT_PASSWORD to P4sSw0rd0!. The container should run in the background.



```
sara@sara: ~
sara@sara:~$ docker run -d \
--name app-database \
-e MYSQL_ROOT_PASSWORD=P4sSw0rd0! \
mysql:latest
Unable to find image 'mysql:latest' locally
latest: Pulling from library/mysql
cea172a6e83b: Pull complete
fa811e9a869e: Pull complete
47a2982daa21: Pull complete
634d7076afe3: Pull complete
aa8a3958f09f: Pull complete
84e4e5ea3754: Pull complete
2275c0ff11a0: Pull complete
2792ea2d4e0e: Pull complete
f488b2cd8494: Pull complete
9451290759df: Pull complete
Digest: sha256:7839322bd6c3174a699586c3ea36314c59b61b4ce01b4146951818b94aef5fd7
Status: Downloaded newer image for mysql:latest
527099071c722da47cd5fdf441521406bb16246b2c722ce4689a39193176b131
sara@sara:~$
```

```
sara@sara: ~
634d7076afe3: Pull complete
aa8a3958f09f: Pull complete
84e4e5ea3754: Pull complete
2275c0ff11a0: Pull complete
2792ea2d4e0e: Pull complete
f488b2cd8494: Pull complete
9451290759df: Pull complete
Digest: sha256:7839322bd6c3174a699586c3ea36314c59b61b4ce01b4146951818b94aef5fd7
Status: Downloaded newer image for mysql:latest
527099071c722da47cd5fdf441521406bb16246b2c722ce4689a39193176b131
sara@sara:~$ docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                               NAMES
527099071c72   mysql:latest "docker-entrypoint.s..." 5 minutes ago  Up 4 minutes  3306/tcp, 33060/tcp               app-database
sara@sara:~$ docker exec -it app-database bash
bash-5.1# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 9.3.0 MySQL Community Server - GPL

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

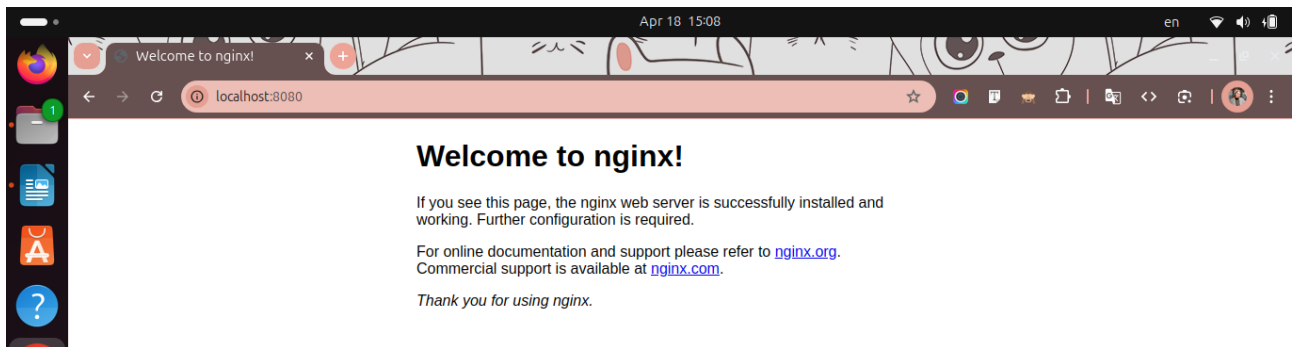
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

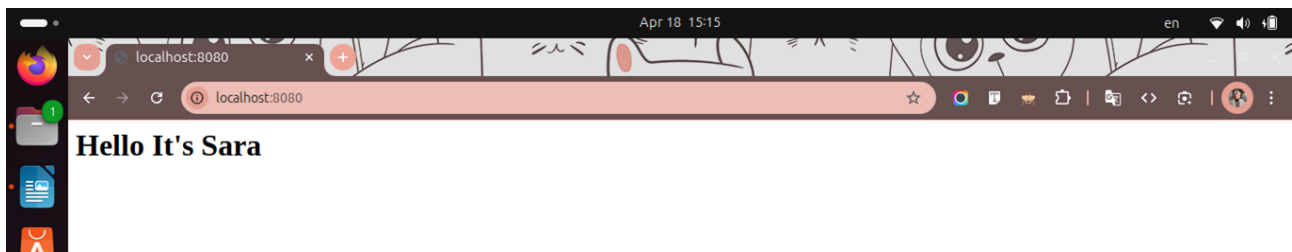
mysql>
```

- Problem 4:
- Run the image Nginx
 - Add html static files to the container and make sure they are accessible
 - Commit the container with image name IMAGE_NAME

```
Apr 18 15:07
sara@sara: ~
sara@sara: ~
sara@sara: ~
sara@sara: ~
sara@sara:~$ docker run -d --name my-nginx -p 8080:80 nginx
2c1e82a2a6445b1b927637490bc82502968cf6a47e881b8c9a9dd710d474fc5d
sara@sara:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS
NAMES
2c1e82a2a644   nginx     "/docker-entrypoint..." 15 seconds ago Up 14 seconds  0.0.0.0:8080->80/tcp, [::]:8080->80/tcp
my-nginx
527099071c72   mysql:latest "docker-entrypoint.s..." 12 minutes ago Up 12 minutes  3306/tcp, 33060/tcp
app-database
sara@sara:~$
```



```
Apr 18 15:24
sara@sara: ~/static
sara@sara: ~/static
sara@sara: ~/static
sara@sara:~/static$ docker run -d --name my-nginx -p 8080:80 nginx
2c1e82a2a6445b1b927637490bc82502968cf6a47e881b8c9a9dd710d474fc5d
sara@sara:~/static$ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS
NAMES
2c1e82a2a644   nginx     "/docker-entrypoint..." 15 seconds ago Up 14 seconds  0.0.0.0:8080->80/tcp, [::]:8080->80/tcp
my-nginx
527099071c72   mysql:latest "docker-entrypoint.s..." 12 minutes ago Up 12 minutes  3306/tcp, 33060/tcp
app-database
sara@sara:~/static$ mkdir static
cd static
echo "<h1>Hello It's Sara!</h1>" > index.html
bash: !: event not found
sara@sara:~/static$ docker cp index.html my-nginx:/usr/share/nginx/html/index.html
lsat /home/sara/static/index.html: no such file or directory
sara@sara:~/static$ echo "<h1>Hello It's Sara!</h1>" > index.html
sara@sara:~/static$ docker cp index.html my-nginx:/usr/share/nginx/html/index.html
Successfully copied 2.05kB to my-nginx:/usr/share/nginx/html/index.html
sara@sara:~/static$ docker commit my-nginx custom-nginx-with-html
sha256:441908383729946d8098f2bf69001f8cf753fef3f51e538d3fc9771f14c8eba4
sara@sara:~/static$ docker images
REPOSITORY          TAG         IMAGE ID      CREATED       SIZE
custom-nginx-with-html latest      441908383729  49 seconds ago 192MB
nginx                latest     4e1b6bae1e48  47 hours ago  192MB
mysql                latest     4b2d796bebc2  3 days ago    859MB
ubuntu              latest     602eb6fb314b  10 days ago   78.1MB
sara@sara:~/static$
```



Problem 5:

- Create a python simple app
- Create a dockerfile to containerize the python app
- Build the image and test it
- (Bonus) create a dockerfile for the same app in smaller size using multi staging
- Push the created image into your docker hub repo

```
sara@sara: ~/my-python-app
sara@sara:~$ mkdir my-python-app
cd my-python-app
sara@sara:~/my-python-app$ echo "print('Hello It's Sara')" > app.py
sara@sara:~/my-python-app$ cat app.py
print('Hello It's Sara')
sara@sara:~/my-python-app$ touch Dockerfile
nano Dockerfile
```

```
GNU nano 7.2 Dockerfile *
FROM python:3.10-slim
WORKDIR /app
COPY app.py .
CMD ["python", "app.py"]
```

```
sara@sara:~/my-python-app
sara@sara:~/my-python-app$ docker build -t sara/python-app .
docker run sara/python-app
[+] Building 6.2s (8/8) FINISHED
=> [internal] load build definition from Dockerfile 0.2s
=> == transferring dockerfile: 115B 0.0s
=> [internal] load metadata for docker.io/library/python:3.10-slim 1.6s
=> [internal] load .dockerignore 0.2s
=> == transferring context: 2B 0.0s
=> [1/3] FROM docker.io/library/python:3.10-slim@sha256:65c843653048a3ba22c8d5083a022f44aef774974f0f7f70cbf8cee4e931ac96 0.0s
=> [internal] load build context 0.2s
=> == transferring context: 58B 0.0s
=> CACHED [2/3] WORKDIR /app 0.0s
=> [3/3] COPY app.py . 1.6s
=> exporting to image 0.8s
=> == exporting layers 0.5s
=> == writing image sha256:4e8b34779d3dd4e4aad8b37b2d3149256bd2cbbae20df2623b8356790e14a99 0.1s
=> == naming to docker.io/sara/python-app 0.1s
Hello It's Sara
sara@sara:~/my-python-app$
```

```
Apr 18 15:58
sara@sara: ~
sara@sara:~/my-python-app$ cd
sara@sara:~$ docker login

USING WEB-BASED LOGIN

Info → To sign in with credentials on the command line, use 'docker login -u <username>'

Your one-time device confirmation code is: JPXK-NPPL
Press ENTER to open your browser or submit your device code here: https://login.docker.com/activate

Waiting for authentication in the browser...

WARNING! Your credentials are stored unencrypted in '/home/sara/.docker/config.json'.
Configure a credential helper to remove this warning. See
https://docs.docker.com/go/credential-store/

Login Succeeded
sara@sara:~$ docker tag sara/python-app saradocker23/python-app

sara@sara:~$ docker login
Authenticating with existing credentials... [Username: saradocker23]

Info → To login with a different account, run 'docker logout' followed by 'docker login'

Login Succeeded
sara@sara:~$
```

```
Apr 18 15:59
sara@sara: ~
sara@sara:~$ docker login
Waiting for authentication in the browser...

WARNING! Your credentials are stored unencrypted in '/home/sara/.docker/config.json'.
Configure a credential helper to remove this warning. See
https://docs.docker.com/go/credential-store/

Login Succeeded
sara@sara:~$ docker tag sara/python-app saradocker23/python-app

sara@sara:~$ docker login
Authenticating with existing credentials... [Username: saradocker23]

Info → To login with a different account, run 'docker logout' followed by 'docker login'

Login Succeeded
sara@sara:~$ docker push saradocker23/python-app
Using default tag: latest
The push refers to repository [docker.io/saradocker23/python-app]
11ee6fad57dc: Pushed
2ae63e9efa2d: Pushed
7a8f0764a02f: Mounted from library/python
ee1f7d46a3d7: Mounted from library/python
5cbc228a1525: Mounted from library/python
ea680fbff095: Mounted from library/python
latest: digest: sha256:6abb3c9378d22472ba6911ac2f0911f35923b73a7ba15b5ea7ba10f193b1f5ce size: 1572
sara@sara:~$
```

Apr 18 16:06en

hub.docker.com/r/saradocker23/python-app

dockerhub


ExploreMy Hub

Search Docker Hub

CtrlK

S

Explore / saradocker23 / python-app



saradocker23/python-app


By [saradocker23](#) · Updated 6 minutes ago

[IMAGE](#)

☆0 ↓0

Manage Repository

OverviewTags



No overview available
This repository doesn't have an overview

Docker Pull Command

```
docker pull saradocker23/python-app
```

Copy

By clicking "Accept All Cookies", you agree to the storing of cookies on your device to enhance site navigation, analyze site usage, and assist in our marketing efforts.

Cookies Settings

Reject All

Accept All Cookies

X