Student: ANGEL EFRAIN ORDONEZ GONZALEZ **Professor: Keerthi Nelaturu**

ID Number: 101483544 **Date: 14/January/2024**

Lab 1

Links

https://hub.docker.com/repositories/angelordonez

https://github.com/angelogzz/BCDV-4032/blob/master/Lab1/Readme.md

Prerequisites for Windows users

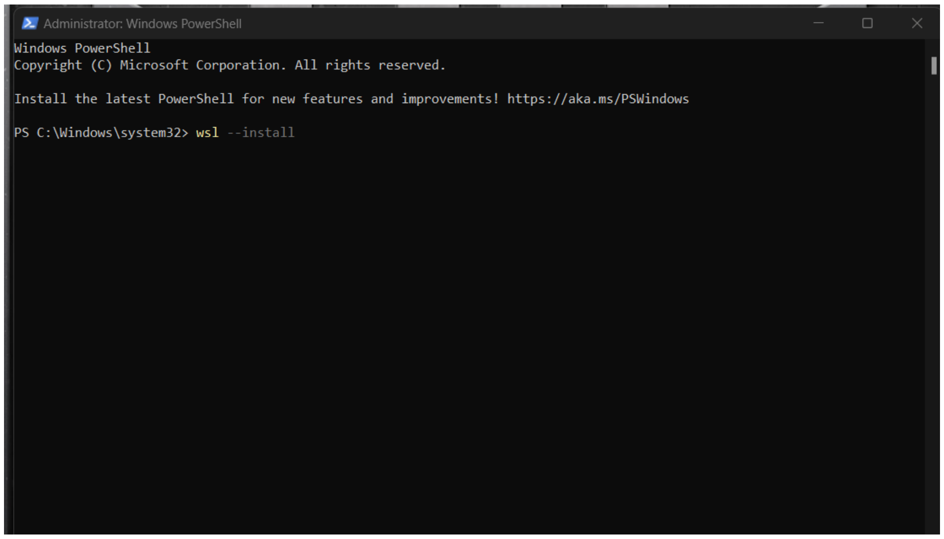
I am working with windows, therefore I must have WSL2, so first of all, I must go to the next pages

https://docs.docker.com/desktop/install/windows-install/

https://learn.microsoft.com/en-us/windows/wsl/install

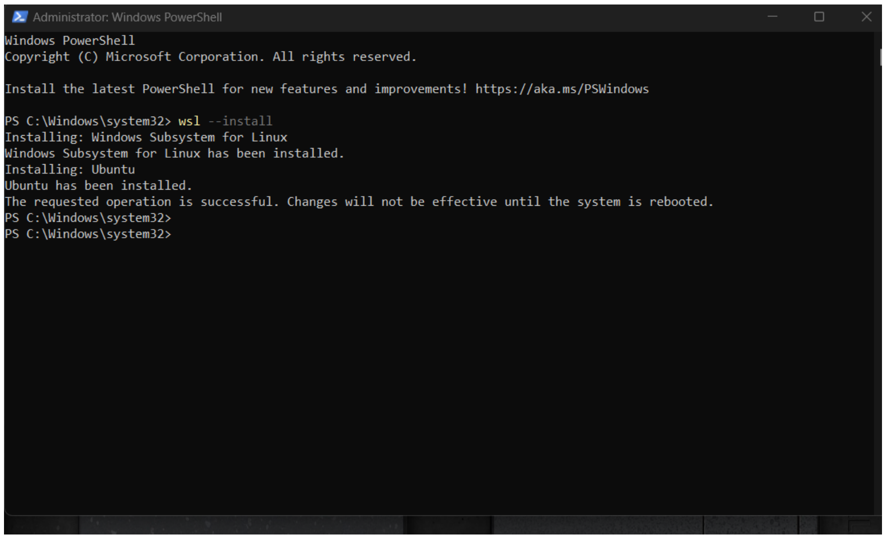
The first command in the Power Shell to install WSL2 is

wsl –install



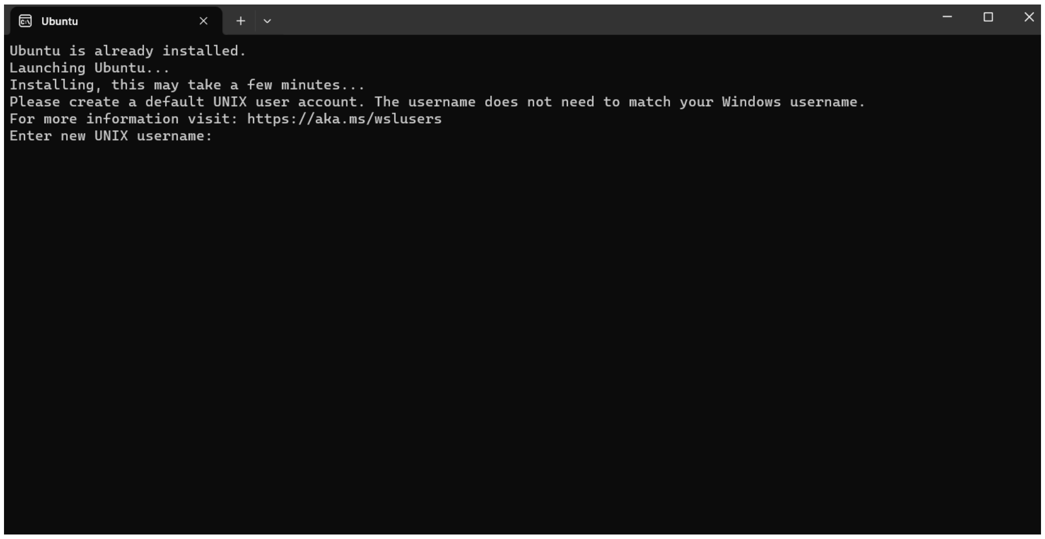
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Windows\system32> wsl --install
```

Here we can see how the installation was successful.



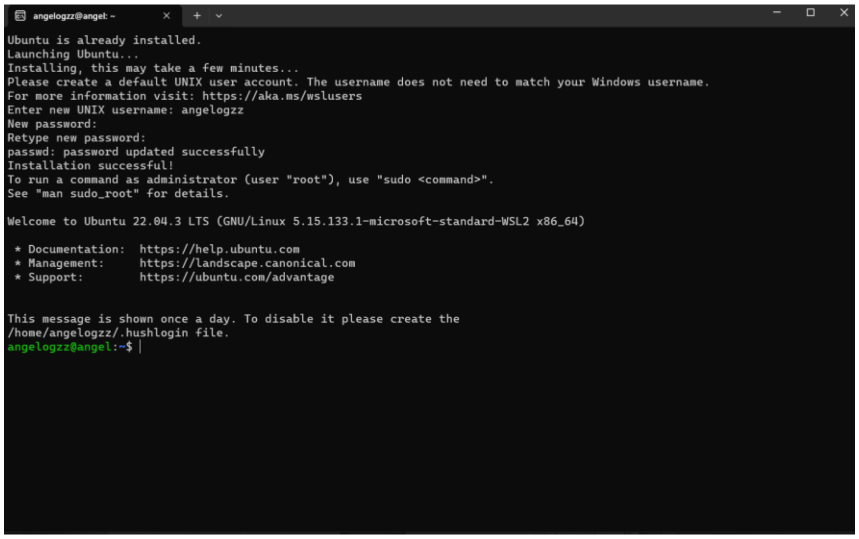
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Windows\system32> wsl --install
Installing: Windows Subsystem for Linux
Windows Subsystem for Linux has been installed.
Installing: Ubuntu
Ubuntu has been installed.
The requested operation is successful. Changes will not be effective until the system is rebooted.
PS C:\Windows\system32>
PS C:\Windows\system32>
```

Now we need to introduce a new user name and password.



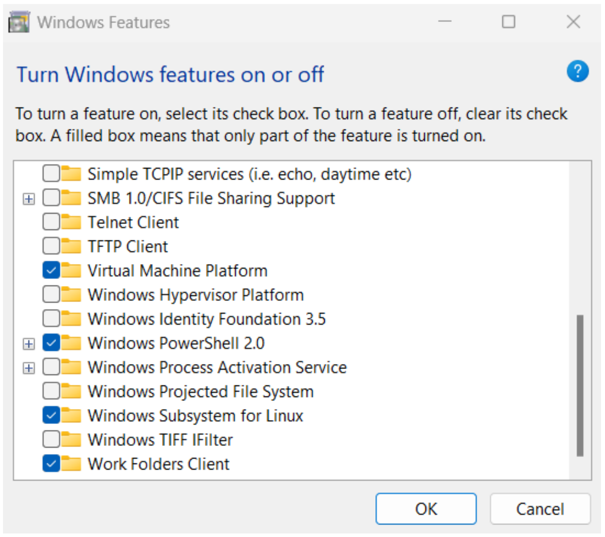
```
Ubuntu is already installed.
Launching Ubuntu...
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username:
```

We can see how Ubuntu 22.04.3 LTS was installed.



```
Ubuntu is already installed.
Launching Ubuntu...
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: angelogzz
New password:
Retype new password:
passwd: password updated successfully
Installation successful!
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.133.1-microsoft-standard-WSL2 x86_64)

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


This message is shown once a day. To disable it please create the
/home/angelogzz/.hushlogin file.
angelogzz@angel:~$ |
```

Also is important check to have on the features “Virtual Machine Platform” and “Windows Subsystem for Linux”.

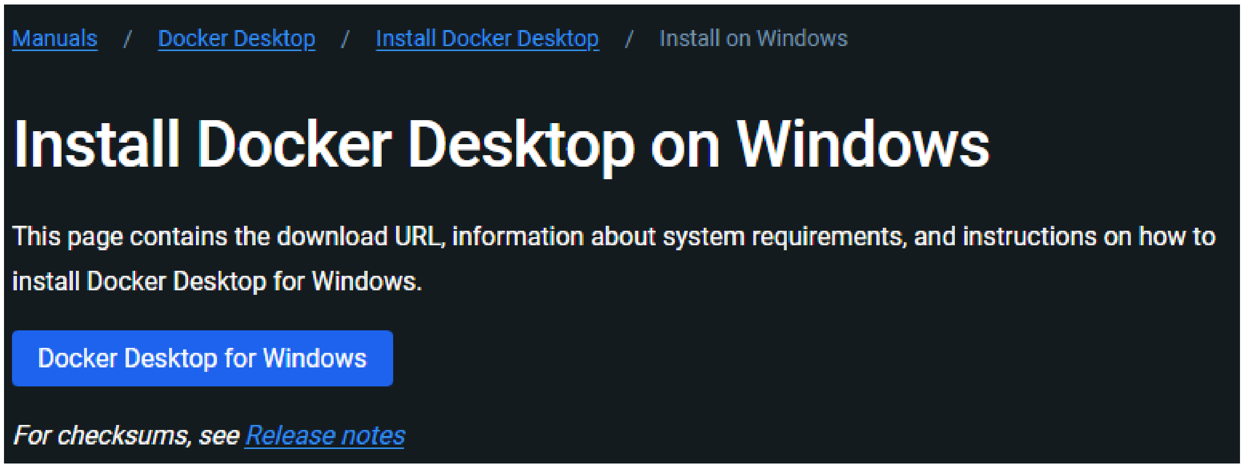


Turn Windows features on or off

To turn a feature on, select its check box. To turn a feature off, clear its check box. A filled box means that only part of the feature is turned on.

- ☐ Simple TCPIP services (i.e. echo, daytime etc)
- ☐ SMB 1.0/CIFS File Sharing Support
- ☐ Telnet Client
- ☐ TFTP Client
- ☑ Virtual Machine Platform
- ☐ Windows Hypervisor Platform
- ☐ Windows Identity Foundation 3.5
- ☑ Windows PowerShell 2.0
- ☐ Windows Process Activation Service
- ☐ Windows Projected File System
- ☑ Windows Subsystem for Linux
- ☐ Windows TIFF IFilter
- ☑ Work Folders Client

Now I should download Docker Desktop for Windows

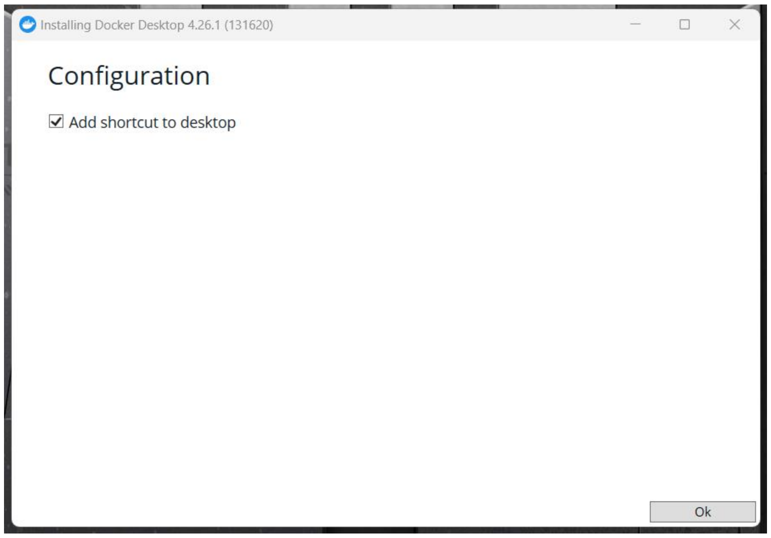

Manuals / Docker Desktop / Install Docker Desktop / Install on Windows

Install Docker Desktop on Windows

This page contains the download URL, information about system requirements, and instructions on how to install Docker Desktop for Windows.

For checksums, see Release notes

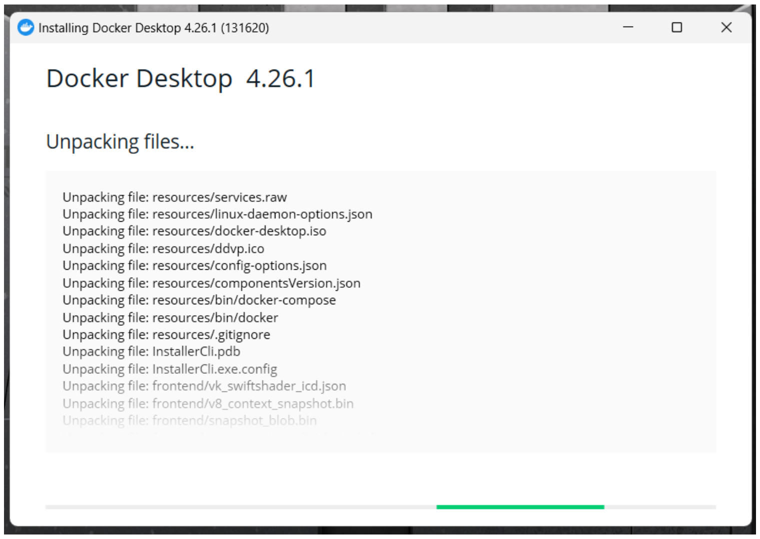
Click on "ok"


Installing Docker Desktop 4.26.1 (131620)

Configuration

☑ Add shortcut to desktop

We can see how the installation will start.

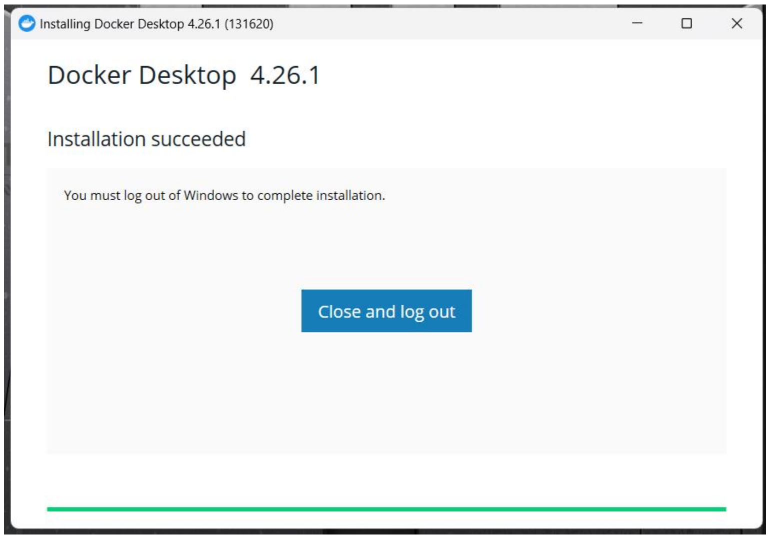

Installing Docker Desktop 4.26.1 (131620)

Docker Desktop 4.26.1

Unpacking files...

```
Unpacking file: resources/services.raw
Unpacking file: resources/linux-daemon-options.json
Unpacking file: resources/docker-desktop.iso
Unpacking file: resources/ddvp.ico
Unpacking file: resources/config-options.json
Unpacking file: resources/componentsVersion.json
Unpacking file: resources/bin/docker-compose
Unpacking file: resources/bin/docker
Unpacking file: resources/.gitignore
Unpacking file: InstallerCli.pdb
Unpacking file: InstallerCli.exe.config
Unpacking file: frontend/vk_swiftshader_icd.json
Unpacking file: frontend/v8_context_snapshot.bin
Unpacking file: frontend/snapshot_blob.bin
```

Here we can see how the installation was successful and we need to click on "Close and log out" and re-entry to own session.

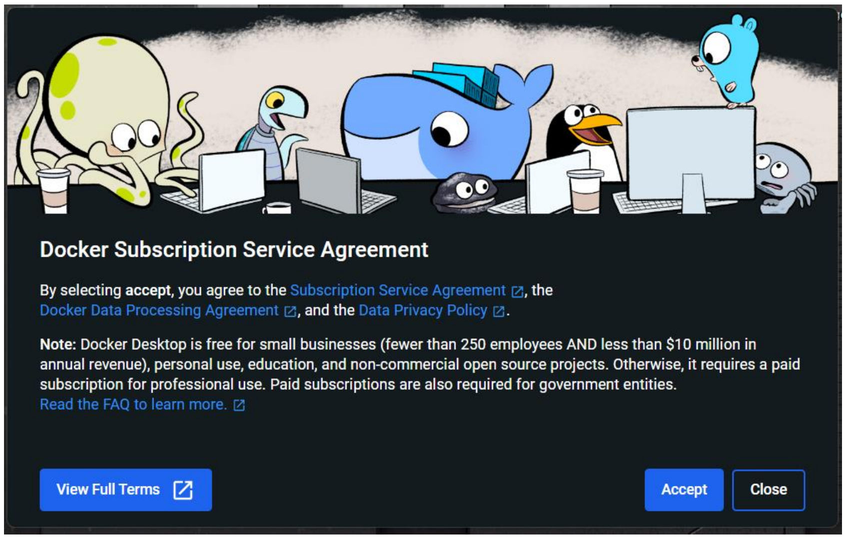

Installing Docker Desktop 4.26.1 (131620)

Docker Desktop 4.26.1

Installation succeeded

You must log out of Windows to complete installation.

Click on "Accept"

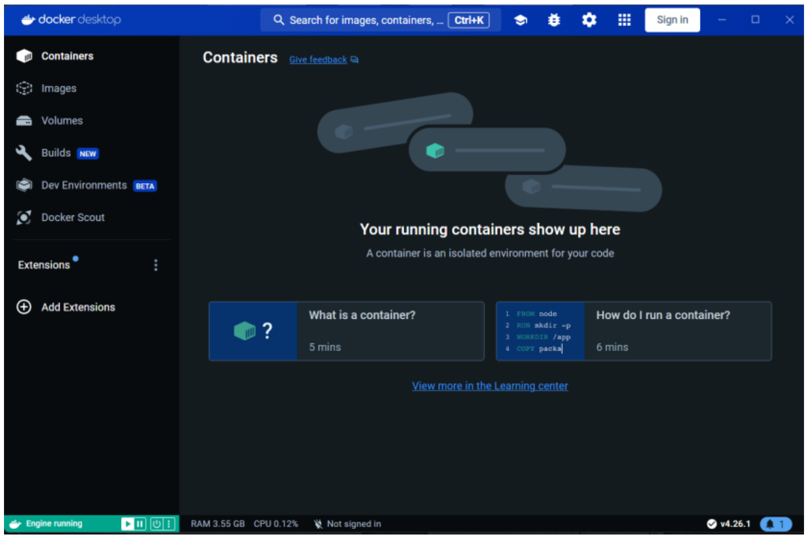


Docker Subscription Service Agreement

By selecting **accept**, you agree to the Subscription Service Agreement, the Docker Data Processing Agreement, and the Data Privacy Policy.

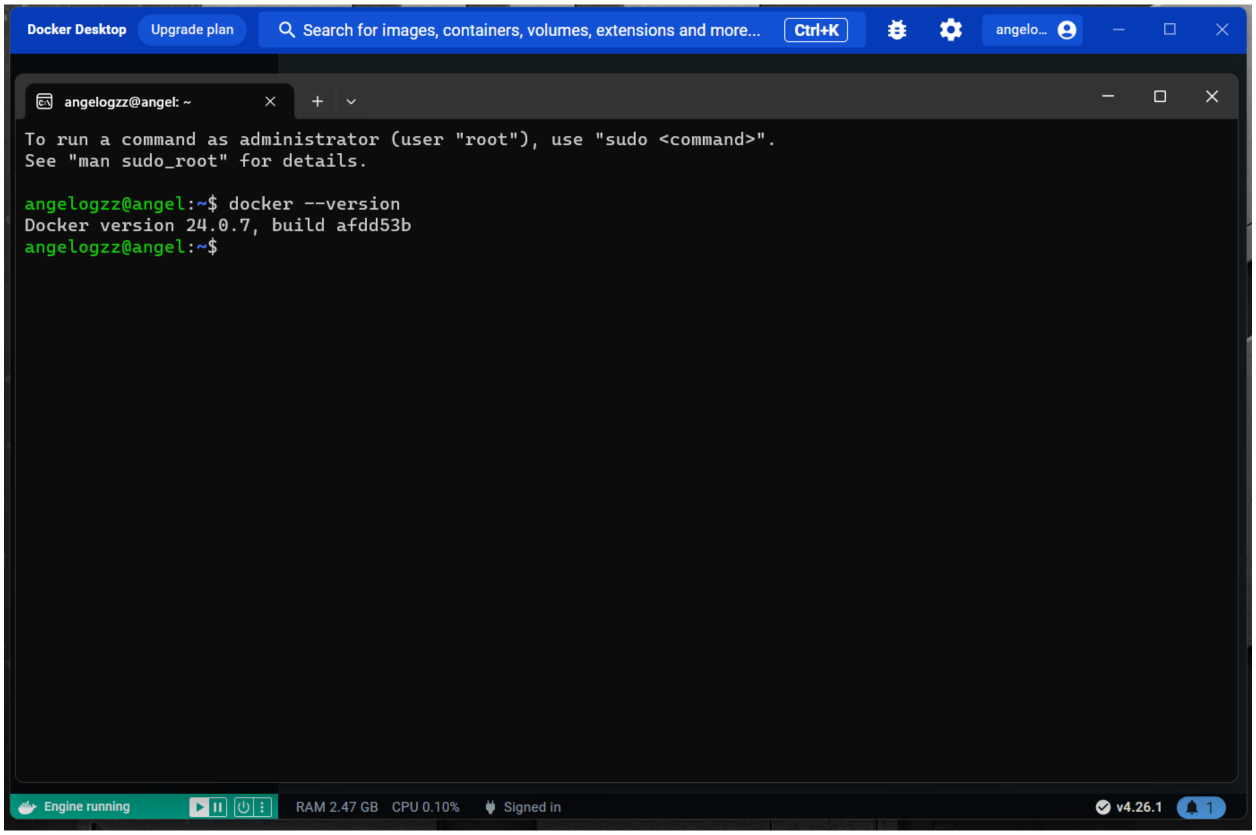
Note: Docker Desktop is free for small businesses (fewer than 250 employees AND less than $10 million in annual revenue), personal use, education, and non-commercial open source projects. Otherwise, it requires a paid subscription for professional use. Paid subscriptions are also required for government entities.
Read the FAQ to learn more.

And finally, Docker is installed.



```
FROM node
RUN mkdir -p
WORKDIR /app
COPY packa
```

We can go to the terminal an check the Docker version

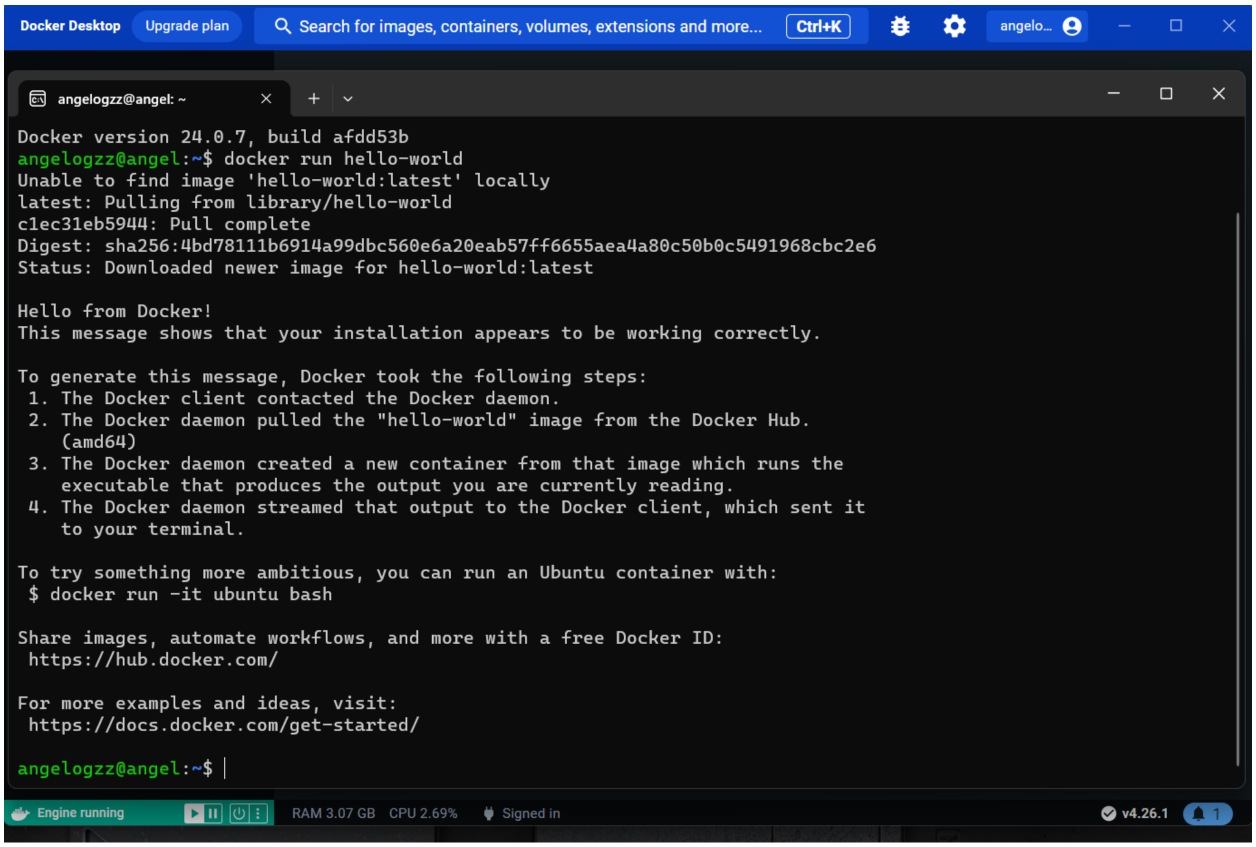


```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

angelogzz@angel:~$ docker --version
Docker version 24.0.7, build afdd53b
angelogzz@angel:~$
```

Setup

Once you are done installing Docker, test your Docker installation by running the following:

```
$ docker run hello-world
```

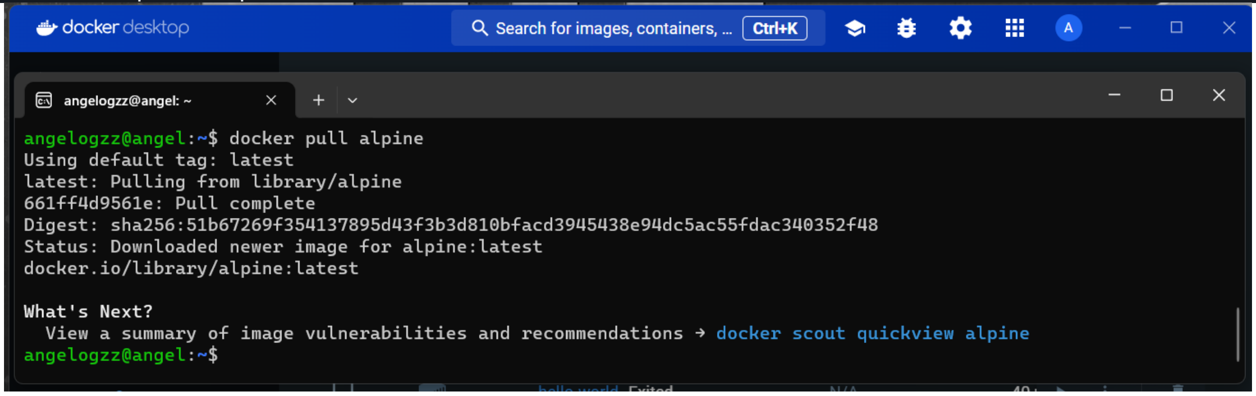


```
Docker version 24.0.7, build afdd53b
angelogzz@angel:~$ docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
c1ec31eb5944: Pull complete
Digest: sha256:4bd78111b6914a99dbc560e6a20eab57ff6655aea4a80c50b0c5491968cbc2e6
Status: Downloaded 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/

angelogzz@angel:~$
```

1.0 Running your first container

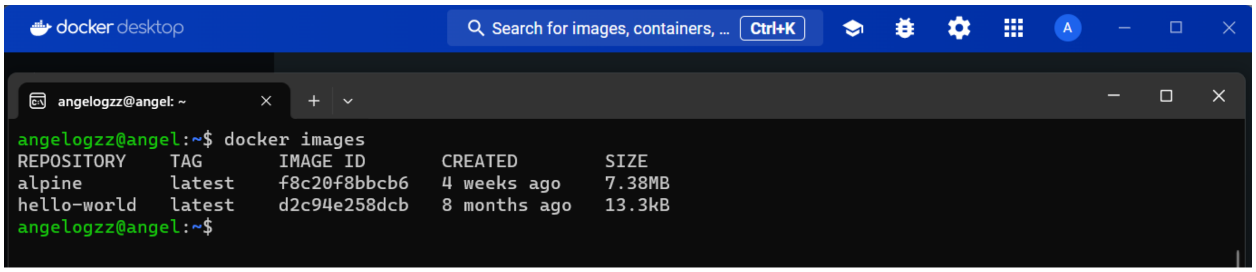
To get started, let's run the following in our terminal:

```
$ docker pull alpine
```



```
angelogzz@angel:~$ docker pull alpine
Using default tag: latest
latest: Pulling from library/alpine
661ff4d9561e: Pull complete
Digest: sha256:51b67269f354137895d43f3b3d810bfacd3945438e94dc5ac55fdac340352f48
Status: Downloaded newer image for alpine:latest
docker.io/library/alpine:latest

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview alpine
angelogzz@angel:~$
```

```
$ docker images
```

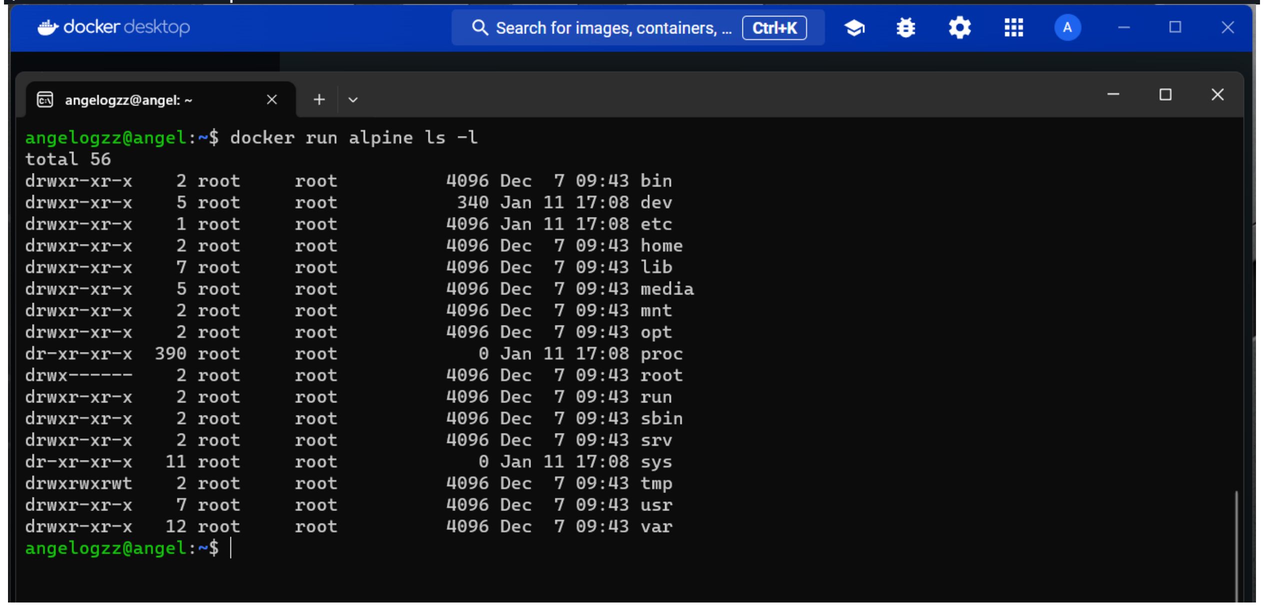


```
angelogzz@angel:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
alpine        latest    f8c20f8bbcb6   4 weeks ago    7.38MB
hello-world   latest    d2c94e258dcb   8 months ago   13.3kB
angelogzz@angel:~$
```

1.1 Docker Run

Let's now run a Docker container based on this image.

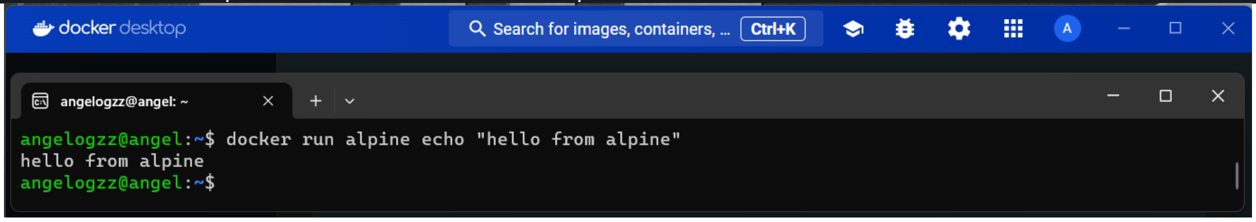
```
$ docker run alpine ls -l
```



```
angelogzz@angel:~$ docker run alpine ls -l
total 56
drwxr-xr-x    2 root     root          4096 Dec  7 09:43 bin
drwxr-xr-x    5 root     root           340 Jan 11 17:08 dev
drwxr-xr-x    1 root     root          4096 Jan 11 17:08 etc
drwxr-xr-x    2 root     root          4096 Dec  7 09:43 home
drwxr-xr-x    7 root     root          4096 Dec  7 09:43 lib
drwxr-xr-x    5 root     root          4096 Dec  7 09:43 media
drwxr-xr-x    2 root     root          4096 Dec  7 09:43 mnt
drwxr-xr-x    2 root     root          4096 Dec  7 09:43 opt
dr-xr-xr-x  390 root     root             0 Jan 11 17:08 proc
drwx------    2 root     root          4096 Dec  7 09:43 root
drwxr-xr-x    2 root     root          4096 Dec  7 09:43 run
drwxr-xr-x    2 root     root          4096 Dec  7 09:43 sbin
drwxr-xr-x    2 root     root          4096 Dec  7 09:43 srv
dr-xr-xr-x   11 root     root             0 Jan 11 17:08 sys
drwxrwxrwt    2 root     root          4096 Dec  7 09:43 tmp
drwxr-xr-x    7 root     root          4096 Dec  7 09:43 usr
drwxr-xr-x   12 root     root          4096 Dec  7 09:43 var
angelogzz@angel:~$
```

Let's try another command.

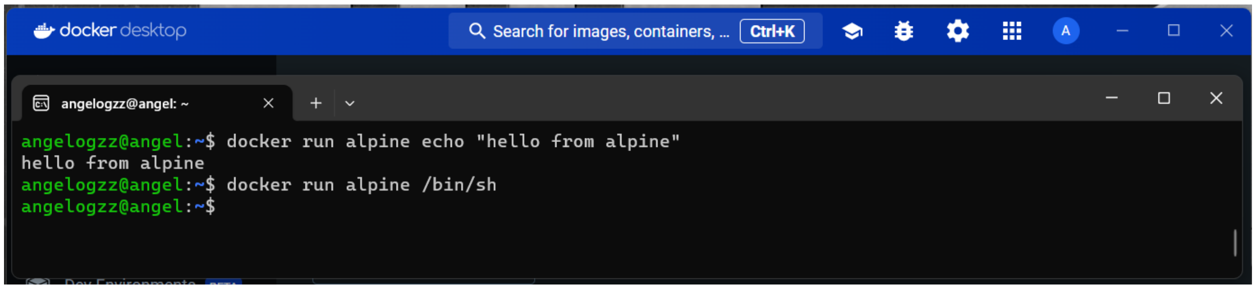
```
$ docker run alpine echo "hello from alpine"
```



```
angelogzz@angel:~$ docker run alpine echo "hello from alpine"
hello from alpine
angelogzz@angel:~$
```

Try another command.

```
$ docker run alpine /bin/sh
```

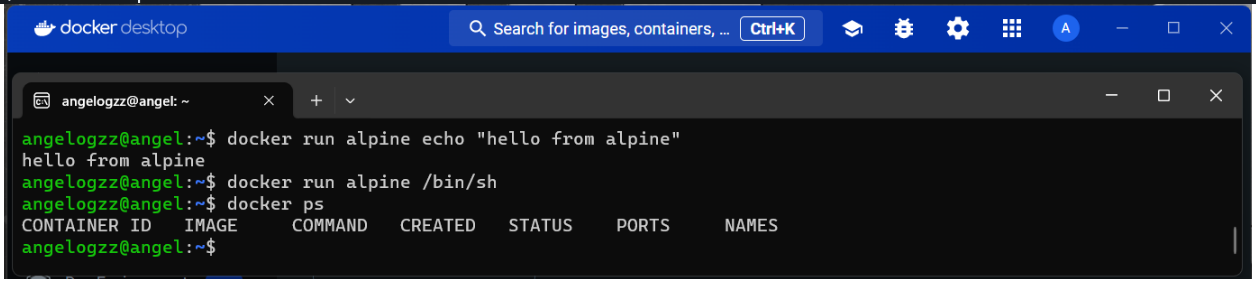


```
angelogzz@angel:~$ docker run alpine echo "hello from alpine"
hello from alpine
angelogzz@angel:~$ docker run alpine /bin/sh
angelogzz@angel:~$
```

Nothing happened. These interactive shells will exit after running any scripted commands, unless they are run in an interactive terminal - so for this example to not exit, you need to docker run -it alpine /bin/sh.

The docker ps command shows you all containers that are currently running.

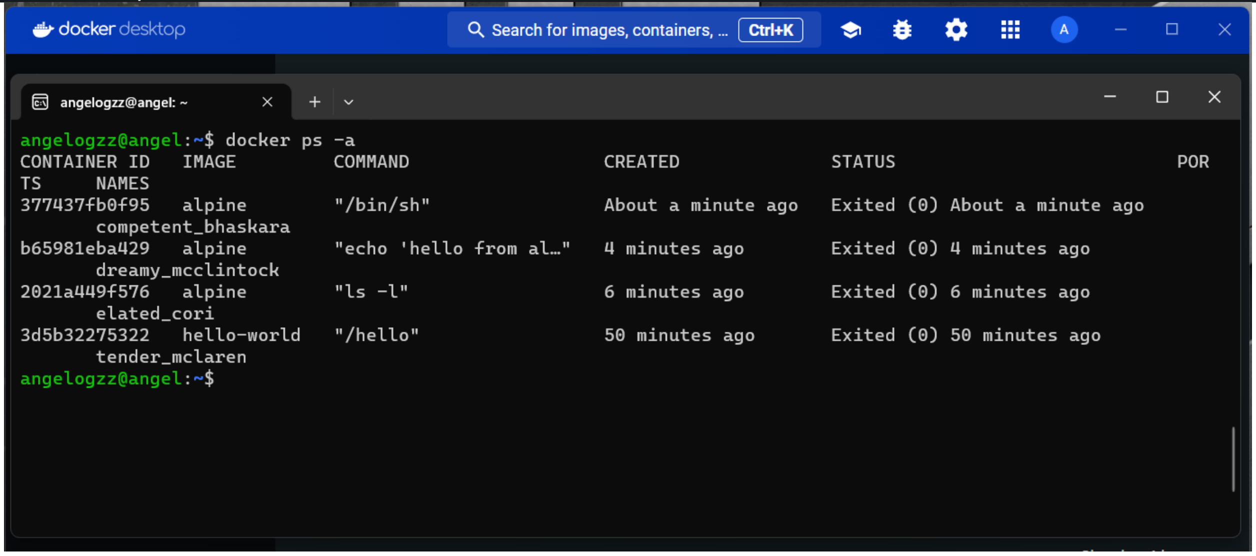
```
$ docker ps
```



```
angelogzz@angel:~$ docker run alpine echo "hello from alpine"
hello from alpine
angelogzz@angel:~$ docker run alpine /bin/sh
angelogzz@angel:~$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
angelogzz@angel:~$
```

Since no containers are running. Let's try a more useful variant: docker ps -a

```
$ docker ps -a
```

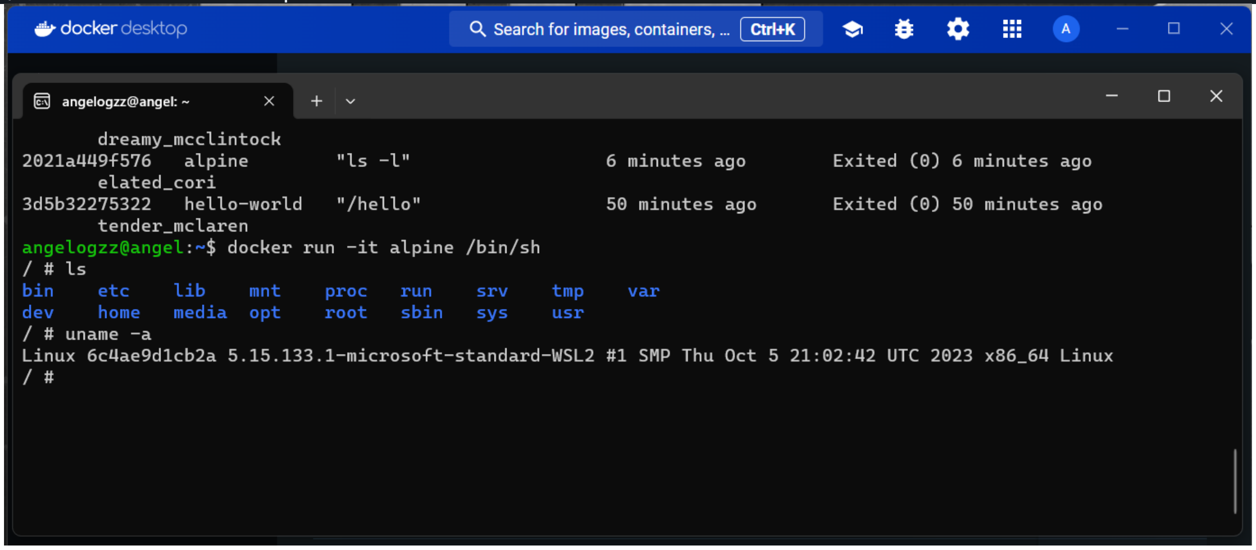


```
angelogzz@angel:~$ docker ps -a
CONTAINER ID   IMAGE         COMMAND                  CREATED              STATUS                          POR
TS     NAMES
377437fb0f95   alpine        "/bin/sh"                About a minute ago   Exited (0) About a minute ago
       competent_bhaskara
b65981eba429   alpine        "echo 'hello from al…"   4 minutes ago        Exited (0) 4 minutes ago
       dreamy_mcclintock
2021a449f576   alpine        "ls -l"                  6 minutes ago        Exited (0) 6 minutes ago
       elated_cori
3d5b32275322   hello-world   "/hello"                 50 minutes ago       Exited (0) 50 minutes ago
       tender_mclaren
angelogzz@angel:~$
```

What you see above is a list of all containers that you ran. Notice that the STATUS column shows that these containers exited a few minutes ago.

You're probably wondering if there is a way to run more than just one command in a container. Let's try that now:

```
$ docker run -it alpine /bin/sh
```



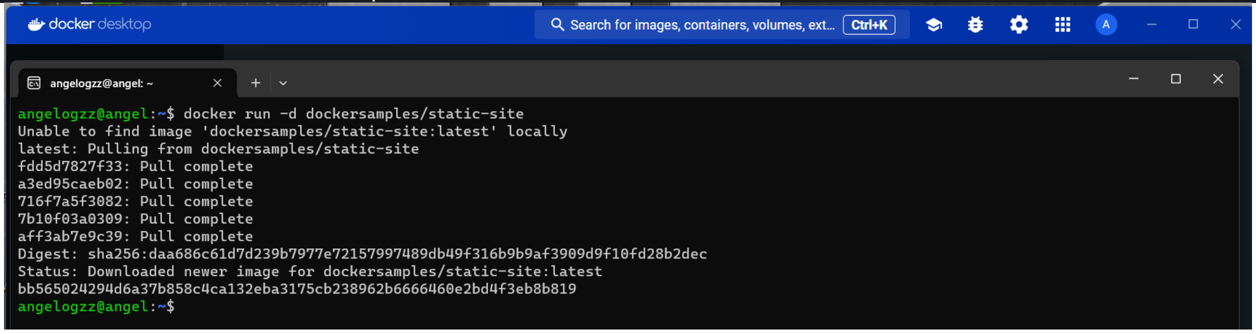
```
        dreamy_mcclintock
2021a449f576   alpine        "ls -l"                  6 minutes ago        Exited (0) 6 minutes ago
        elated_cori
3d5b32275322   hello-world   "/hello"                 50 minutes ago       Exited (0) 50 minutes ago
        tender_mclaren
angelogzz@angel:~$ docker run -it alpine /bin/sh
/ # ls
bin    etc    lib    mnt    proc   run    srv    tmp    var
dev    home   media  opt    root   sbin   sys    usr
/ # uname -a
Linux 6c4ae9d1cb2a 5.15.133.1-microsoft-standard-WSL2 #1 SMP Thu Oct 5 21:02:42 UTC 2023 x86_64 Linux
/ #
```

2.0 Webapps with Docker

Now we are ready to get to the real stuff — deploying web applications with Docker.

2.1 Run a static website in a container

```
$ docker run -d dockersamples/static-site
```

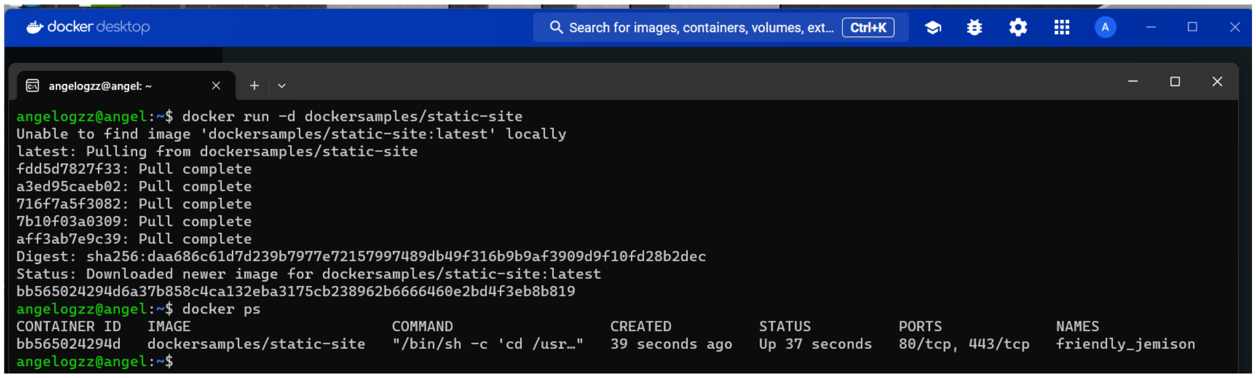


```
angelogzz@angel:~$ docker run -d dockersamples/static-site
Unable to find image 'dockersamples/static-site:latest' locally
latest: Pulling from dockersamples/static-site
fdd5d7827f33: Pull complete
a3ed95caeb02: Pull complete
716f7a5f3082: Pull complete
7b10f03a0309: Pull complete
aff3ab7e9c39: Pull complete
Digest: sha256:daa686c61d7d239b7977e72157997489db49f316b9b9af3909d9f10fd28b2dec
Status: Downloaded newer image for dockersamples/static-site:latest
bb565024294d6a37b858c4ca132eba3175cb238962b6666460e2bd4f3eb8b819
angelogzz@angel:~$
```

Now that the server is running.

Run docker ps to view the running containers.

```
$ docker ps
```

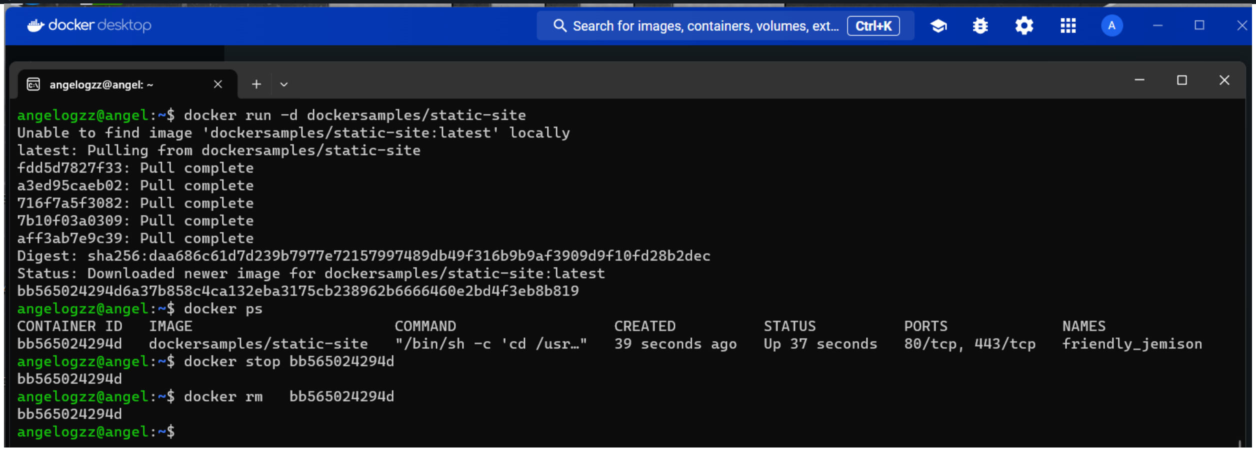


```
angelogzz@angel:~$ docker run -d dockersamples/static-site
Unable to find image 'dockersamples/static-site:latest' locally
latest: Pulling from dockersamples/static-site
fdd5d7827f33: Pull complete
a3ed95caeb02: Pull complete
716f7a5f3082: Pull complete
7b10f03a0309: Pull complete
aff3ab7e9c39: Pull complete
Digest: sha256:daa686c61d7d239b7977e72157997489db49f316b9b9af3909d9f10fd28b2dec
Status: Downloaded newer image for dockersamples/static-site:latest
bb565024294d6a37b858c4ca132eba3175cb238962b6666460e2bd4f3eb8b819
angelogzz@angel:~$ docker ps
CONTAINER ID   IMAGE                       COMMAND                  CREATED          STATUS          PORTS              NAMES
bb565024294d   dockersamples/static-site   "/bin/sh -c 'cd /usr…"   39 seconds ago   Up 37 seconds   80/tcp, 443/tcp    friendly_jemison
angelogzz@angel:~$
```

You will need to use the CONTAINER ID value, a long sequence of characters, to identify the container you want to stop, and then to remove it.

The example below provides the CONTAINER ID on our system; you should use the value that you see in your terminal.

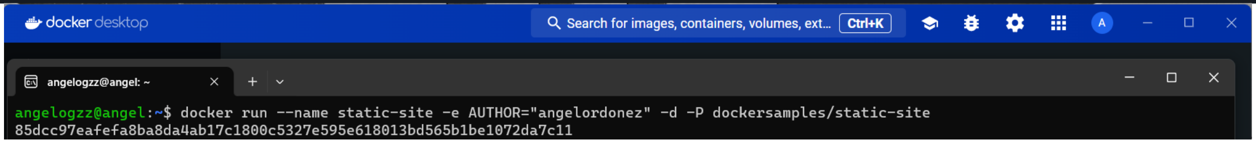
```
$ docker stop bb565024294d
$ docker rm   bb565024294d
```



```
angelogzz@angel:~$ docker run -d dockersamples/static-site
Unable to find image 'dockersamples/static-site:latest' locally
latest: Pulling from dockersamples/static-site
fdd5d7827f33: Pull complete
a3ed95caeb02: Pull complete
716f7a5f3082: Pull complete
7b10f03a0309: Pull complete
aff3ab7e9c39: Pull complete
Digest: sha256:daa686c61d7d239b7977e72157997489db49f316b9b9af3909d9f10fd28b2dec
Status: Downloaded newer image for dockersamples/static-site:latest
bb565024294d6a37b858c4ca132eba3175cb238962b6666460e2bd4f3eb8b819
angelogzz@angel:~$ docker ps
CONTAINER ID   IMAGE                       COMMAND                  CREATED          STATUS          PORTS              NAMES
bb565024294d   dockersamples/static-site   "/bin/sh -c 'cd /usr…"   39 seconds ago   Up 37 seconds   80/tcp, 443/tcp    friendly_jemison
angelogzz@angel:~$ docker stop bb565024294d
bb565024294d
angelogzz@angel:~$ docker rm   bb565024294d
bb565024294d
angelogzz@angel:~$
```

Now, let's launch a container in detached mode as shown below:

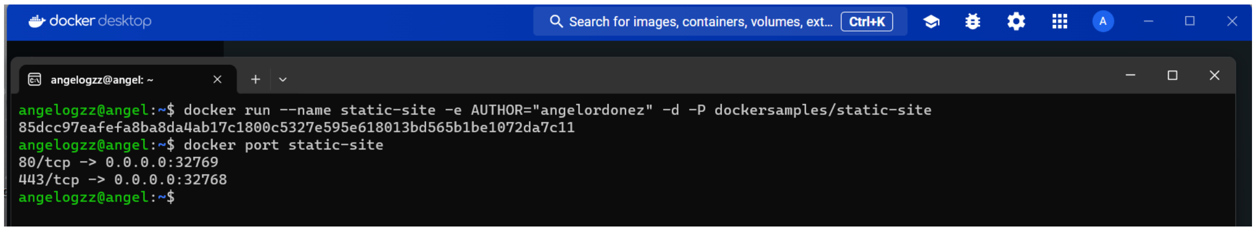
```
$ docker run --name static-site -e AUTHOR="angelordonez" -d -P dockersamples/static-
site
```



```
angelogzz@angel:~$ docker run --name static-site -e AUTHOR="angelordonez" -d -P dockersamples/static-site
85dcc97eafefa8ba8da4ab17c1800c5327e595e618013bd565b1be1072da7c11
```

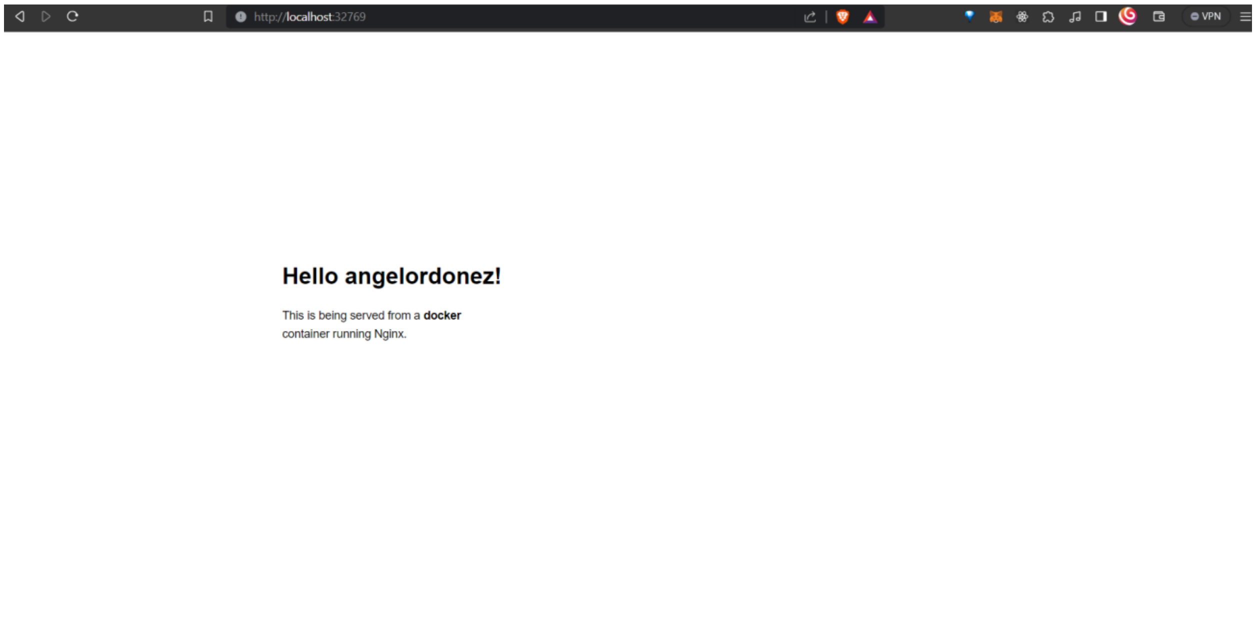
Now you can see the ports by running the docker port command.

```
$ docker port static-site
443/tcp -> 0.0.0.0:32769
80/tcp -> 0.0.0.0:32768
```



```
angelogzz@angel:~$ docker run --name static-site -e AUTHOR="angelordonez" -d -P dockersamples/static-site
85dcc97eafefa8ba8da4ab17c1800c5327e595e618013bd565b1be1072da7c11
angelogzz@angel:~$ docker port static-site
80/tcp -> 0.0.0.0:32769
443/tcp -> 0.0.0.0:32768
angelogzz@angel:~$
```

If you are running Docker for Mac, Docker for Windows, or Docker on Linux, you can open http://localhost:[YOUR_PORT_FOR 80/tcp]. For our example this is http://localhost:32769.



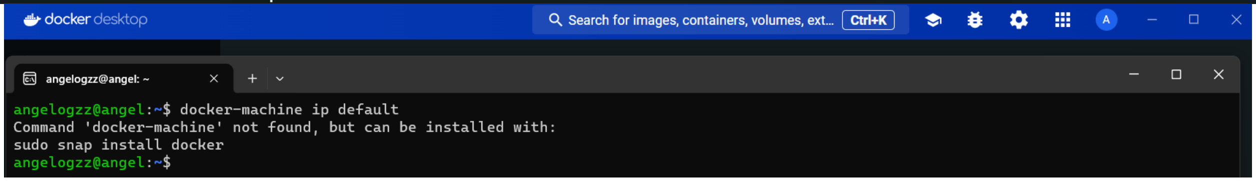
Hello angelordonez!

This is being served from a **docker** container running Nginx.

If you are using Docker Machine on Mac or Windows, you can find the hostname on the command line using docker-machine as follows (assuming you are using the default machine).

Because we are not using Docker Machine it gives us a message where it cannot find the command.

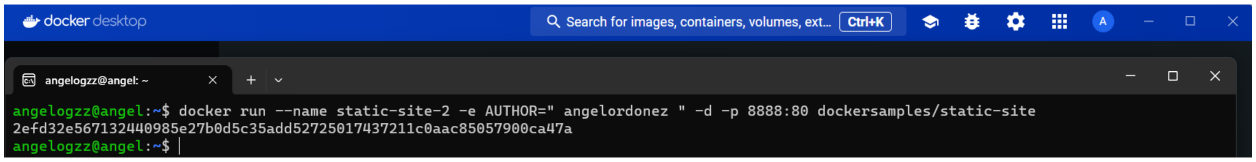
```
$ docker-machine ip default
```



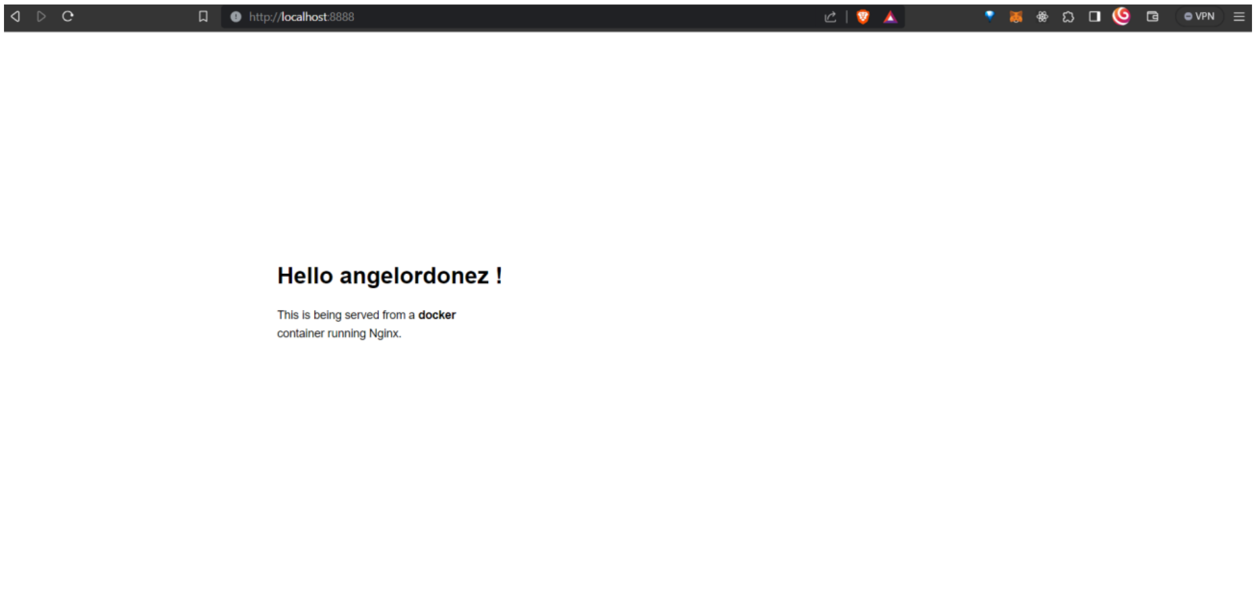
```
angelogzz@angel:~$ docker-machine ip default
Command 'docker-machine' not found, but can be installed with:
sudo snap install docker
angelogzz@angel:~$
```

You can also run a second webserver at the same time, specifying a custom host port mapping to the container's webserver.

```
$ docker run --name static-site-2 -e AUTHOR="  angelordonez " -d -p 8888:80
dockersamples/static-site
```



```
angelogzz@angel:~$ docker run --name static-site-2 -e AUTHOR=" angelordonez " -d -p 8888:80 dockersamples/static-site
2efd32e567132440985e27b0d5c35add52725017437211c0aac85057900ca47a
angelogzz@angel:~$ |
```

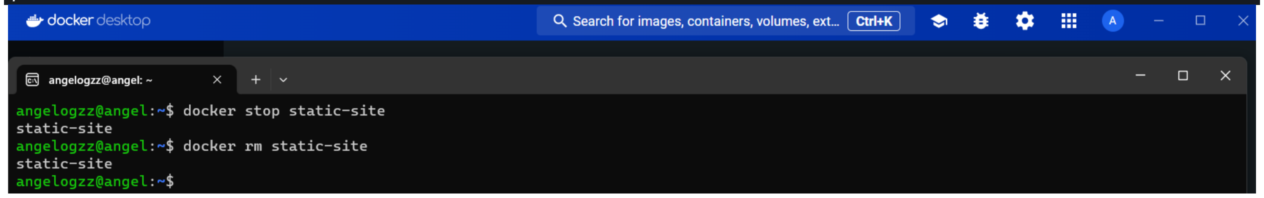


Hello angelordonez !

This is being served from a **docker** container running Nginx.

Let's stop and remove the containers since you won't be using them anymore.

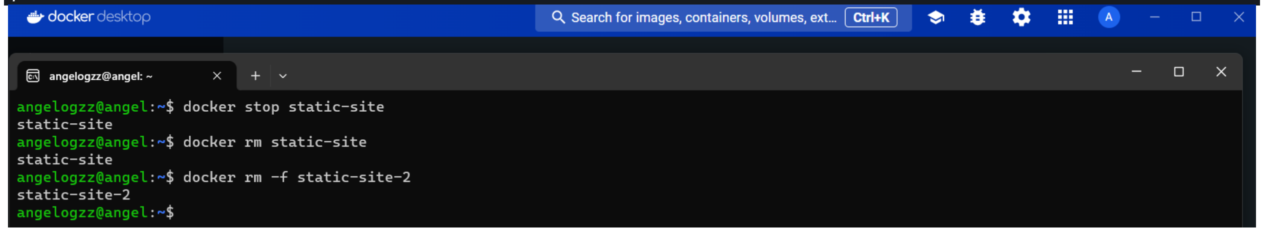
```
$ docker stop static-site
$ docker rm static-site
```



```
angelogzz@angel:~$ docker stop static-site
static-site
angelogzz@angel:~$ docker rm static-site
static-site
angelogzz@angel:~$
```

Let's use a shortcut to remove the second site:

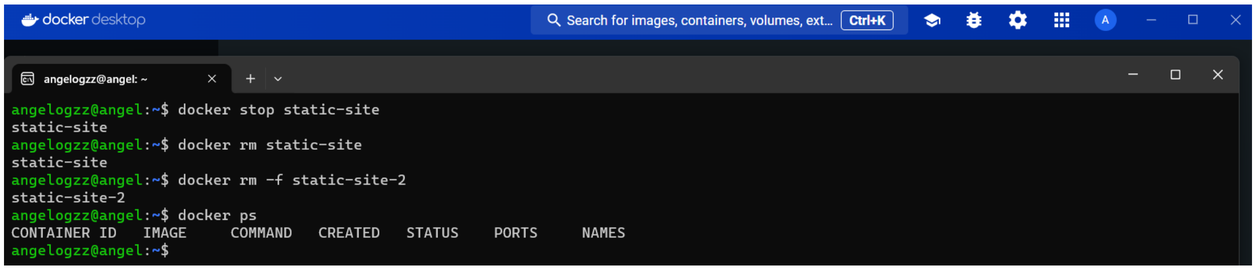
```
$ docker rm -f static-site-2
```



```
angelogzz@angel:~$ docker stop static-site
static-site
angelogzz@angel:~$ docker rm static-site
static-site
angelogzz@angel:~$ docker rm -f static-site-2
static-site-2
angelogzz@angel:~$
```

Run docker ps to make sure the containers are gone.

```
$ docker ps
```

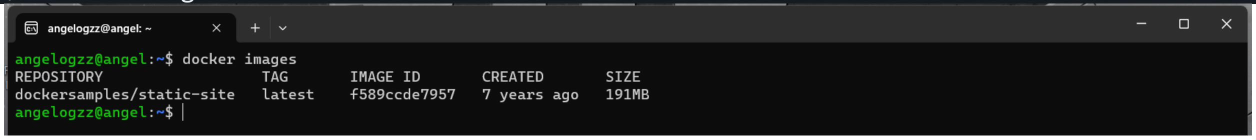


```
angelogzz@angel:~$ docker stop static-site
static-site
angelogzz@angel:~$ docker rm static-site
static-site
angelogzz@angel:~$ docker rm -f static-site-2
static-site-2
angelogzz@angel:~$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
angelogzz@angel:~$
```

2.2 Docker Images

To see the list of images that are available locally on your system, run the docker images command.

```
$ docker images
```



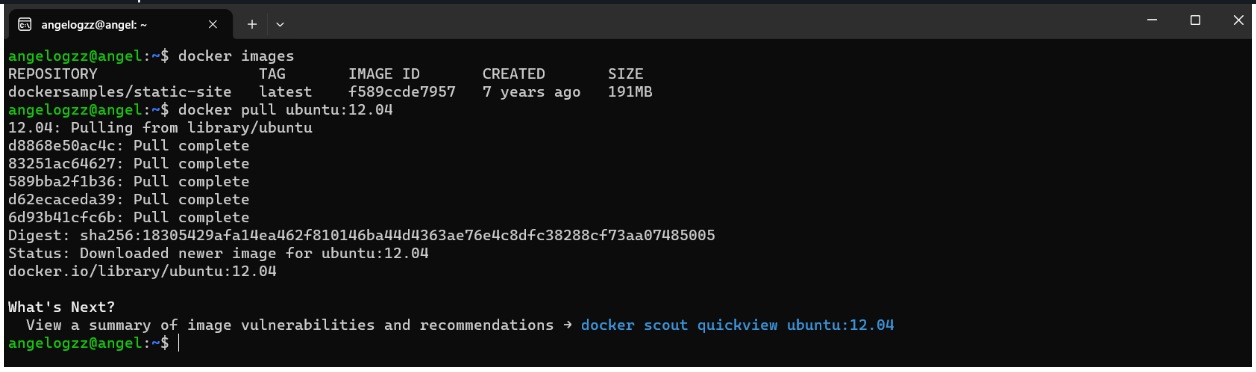
```
angelogzz@angel:~$ docker images
REPOSITORY                   TAG       IMAGE ID       CREATED       SIZE
dockersamples/static-site    latest    f589ccde7957   7 years ago   191MB
angelogzz@angel:~$
```

You will have a different list of images on your machine. The TAG refers to a particular snapshot of the image and the ID is the corresponding unique identifier for that image.

When you do not provide a specific version number, the client defaults to latest.

For example you could pull a specific version of ubuntu image as follows:

```
$ docker pull ubuntu:12.04
```

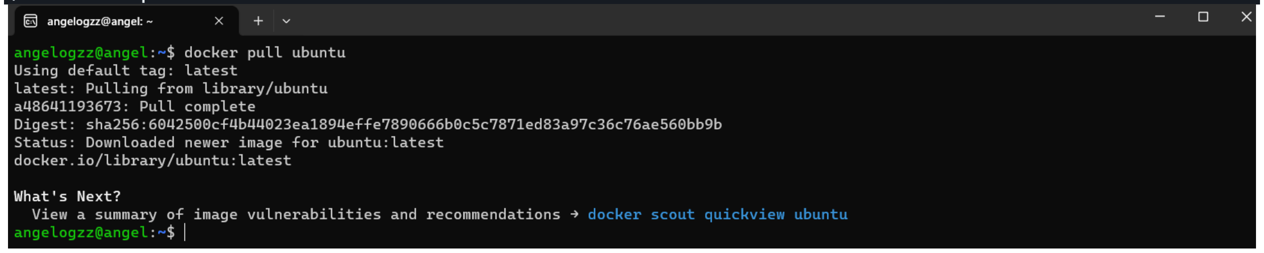


```
angelogzz@angel:~$ docker images
REPOSITORY                   TAG       IMAGE ID       CREATED       SIZE
dockersamples/static-site    latest    f589ccde7957   7 years ago   191MB
angelogzz@angel:~$ docker pull ubuntu:12.04
12.04: Pulling from library/ubuntu
d8868e50ac4c: Pull complete
83251ac64627: Pull complete
589bba2f1b36: Pull complete
d62ecaceda39: Pull complete
6d93b41cfc6b: Pull complete
Digest: sha256:18305429afa14ea462f810146ba44d4363ae76e4c8dfc38288cf73aa07485005
Status: Downloaded newer image for ubuntu:12.04
docker.io/library/ubuntu:12.04

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview ubuntu:12.04
angelogzz@angel:~$
```

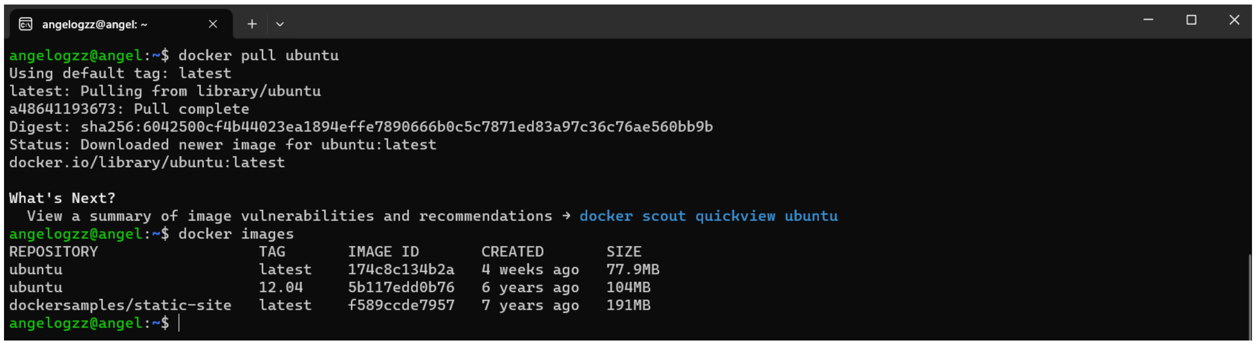
If you do not specify the version number of the image then, as mentioned, the Docker client will default to a version named latest.

For example, the docker pull command given below will pull an image named ubuntu:latest:

```
$ docker pull ubuntu
```



```
angelogzz@angel:~$ docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
a48641193673: Pull complete
Digest: sha256:6042500cf4b44023ea1894effe7890666b0c5c7871ed83a97c36c76ae560bb9b
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview ubuntu
angelogzz@angel:~$
```


angelogzz@angel: ~

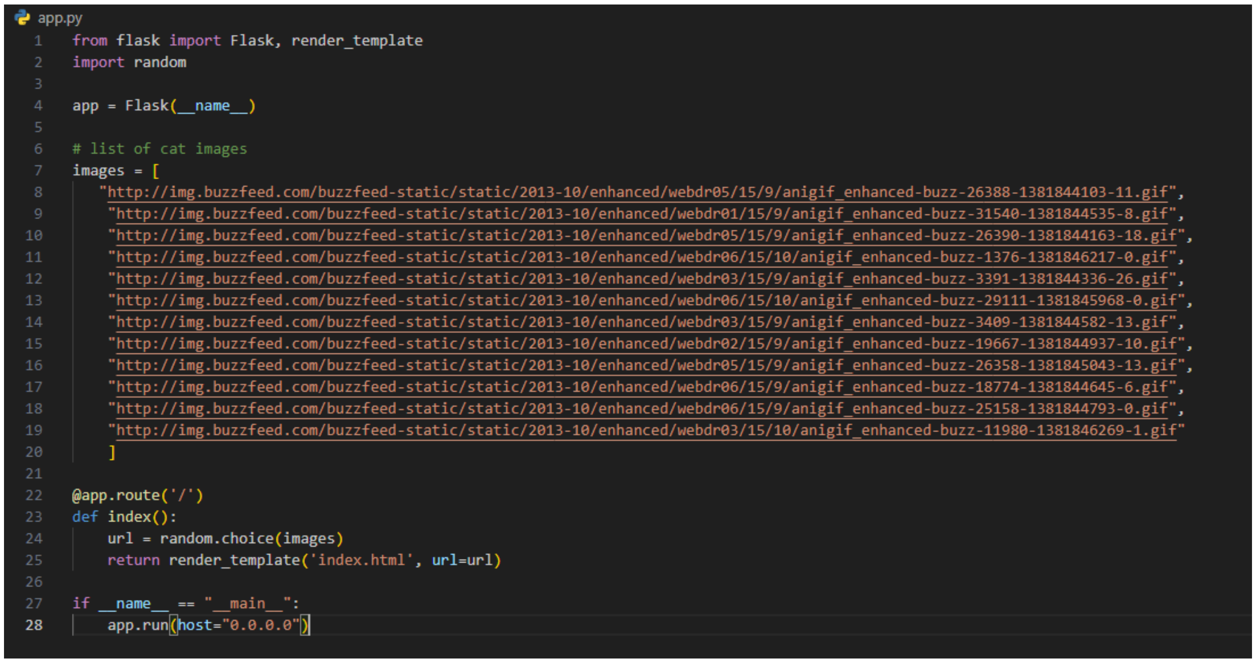
```
angelogzz@angel:~$ docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
a48641193673: Pull complete
Digest: sha256:6042500cf4b44023ea1894effe7890666b0c5c7871ed83a97c36c76ae560bb9b
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview ubuntu
angelogzz@angel:~$ docker images
REPOSITORY                   TAG       IMAGE ID       CREATED        SIZE
ubuntu                       latest    174c8c134b2a   4 weeks ago    77.9MB
ubuntu                       12.04     5b117edd0b76   6 years ago    104MB
dockersamples/static-site    latest    f589ccde7957   7 years ago    191MB
angelogzz@angel:~$ 
```

2.3 Create your first image

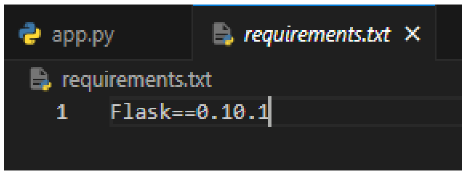
The goal of this exercise is to create a Docker image which will run a Flask app.

You Can see the code here:

2.3.1 Create a Python Flask app that displays random cat pix

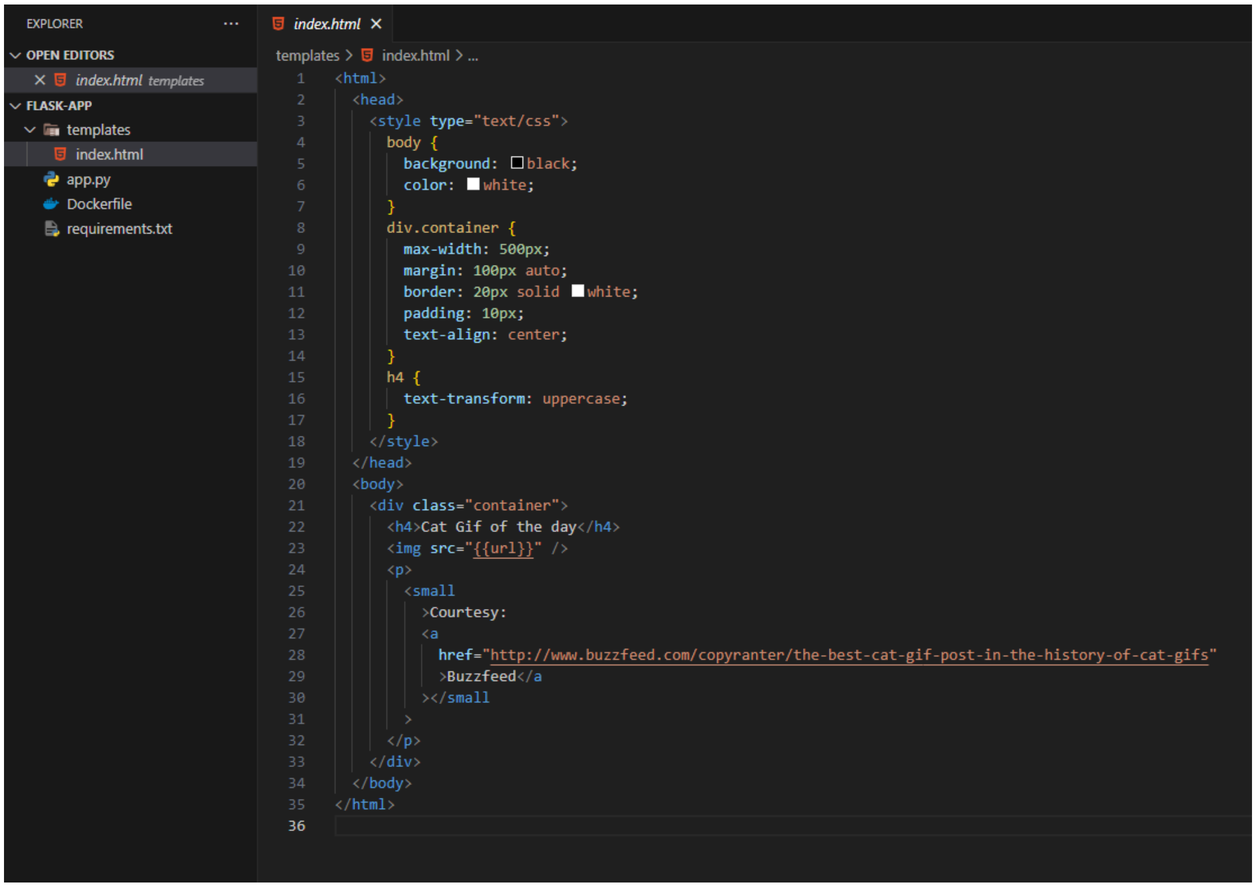

app.py

```python
from flask import Flask, render_template
import random

app = Flask(__name__)

# list of cat images
images = [
    "http://img.buzzfeed.com/buzzfeed-static/static/2013-10/enhanced/webdr05/15/9/anigif_enhanced-buzz-26388-1381844103-11.gif",
     "http://img.buzzfeed.com/buzzfeed-static/static/2013-10/enhanced/webdr01/15/9/anigif_enhanced-buzz-31540-1381844535-8.gif",
     "http://img.buzzfeed.com/buzzfeed-static/static/2013-10/enhanced/webdr05/15/9/anigif_enhanced-buzz-26390-1381844163-18.gif",
     "http://img.buzzfeed.com/buzzfeed-static/static/2013-10/enhanced/webdr06/15/10/anigif_enhanced-buzz-1376-1381846217-0.gif",
     "http://img.buzzfeed.com/buzzfeed-static/static/2013-10/enhanced/webdr03/15/9/anigif_enhanced-buzz-3391-1381844336-26.gif",
     "http://img.buzzfeed.com/buzzfeed-static/static/2013-10/enhanced/webdr06/15/10/anigif_enhanced-buzz-29111-1381845968-0.gif",
     "http://img.buzzfeed.com/buzzfeed-static/static/2013-10/enhanced/webdr03/15/9/anigif_enhanced-buzz-3409-1381844582-13.gif",
     "http://img.buzzfeed.com/buzzfeed-static/static/2013-10/enhanced/webdr02/15/9/anigif_enhanced-buzz-19667-1381844937-10.gif",
     "http://img.buzzfeed.com/buzzfeed-static/static/2013-10/enhanced/webdr05/15/9/anigif_enhanced-buzz-26358-1381845043-13.gif",
     "http://img.buzzfeed.com/buzzfeed-static/static/2013-10/enhanced/webdr06/15/9/anigif_enhanced-buzz-18774-1381844645-6.gif",
     "http://img.buzzfeed.com/buzzfeed-static/static/2013-10/enhanced/webdr06/15/9/anigif_enhanced-buzz-25158-1381844793-0.gif",
     "http://img.buzzfeed.com/buzzfeed-static/static/2013-10/enhanced/webdr03/15/10/anigif_enhanced-buzz-11980-1381846269-1.gif"
     ]

@app.route('/')
def index():
    url = random.choice(images)
    return render_template('index.html', url=url)

if __name__ == "__main__":
    app.run(host="0.0.0.0")
```



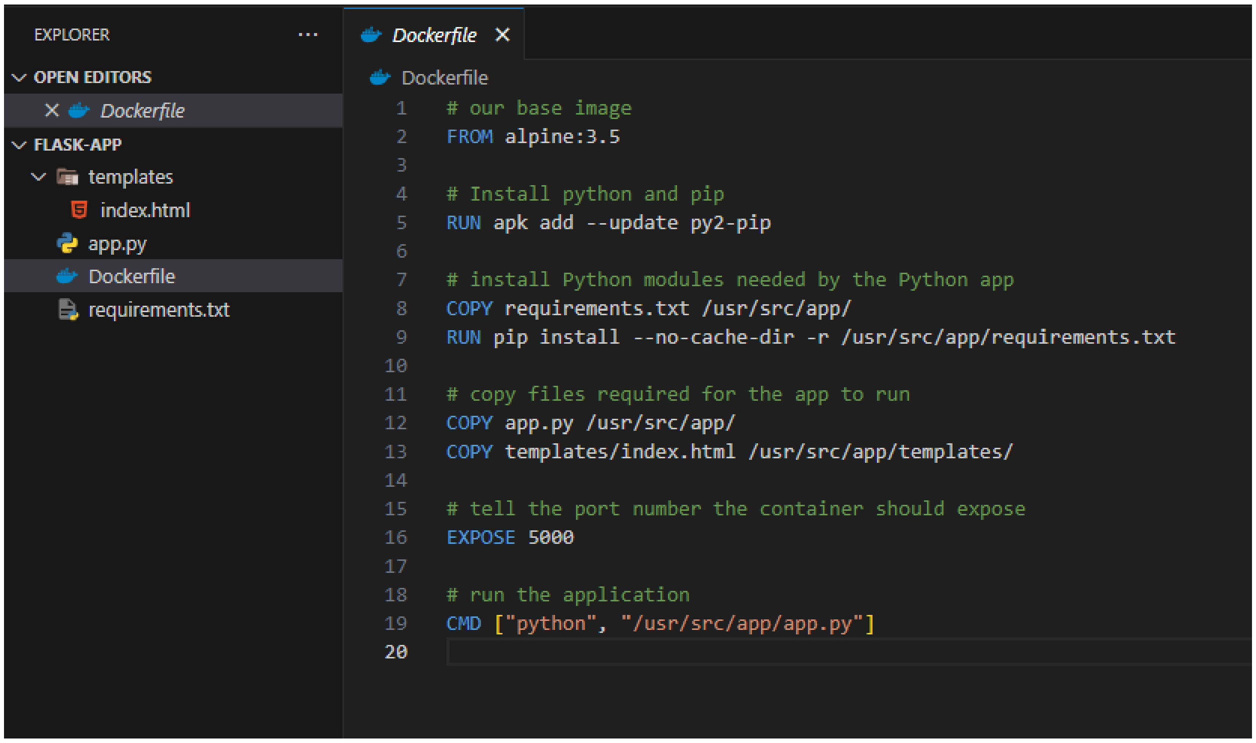
requirements.txt

```
Flask==0.10.1
```



```html
<html>
  <head>
    <style type="text/css">
      body {
        background: black;
        color: white;
      }
      div.container {
        max-width: 500px;
        margin: 100px auto;
        border: 20px solid white;
        padding: 10px;
        text-align: center;
      }
      h4 {
        text-transform: uppercase;
      }
    </style>
  </head>
  <body>
    <div class="container">
      <h4>Cat Gif of the day</h4>
      <img src="{{url}}" />
      <p>
        <small
          >Courtesy:
          <a
            href="http://www.buzzfeed.com/copyranter/the-best-cat-gif-post-in-the-history-of-cat-gifs"
            >Buzzfeed</a
          ></small
        >
      </p>
    </div>
  </body>
</html>
```

2.3.2 Write a Dockerfile

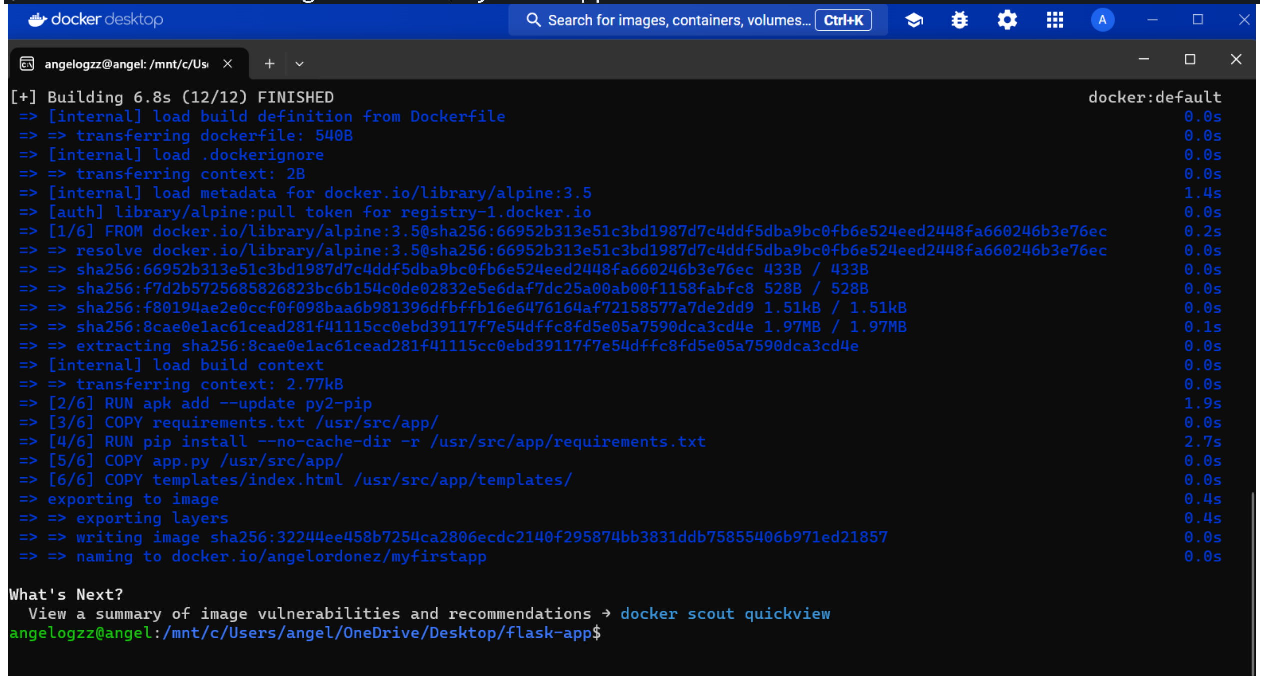


```dockerfile
# our base image
FROM alpine:3.5

# Install python and pip
RUN apk add --update py2-pip

# install Python modules needed by the Python app
COPY requirements.txt /usr/src/app/
RUN pip install --no-cache-dir -r /usr/src/app/requirements.txt

# copy files required for the app to run
COPY app.py /usr/src/app/
COPY templates/index.html /usr/src/app/templates/

# tell the port number the container should expose
EXPOSE 5000

# run the application
CMD ["python", "/usr/src/app/app.py"]
```

2.3.3 Build the image

Build the image with this command:

```
$ docker build -t angelordonez/myfirstapp .
```

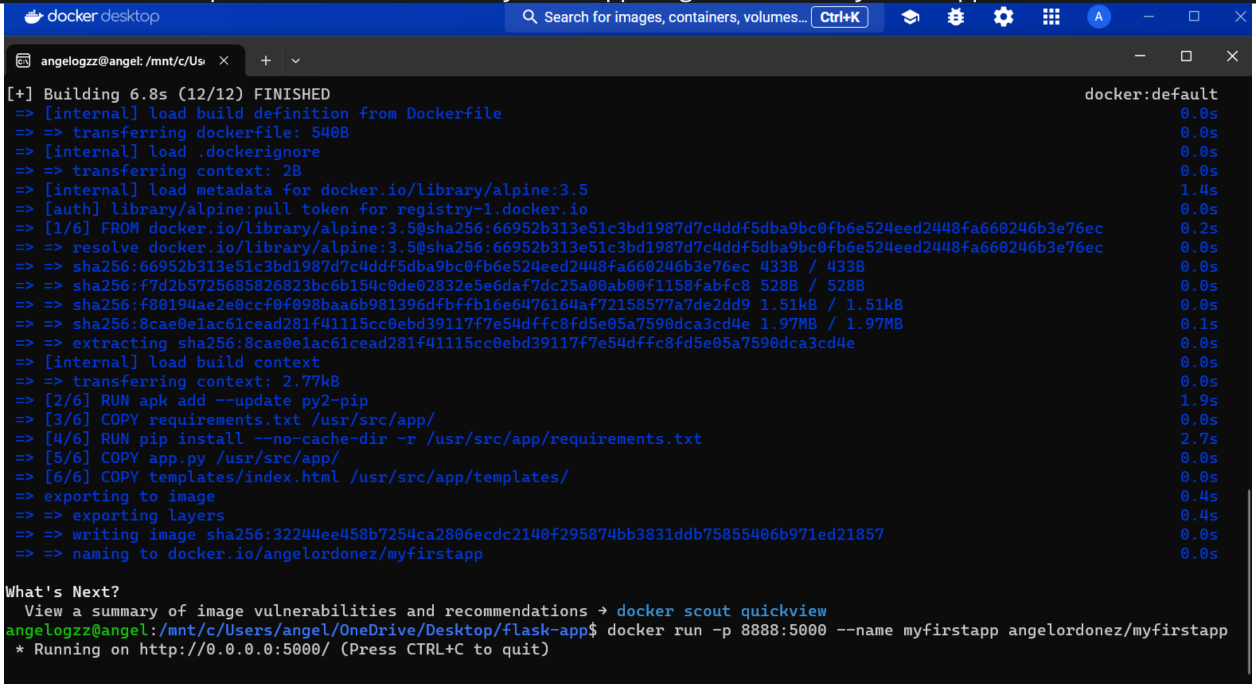


```
[+] Building 6.8s (12/12) FINISHED                                                                   docker:default
 => [internal] load build definition from Dockerfile                                                           0.0s
 => => transferring dockerfile: 540B                                                                           0.0s
 => [internal] load .dockerignore                                                                              0.0s
 => => transferring context: 2B                                                                                0.0s
 => [internal] load metadata for docker.io/library/alpine:3.5                                                  1.4s
 => [auth] library/alpine:pull token for registry-1.docker.io                                                  0.0s
 => [1/6] FROM docker.io/library/alpine:3.5@sha256:66952b313e51c3bd1987d7c4ddf5dba9bc0fb6e524eed2448fa660246b3e76ec  0.2s
 => => resolve docker.io/library/alpine:3.5@sha256:66952b313e51c3bd1987d7c4ddf5dba9bc0fb6e524eed2448fa660246b3e76ec  0.0s
 => => sha256:66952b313e51c3bd1987d7c4ddf5dba9bc0fb6e524eed2448fa660246b3e76ec 433B / 433B                     0.0s
 => => sha256:f7d2b5725685826823bc6b154c0de02832e5e6daf7dc25a00ab00f1158fabfc8 528B / 528B                     0.0s
 => => sha256:f80194ae2e0ccf0f098baa6b981396dfbffb16e6476164af72158577a7de2dd9 1.51kB / 1.51kB                 0.0s
 => => sha256:8cae0e1ac61cead281f41115cc0ebd39117f7e54dffc8fd5e05a7590dca3cd4e 1.97MB / 1.97MB                 0.1s
 => => extracting sha256:8cae0e1ac61cead281f41115cc0ebd39117f7e54dffc8fd5e05a7590dca3cd4e                      0.0s
 => [internal] load build context                                                                              0.0s
 => => transferring context: 2.77kB                                                                            0.0s
 => [2/6] RUN apk add --update py2-pip                                                                         1.9s
 => [3/6] COPY requirements.txt /usr/src/app/                                                                  0.0s
 => [4/6] RUN pip install --no-cache-dir -r /usr/src/app/requirements.txt                                      2.7s
 => [5/6] COPY app.py /usr/src/app/                                                                            0.0s
 => [6/6] COPY templates/index.html /usr/src/app/templates/                                                    0.0s
 => exporting to image                                                                                         0.4s
 => => exporting layers                                                                                        0.4s
 => => writing image sha256:32244ee458b7254ca2806ecdc2140f295874bb3831ddb75855406b971ed21857                   0.0s
 => => naming to docker.io/angelordonez/myfirstapp                                                             0.0s

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$
```

2.3.4 Run your image

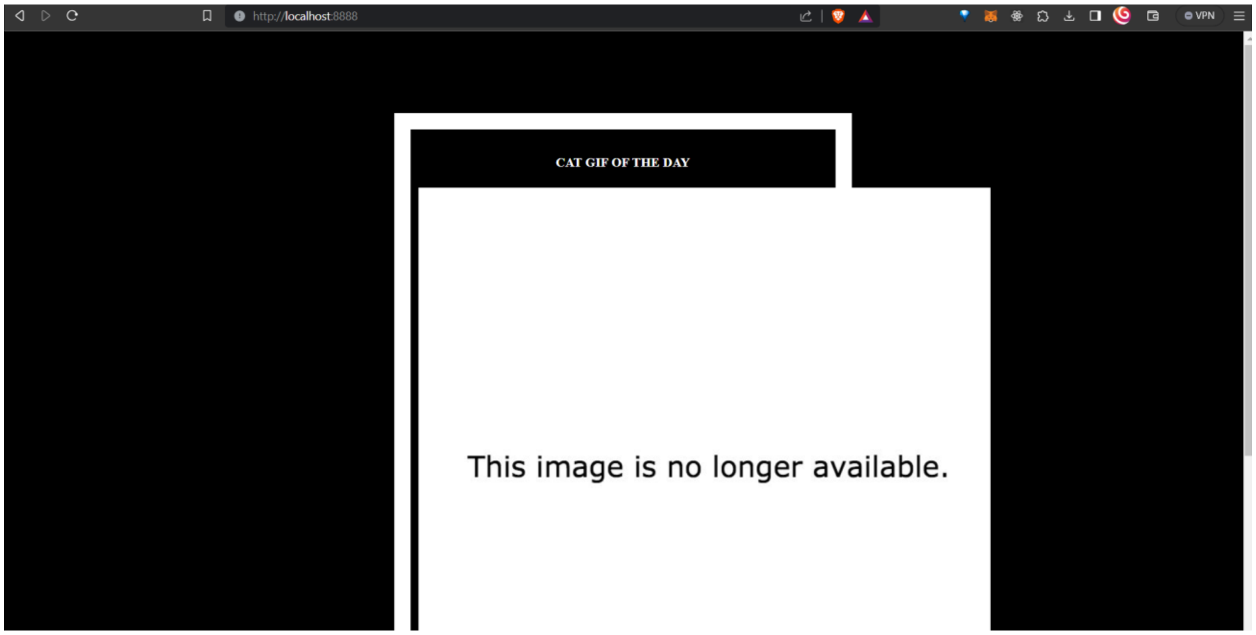
Now you need run the image

```
$ docker run -p 8888:5000 --name myfirstapp angelordonez/myfirstapp
```



```
[+] Building 6.8s (12/12) FINISHED                                                                   docker:default
 => [internal] load build definition from Dockerfile                                                           0.0s
 => => transferring dockerfile: 540B                                                                           0.0s
 => [internal] load .dockerignore                                                                              0.0s
 => => transferring context: 2B                                                                                0.0s
 => [internal] load metadata for docker.io/library/alpine:3.5                                                  1.4s
 => [auth] library/alpine:pull token for registry-1.docker.io                                                  0.0s
 => [1/6] FROM docker.io/library/alpine:3.5@sha256:66952b313e51c3bd1987d7c4ddf5dba9bc0fb6e524eed2448fa660246b3e76ec  0.2s
 => => resolve docker.io/library/alpine:3.5@sha256:66952b313e51c3bd1987d7c4ddf5dba9bc0fb6e524eed2448fa660246b3e76ec  0.0s
 => => sha256:66952b313e51c3bd1987d7c4ddf5dba9bc0fb6e524eed2448fa660246b3e76ec 433B / 433B                     0.0s
 => => sha256:f7d2b5725685826823bc6b154c0de02832e5e6daf7dc25a00ab00f1158fabfc8 528B / 528B                     0.0s
 => => sha256:f80194ae2e0ccf0f098baa6b981396dfbffb16e6476164af72158577a7de2dd9 1.51kB / 1.51kB                 0.0s
 => => sha256:8cae0e1ac61cead281f41115cc0ebd39117f7e54dffc8fd5e05a7590dca3cd4e 1.97MB / 1.97MB                 0.1s
 => => extracting sha256:8cae0e1ac61cead281f41115cc0ebd39117f7e54dffc8fd5e05a7590dca3cd4e                      0.0s
 => [internal] load build context                                                                              0.0s
 => => transferring context: 2.77kB                                                                            0.0s
 => [2/6] RUN apk add --update py2-pip                                                                         1.9s
 => [3/6] COPY requirements.txt /usr/src/app/                                                                  0.0s
 => [4/6] RUN pip install --no-cache-dir -r /usr/src/app/requirements.txt                                      2.7s
 => [5/6] COPY app.py /usr/src/app/                                                                            0.0s
 => [6/6] COPY templates/index.html /usr/src/app/templates/                                                    0.0s
 => exporting to image                                                                                         0.4s
 => => exporting layers                                                                                        0.4s
 => => writing image sha256:32244ee458b7254ca2806ecdc2140f295874bb3831ddb75855406b971ed21857                   0.0s
 => => naming to docker.io/angelordonez/myfirstapp                                                             0.0s

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$ docker run -p 8888:5000 --name myfirstapp angelordonez/myfirstapp
 * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
```

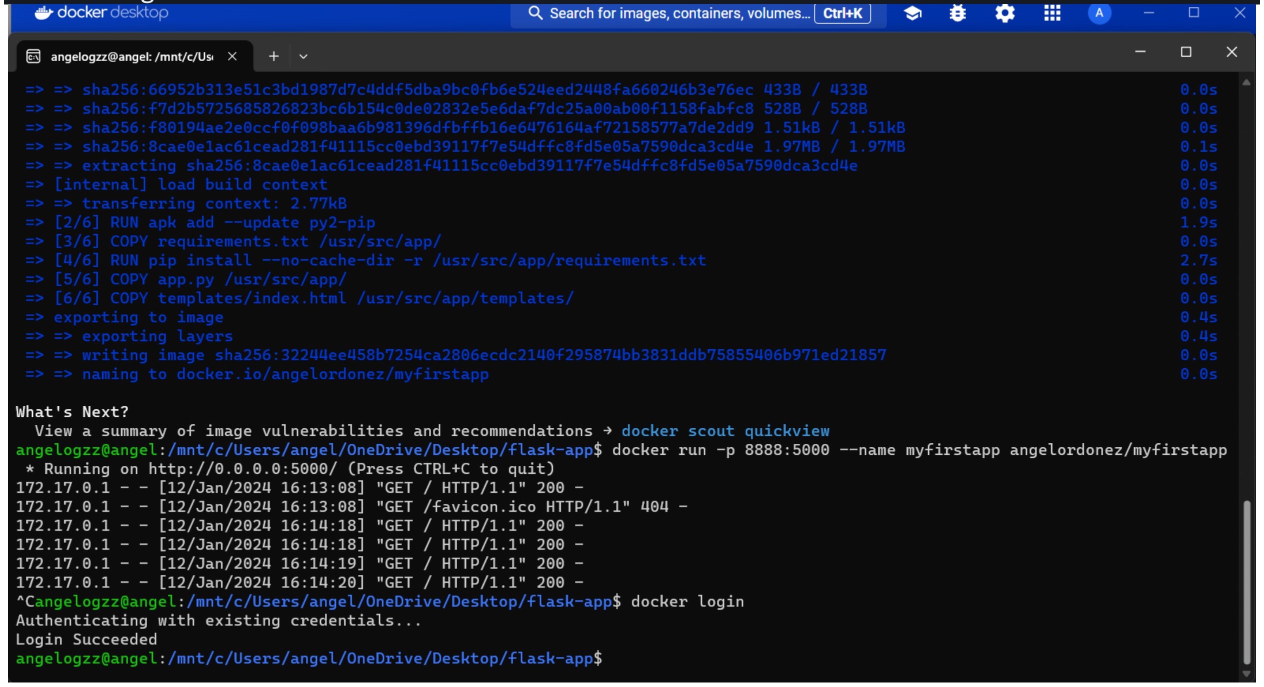
Go to http://localhost:8888



2.3.4 Push your image

Now to push the image we nee to login in docker

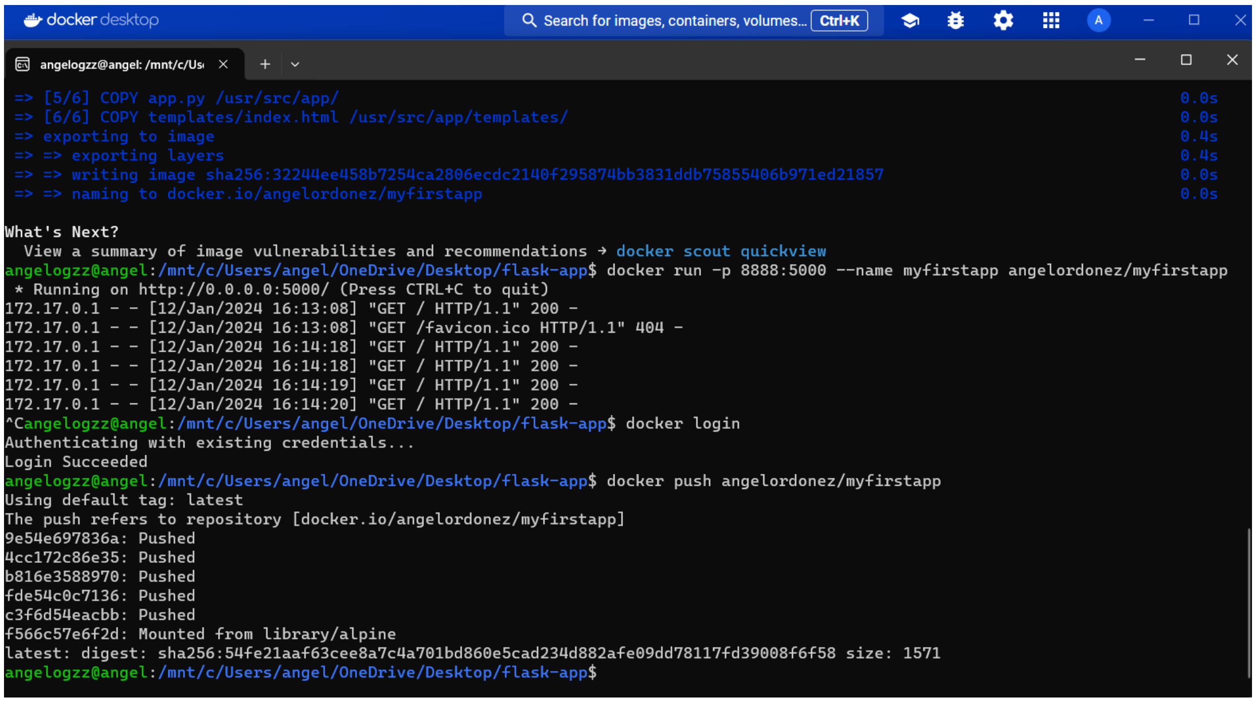
```
docker login
```


docker desktop
Search for images, containers, volumes... Ctrl+K
angelogzz@angel: /mnt/c/Us

```
 => => sha256:66952b313e51c3bd1987d7c4ddf5dba9bc0fb6e524eed2448fa660246b3e76ec 433B / 433B                          0.0s
 => => sha256:f7d2b5725685826823bc6b154c0de02832e5e6daf7dc25a00ab00f1158fabfc8 528B / 528B                          0.0s
 => => sha256:f80194ae2e0ccf0f098baa6b981396dfbffb16e6476164af72158577a7de2dd9 1.51kB / 1.51kB                      0.0s
 => => sha256:8cae0e1ac61cead281f41115cc0ebd39117f7e54dffc8fd5e05a7590dca3cd4e 1.97MB / 1.97MB                      0.1s
 => => extracting sha256:8cae0e1ac61cead281f41115cc0ebd39117f7e54dffc8fd5e05a7590dca3cd4e                           0.0s
 => [internal] load build context                                                                                   0.0s
 => => transferring context: 2.77kB                                                                                 0.0s
 => [2/6] RUN apk add --update py2-pip                                                                              1.9s
 => [3/6] COPY requirements.txt /usr/src/app/                                                                       0.0s
 => [4/6] RUN pip install --no-cache-dir -r /usr/src/app/requirements.txt                                           2.7s
 => [5/6] COPY app.py /usr/src/app/                                                                                 0.0s
 => [6/6] COPY templates/index.html /usr/src/app/templates/                                                         0.0s
 => exporting to image                                                                                              0.4s
 => => exporting layers                                                                                             0.4s
 => => writing image sha256:32244ee458b7254ca2806ecdc2140f295874bb3831ddb75855406b971ed21857                        0.0s
 => => naming to docker.io/angelordonez/myfirstapp                                                                  0.0s

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$ docker run -p 8888:5000 --name myfirstapp angelordonez/myfirstapp
 * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
172.17.0.1 - - [12/Jan/2024 16:13:08] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Jan/2024 16:13:08] "GET /favicon.ico HTTP/1.1" 404 -
172.17.0.1 - - [12/Jan/2024 16:14:18] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Jan/2024 16:14:18] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Jan/2024 16:14:19] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Jan/2024 16:14:20] "GET / HTTP/1.1" 200 -
^Cangelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$ docker login
Authenticating with existing credentials...
Login Succeeded
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$
```

Now we need to try with this command

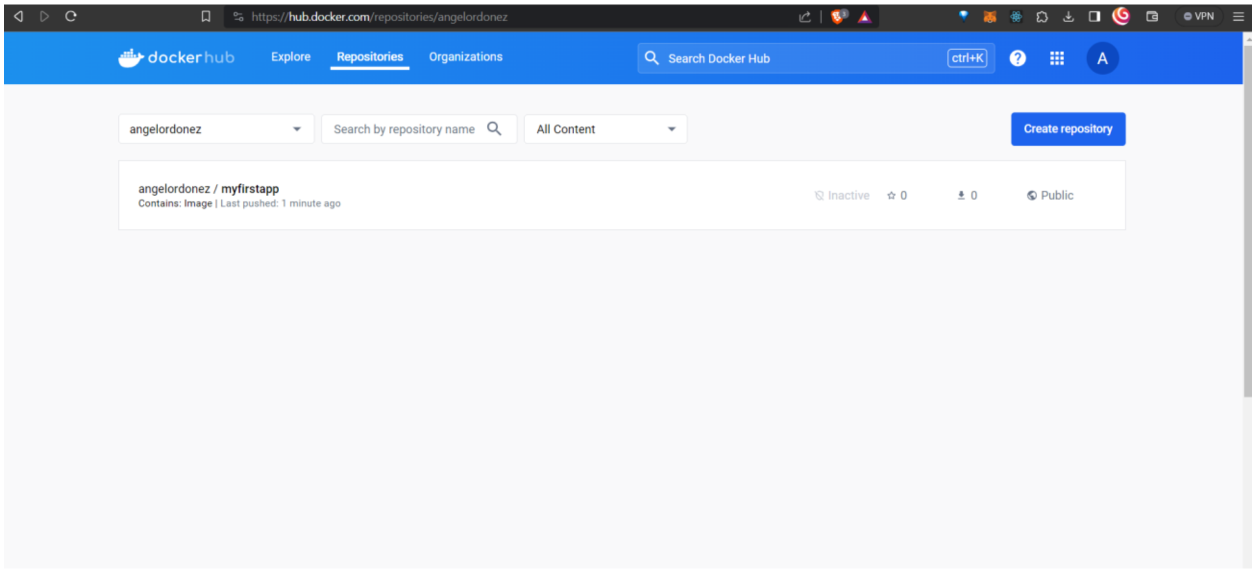
```
docker push angelordonez/myfirstapp
```


docker desktop
Search for images, containers, volumes... Ctrl+K
angelogzz@angel: /mnt/c/Us

```
=> [5/6] COPY app.py /usr/src/app/                                                                     0.0s
=> [6/6] COPY templates/index.html /usr/src/app/templates/                                             0.0s
=> exporting to image                                                                                  0.4s
=> => exporting layers                                                                                 0.4s
=> => writing image sha256:32244ee458b7254ca2806ecdc2140f295874bb3831ddb75855406b971ed21857            0.0s
=> => naming to docker.io/angelordonez/myfirstapp                                                      0.0s

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$ docker run -p 8888:5000 --name myfirstapp angelordonez/myfirstapp
 * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
172.17.0.1 - - [12/Jan/2024 16:13:08] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Jan/2024 16:13:08] "GET /favicon.ico HTTP/1.1" 404 -
172.17.0.1 - - [12/Jan/2024 16:14:18] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Jan/2024 16:14:18] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Jan/2024 16:14:19] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Jan/2024 16:14:20] "GET / HTTP/1.1" 200 -
^Cangelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$ docker login
Authenticating with existing credentials...
Login Succeeded
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$ docker push angelordonez/myfirstapp
Using default tag: latest
The push refers to repository [docker.io/angelordonez/myfirstapp]
9e54e697836a: Pushed
4cc172c86e35: Pushed
b816e3588970: Pushed
fde54c0c7136: Pushed
c3f6d54eacbb: Pushed
f566c57e6f2d: Mounted from library/alpine
latest: digest: sha256:54fe21aaf63cee8a7c4a701bd860e5cad234d882afe09dd78117fd39008f6f58 size: 1571
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$
```

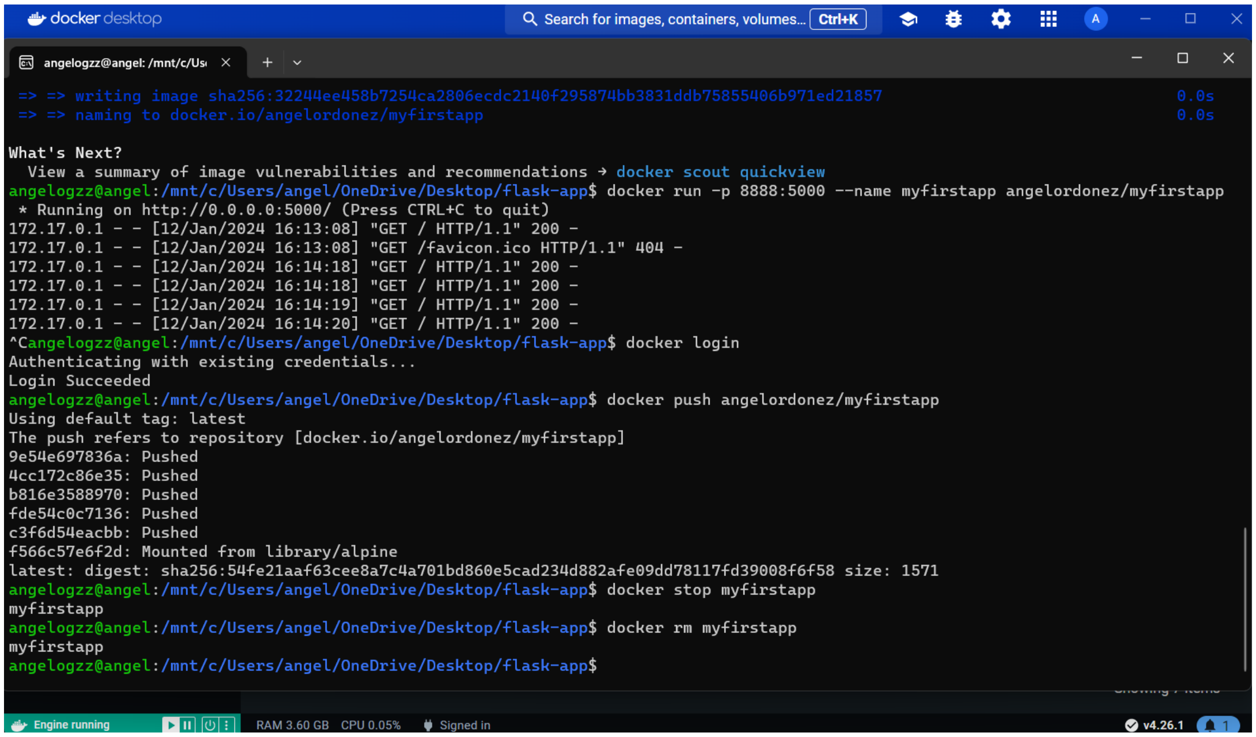
And we can see all the repositories in docker hub and we can click myfirstapp to see it particularly.

https://hub.docker.com/repository/docker/angelordonez/myfirstapp


https://hub.docker.com/repositories/angelordonez
docker hub
Explore
Repositories
Organizations
Search Docker Hub
ctrl+K
angelordonez
Search by repository name
All Content
Create repository
angelordonez / myfirstapp
Contains: Image | Last pushed: 1 minute ago
Inactive
0
0
Public

Now is time to stop and remove it.

```
$ docker stop myfirstapp
$ docker rm myfirstapp
```

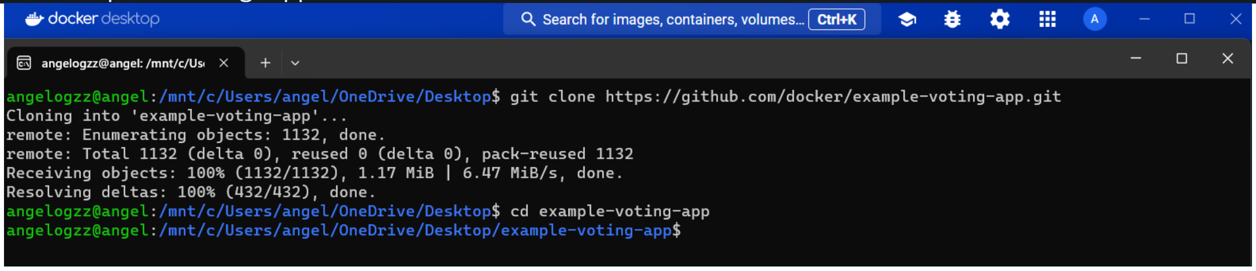


```
 => => writing image sha256:32244ee458b7254ca2806ecdc2140f295874bb3831ddb75855406b971ed21857      0.0s
 => => naming to docker.io/angelordonez/myfirstapp                                                 0.0s

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$ docker run -p 8888:5000 --name myfirstapp angelordonez/myfirstapp
 * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
172.17.0.1 - - [12/Jan/2024 16:13:08] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Jan/2024 16:13:08] "GET /favicon.ico HTTP/1.1" 404 -
172.17.0.1 - - [12/Jan/2024 16:14:18] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Jan/2024 16:14:18] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Jan/2024 16:14:19] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Jan/2024 16:14:20] "GET / HTTP/1.1" 200 -
^Cangelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$ docker login
Authenticating with existing credentials...
Login Succeeded
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$ docker push angelordonez/myfirstapp
Using default tag: latest
The push refers to repository [docker.io/angelordonez/myfirstapp]
9e54e697836a: Pushed
4cc172c86e35: Pushed
b816e3588970: Pushed
fde54c0c7136: Pushed
c3f6d54eacbb: Pushed
f566c57e6f2d: Mounted from library/alpine
latest: digest: sha256:54fe21aaf63cee8a7c4a701bd860e5cad234d882afe09dd78117fd39008f6f58 size: 1571
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$ docker stop myfirstapp
myfirstapp
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$ docker rm myfirstapp
myfirstapp
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/flask-app$
```

3.0 Deploying an app to a Swarm

Clone the repository onto your machine and cd into the directory:

```
git clone https://github.com/docker/example-voting-app.git
cd example-voting-app
```

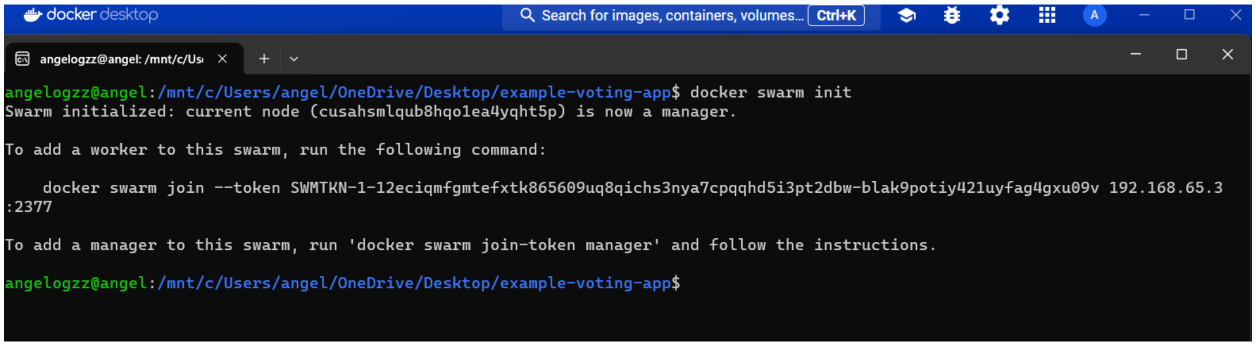


```
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop$ git clone https://github.com/docker/example-voting-app.git
Cloning into 'example-voting-app'...
remote: Enumerating objects: 1132, done.
remote: Total 1132 (delta 0), reused 0 (delta 0), pack-reused 1132
Receiving objects: 100% (1132/1132), 1.17 MiB | 6.47 MiB/s, done.
Resolving deltas: 100% (432/432), done.
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop$ cd example-voting-app
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/example-voting-app$
```

3.1 Deploying the app

First, create a Swarm.

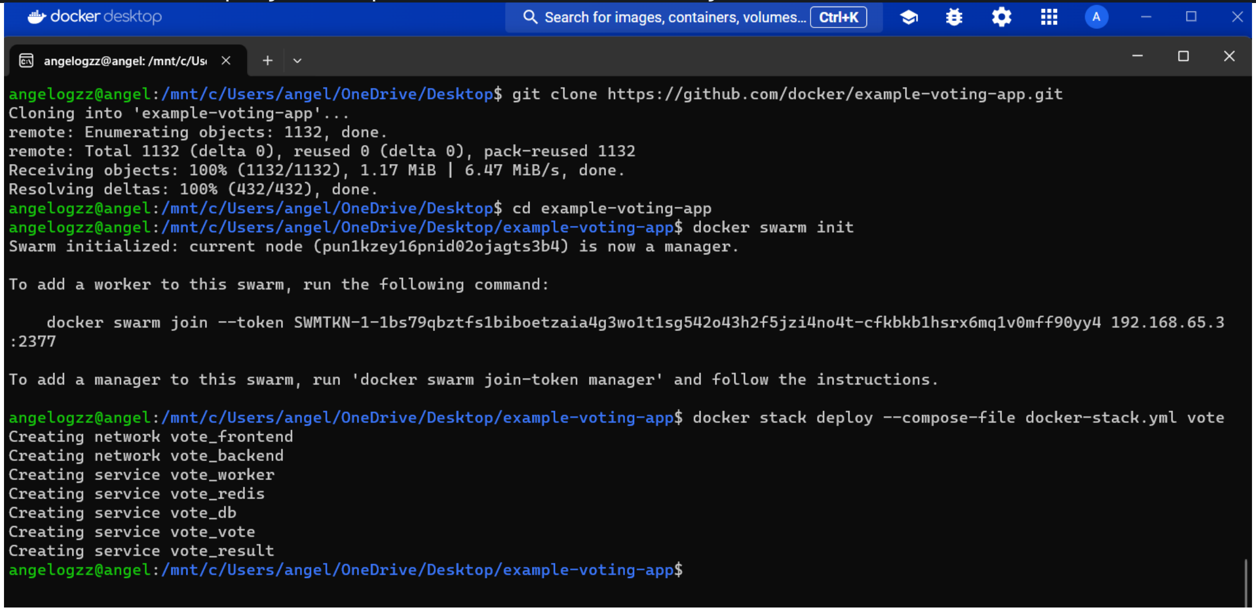
```
docker swarm init
```


docker desktop
Search for images, containers, volumes... Ctrl+K
angelogzz@angel: /mnt/c/Us

```
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/example-voting-app$ docker swarm init
Swarm initialized: current node (cusahsmlqub8hqo1ea4yqht5p) is now a manager.

To add a worker to this swarm, run the following command:

    docker swarm join --token SWMTKN-1-12eciqmfgmtefxtk865609uq8qichs3nya7cpqqhd5i3pt2dbw-blak9potiy421uyfag4gxu09v 192.168.65.3
:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.

angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/example-voting-app$
```

Now we need to try with this command to deploy.

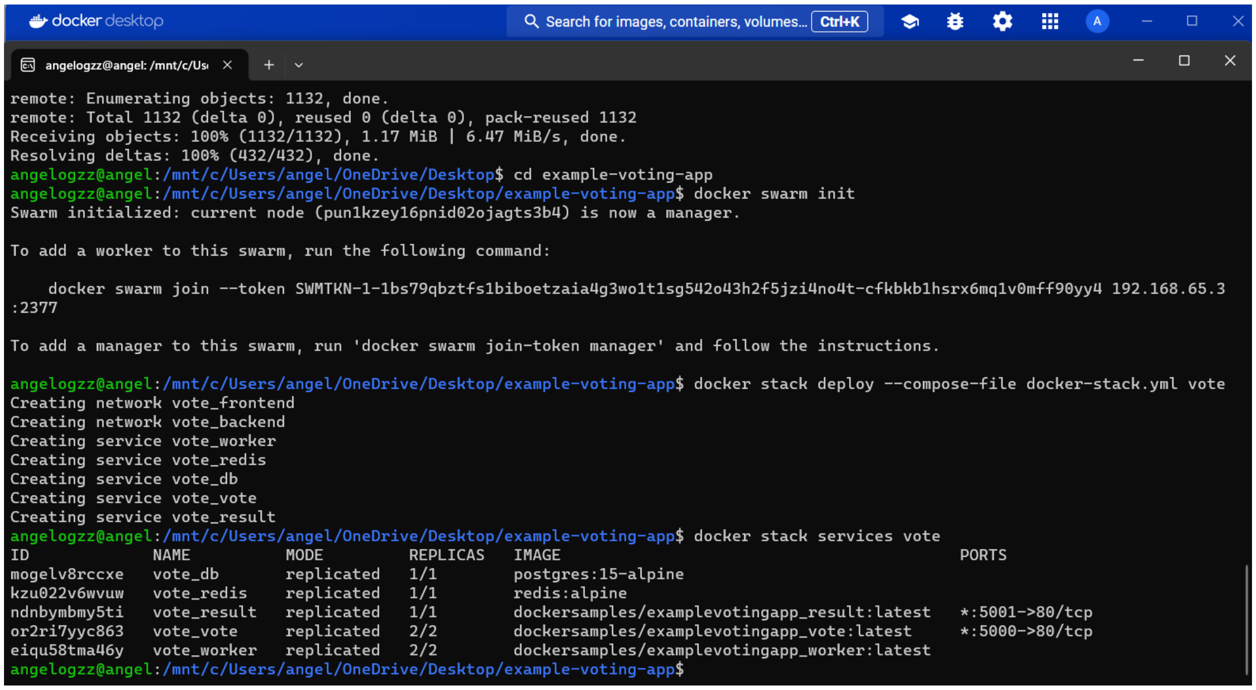
```
docker stack deploy --compose-file docker-stack.yml vote
```


docker desktop
Search for images, containers, volumes... Ctrl+K
angelogzz@angel: /mnt/c/Us

```
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop$ git clone https://github.com/docker/example-voting-app.git
Cloning into 'example-voting-app'...
remote: Enumerating objects: 1132, done.
remote: Total 1132 (delta 0), reused 0 (delta 0), pack-reused 1132
Receiving objects: 100% (1132/1132), 1.17 MiB | 6.47 MiB/s, done.
Resolving deltas: 100% (432/432), done.
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop$ cd example-voting-app
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/example-voting-app$ docker swarm init
Swarm initialized: current node (pun1kzey16pnid02ojagts3b4) is now a manager.

To add a worker to this swarm, run the following command:

    docker swarm join --token SWMTKN-1-1bs79qbztfs1biboetzaia4g3wo1t1sg542o43h2f5jzi4no4t-cfkbkb1hsrx6mq1v0mff90yy4 192.168.65.3
:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.

angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/example-voting-app$ docker stack deploy --compose-file docker-stack.yml vote
Creating network vote_frontend
Creating network vote_backend
Creating service vote_worker
Creating service vote_redis
Creating service vote_db
Creating service vote_vote
Creating service vote_result
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/example-voting-app$
```

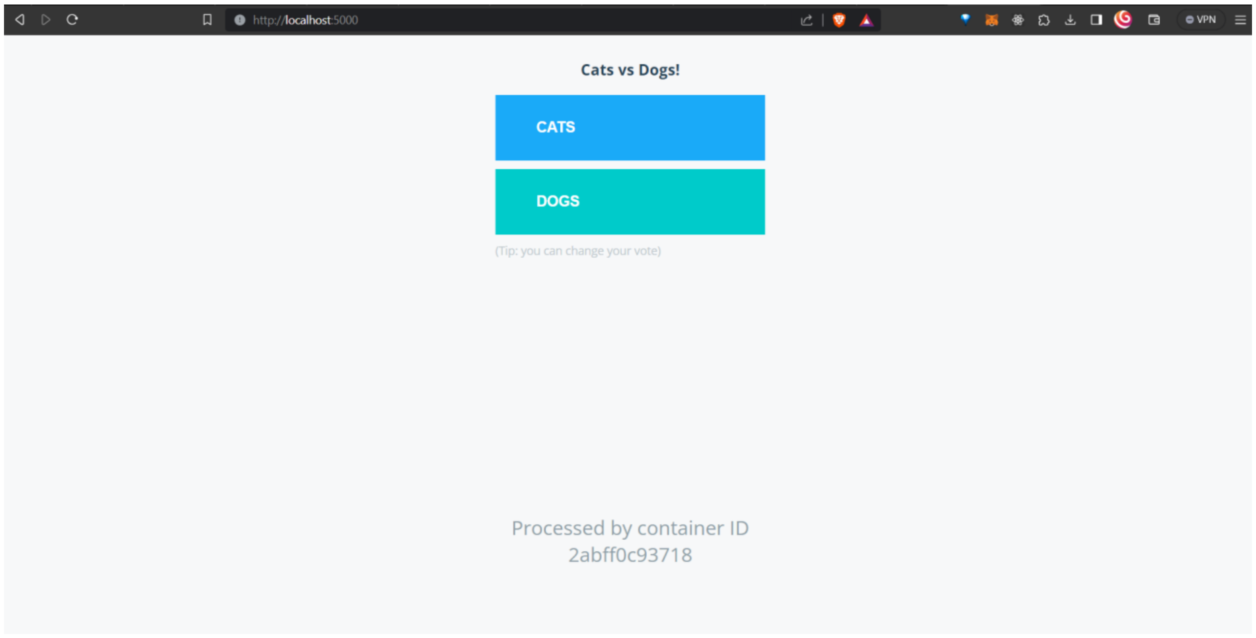
To verify your stack has deployed, use this command.

```
docker stack services vote
```

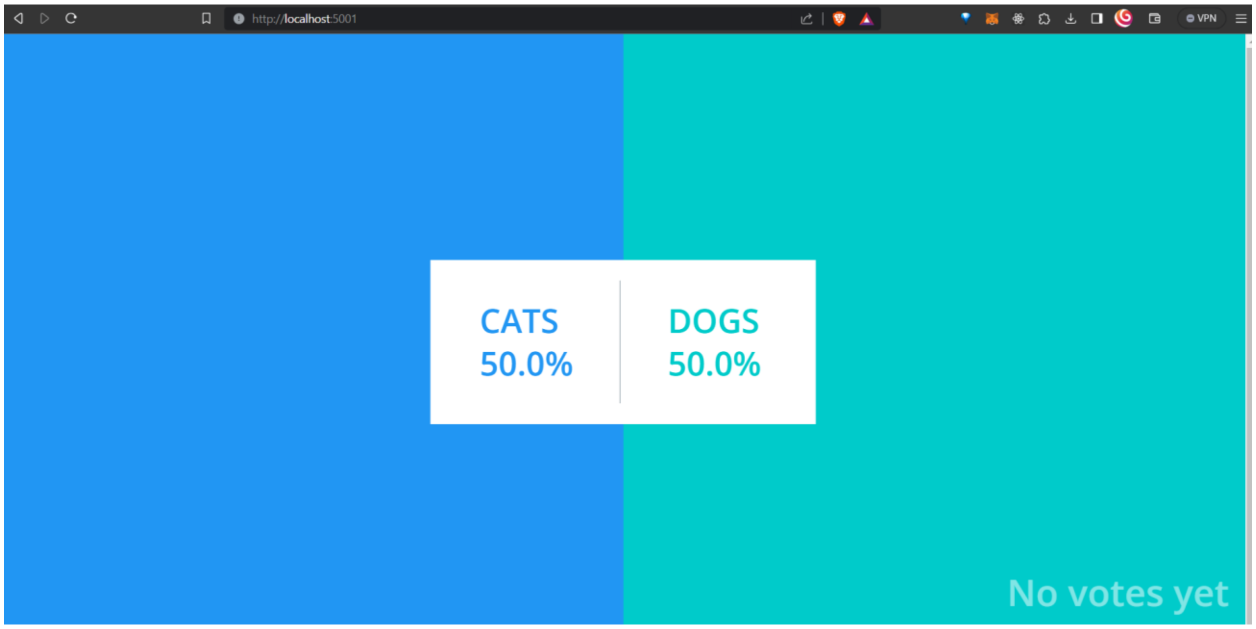


```
remote: Enumerating objects: 1132, done.
remote: Total 1132 (delta 0), reused 0 (delta 0), pack-reused 1132
Receiving objects: 100% (1132/1132), 1.17 MiB | 6.47 MiB/s, done.
Resolving deltas: 100% (432/432), done.
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop$ cd example-voting-app
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/example-voting-app$ docker swarm init
Swarm initialized: current node (pun1kzey16pnid02ojagts3b4) is now a manager.

To add a worker to this swarm, run the following command:

    docker swarm join --token SWMTKN-1-1bs79qbztfs1biboetzaia4g3wo1t1sg542o43h2f5jzi4no4t-cfkbkb1hsrx6mq1v0mff90yy4 192.168.65.3
:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.

angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/example-voting-app$ docker stack deploy --compose-file docker-stack.yml vote
Creating network vote_frontend
Creating network vote_backend
Creating service vote_worker
Creating service vote_redis
Creating service vote_db
Creating service vote_vote
Creating service vote_result
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/example-voting-app$ docker stack services vote
ID             NAME          MODE         REPLICAS   IMAGE                                          PORTS
mogelv8rccxe   vote_db       replicated   1/1        postgres:15-alpine
kzu022v6wvuw   vote_redis    replicated   1/1        redis:alpine
ndnbymbmy5ti   vote_result   replicated   1/1        dockersamples/examplevotingapp_result:latest   *:5001->80/tcp
or2ri7yyc863   vote_vote     replicated   2/2        dockersamples/examplevotingapp_vote:latest     *:5000->80/tcp
eiqu58tma46y   vote_worker   replicated   2/2        dockersamples/examplevotingapp_worker:latest
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/example-voting-app$
```

Go to http://localhost:5000/ to vote



And go to http://localhost:5001/ to see the results.

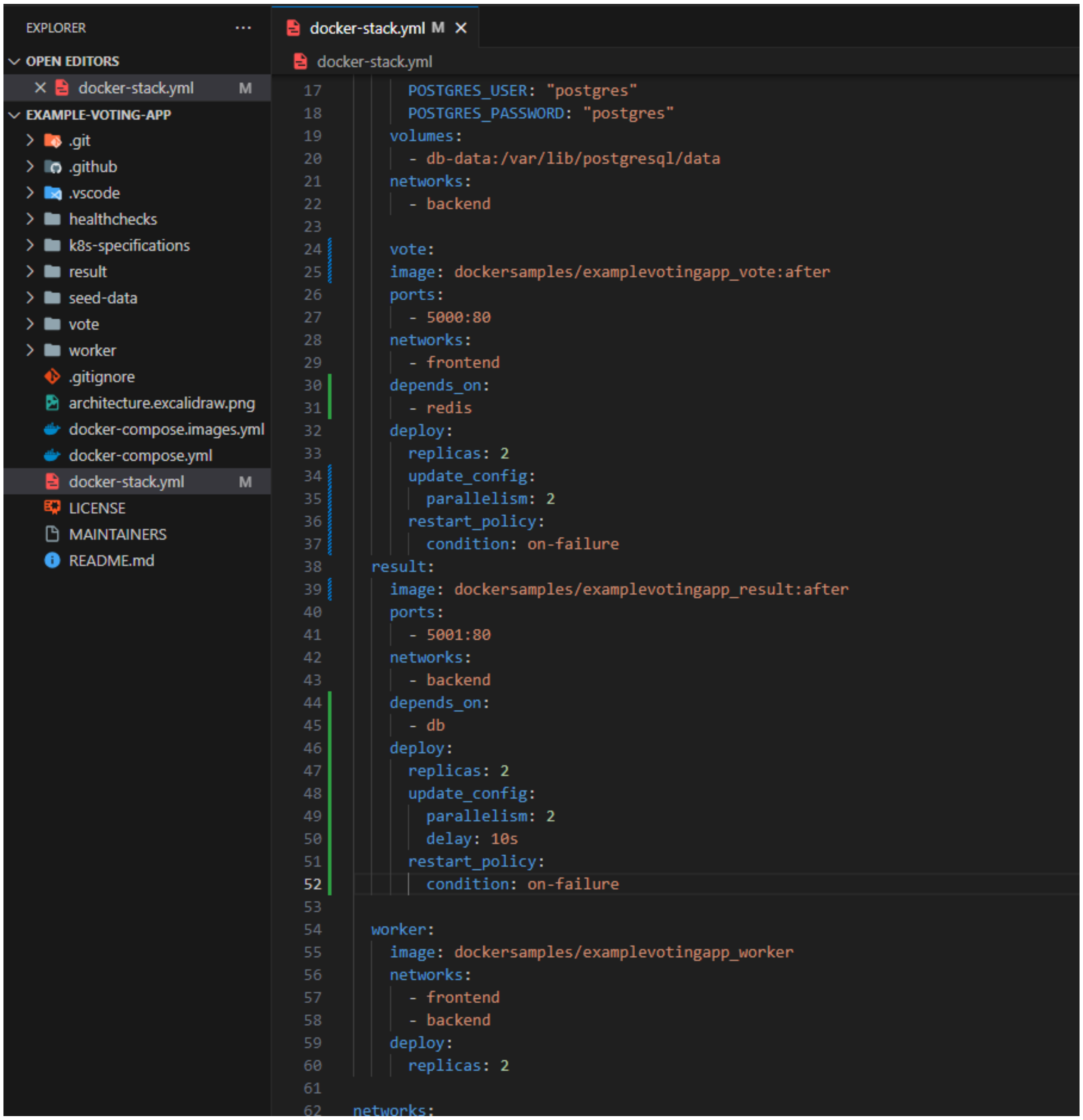


3.2 Customize the app

In this step, you will customize the app and redeploy it.

3.2.1 Change the images used

Going back to docker-stack.yml, change the vote and result images to use the after tag, so they look like this:

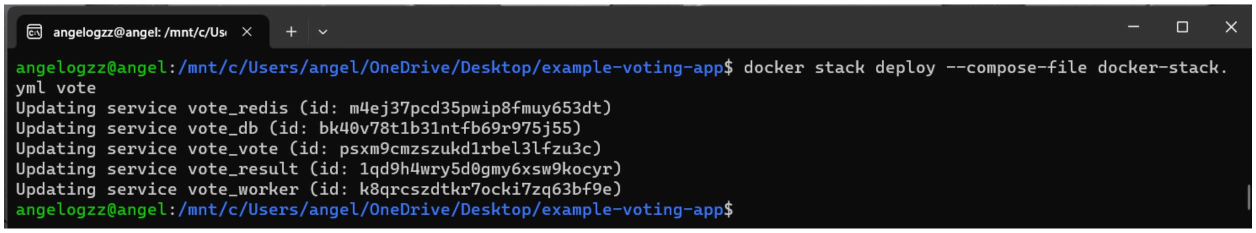


```yaml
      POSTGRES_USER: "postgres"
      POSTGRES_PASSWORD: "postgres"
    volumes:
      - db-data:/var/lib/postgresql/data
    networks:
      - backend

    vote:
    image: dockersamples/examplevotingapp_vote:after
    ports:
      - 5000:80
    networks:
      - frontend
    depends_on:
      - redis
    deploy:
      replicas: 2
      update_config:
        parallelism: 2
      restart_policy:
        condition: on-failure
  result:
    image: dockersamples/examplevotingapp_result:after
    ports:
      - 5001:80
    networks:
      - backend
    depends_on:
      - db
    deploy:
      replicas: 2
      update_config:
        parallelism: 2
        delay: 10s
      restart_policy:
        condition: on-failure

  worker:
    image: dockersamples/examplevotingapp_worker
    networks:
      - frontend
      - backend
    deploy:
      replicas: 2

networks:
```

3.2.3 Redeploy

Redeploy with this command

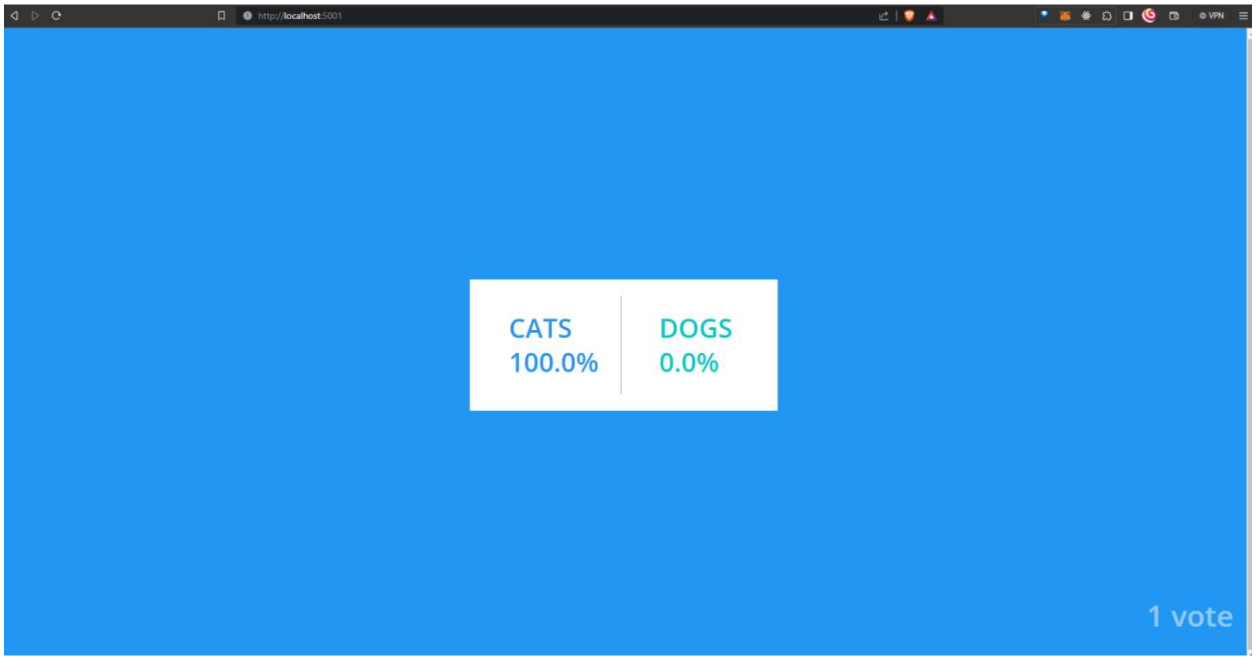
```
docker stack deploy --compose-file docker-stack.yml vote
```



```
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/example-voting-app$ docker stack deploy --compose-file docker-stack.
yml vote
Updating service vote_redis (id: m4ej37pcd35pwip8fmuy653dt)
Updating service vote_db (id: bk40v78t1b31ntfb69r975j55)
Updating service vote_vote (id: psxm9cmzszukd1rbel3lfzu3c)
Updating service vote_result (id: 1qd9h4wry5d0gmy6xsw9kocyr)
Updating service vote_worker (id: k8qrcszdtkr7ocki7zq63bf9e)
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/example-voting-app$
```

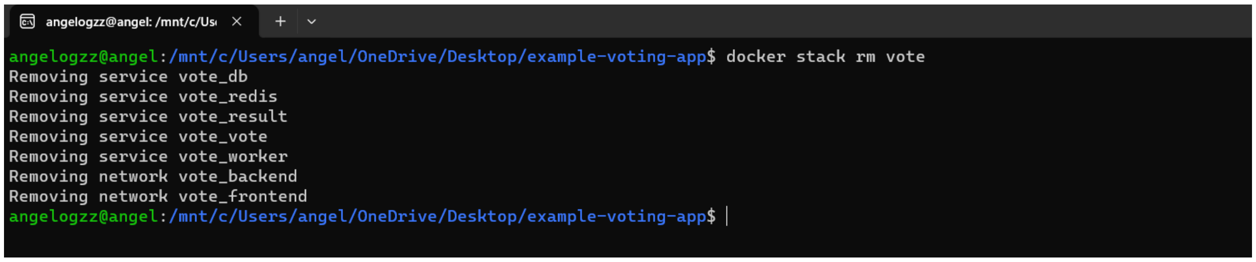
3.2.4 Another test run

Go to the URLs and see the new votes.



3.2.5 Remove the stack

And finally, remove the stack from the swarm



```
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/example-voting-app$ docker stack rm vote
Removing service vote_db
Removing service vote_redis
Removing service vote_result
Removing service vote_vote
Removing service vote_worker
Removing network vote_backend
Removing network vote_frontend
angelogzz@angel:/mnt/c/Users/angel/OneDrive/Desktop/example-voting-app$
```