

### Sommaire

1. Installation / Activation de WSL.....	2
2. Installation Ubuntu. ....	5
3. Installation Docker. ....	7
4. Premier test.....	12
5. Deuxième test. ....	15
6. Troisième test.....	18
7. Quatrième test. ....	23
8. Cinquième test : mode interactif d'utilisation d'un conteneur. ....	26
10. Première approche des volumes. ....	30
11. Gestion des volumes en écriture. ....	32
12. Utiliser un conteneur Docker et Visual Studio Code. ....	44
13. Monter une stack applicative avec Docker Compose. ....	56

# s1. Installation / Activation de WSL.

- Je tape wsl.exe pour installer la distribution Linux :

```
PS C:\WINDOWS\system32> wsl.exe --install
Téléchargement en cours : Sous-système Windows pour Linux 2.6.3
Installation en cours : Sous-système Windows pour Linux 2.6.3
Sous-système Windows pour Linux 2.6.3 a été installé.
Installation du composant facultatif Windows : VirtualMachinePlatform

Outil Gestion et maintenance des images de déploiement
Version : 10.0.26100.5074

Version de l'image : 10.0.26200.7922

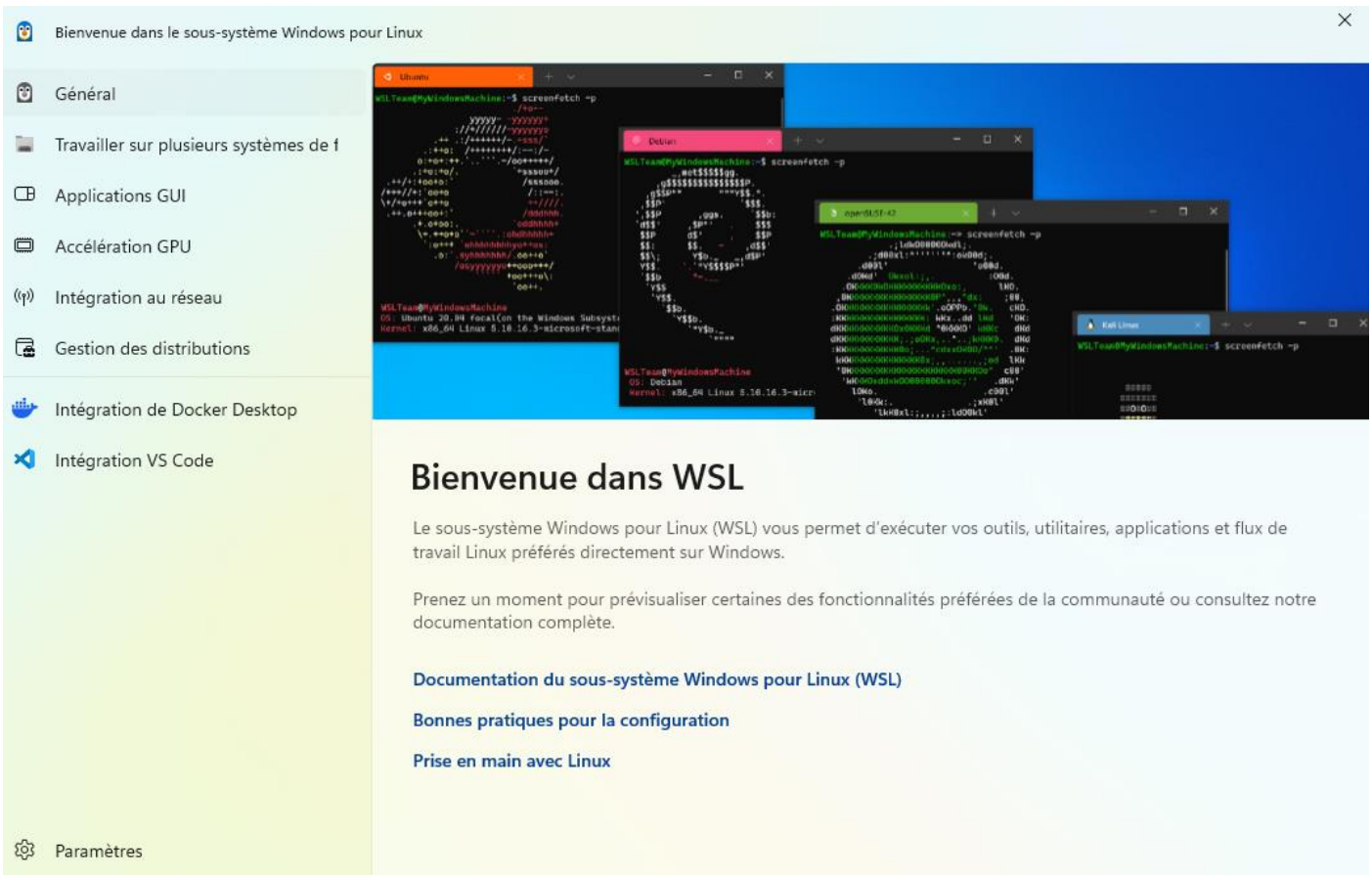
Activation de la ou des fonctionnalités
[=====100.0%=====]
L'opération a réussi.
L'opération demandée est réussie. Les modifications ne seront pas effectives avant que le système ne soit réamorcé.
Téléchargement : Ubuntu
Installation : Ubuntu
La distribution a été installée. Il peut être lancé via 'wsl.exe -d Ubuntu'
Lancement : Ubuntu...
```

- Je mets en paramètre --list --online pour lister les images trouvables en ligne, je constate la présence de l'image Ubuntu 24.04 LTS :

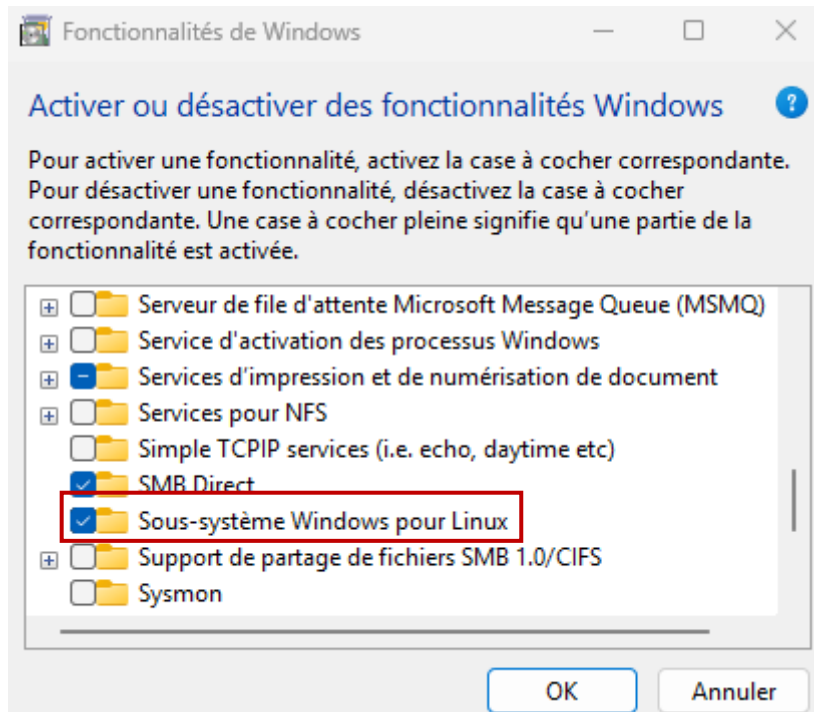
```
PS C:\WINDOWS\system32> wsl.exe --list --online
Voici la liste des distributions valides qui peuvent être installées.
Installez à l'aide de 'wsl.exe --install <Distro>'.

NAME                                FRIENDLY NAME
Ubuntu                               Ubuntu
Ubuntu-24.04                         Ubuntu 24.04 LTS
openSUSE-Tumbleweed                  openSUSE Tumbleweed
openSUSE-Leap-16.0                   openSUSE Leap 16.0
SUSE-Linux-Enterprise-15-SP7         SUSE Linux Enterprise 15 SP7
SUSE-Linux-Enterprise-16.0          SUSE Linux Enterprise 16.0
kali-linux                           Kali Linux Rolling
Debian                               Debian GNU/Linux
AlmaLinux-8                          AlmaLinux OS 8
AlmaLinux-9                          AlmaLinux OS 9
AlmaLinux-Kitten-10                 AlmaLinux OS Kitten 10
AlmaLinux-10                         AlmaLinux OS 10
archlinux                            Arch Linux
FedoraLinux-43                       Fedora Linux 43
FedoraLinux-42                       Fedora Linux 42
eLxR                                 eLxR 12.12.0.0 GNU/Linux
Ubuntu-20.04                         Ubuntu 20.04 LTS
Ubuntu-22.04                         Ubuntu 22.04 LTS
OracleLinux_7_9                      Oracle Linux 7.9
OracleLinux_8_10                     Oracle Linux 8.10
OracleLinux_9_5                      Oracle Linux 9.5
openSUSE-Leap-15.6                   openSUSE Leap 15.6
SUSE-Linux-Enterprise-15-SP6         SUSE Linux Enterprise 15 SP6
PS C:\WINDOWS\system32>
```

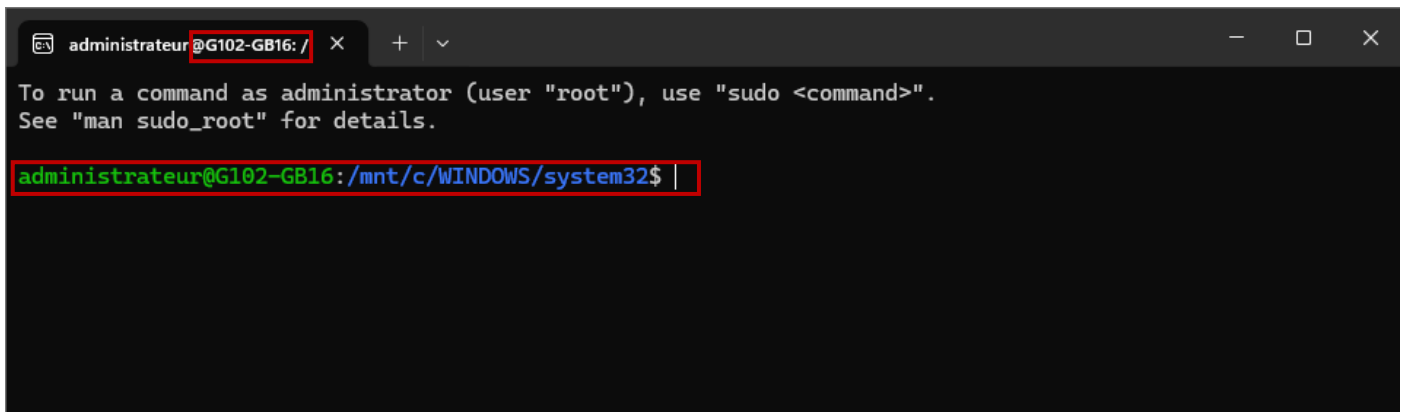
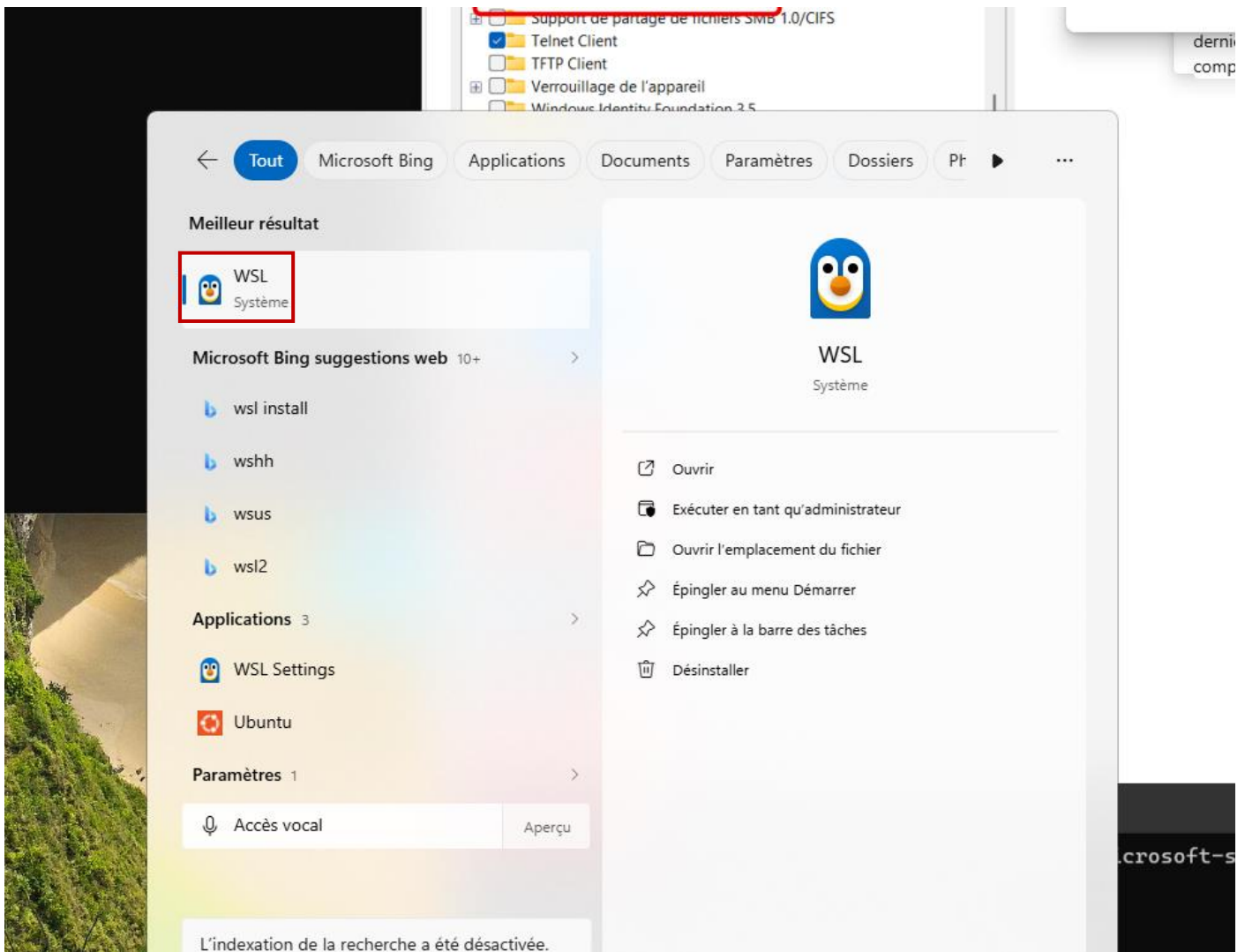
➤ J'ouvre WSL et je constate que je peux l'intégrer à VS Code :



➤ Dans les fonctionnalités Windows j'active Sous-système Windows pour Linux :

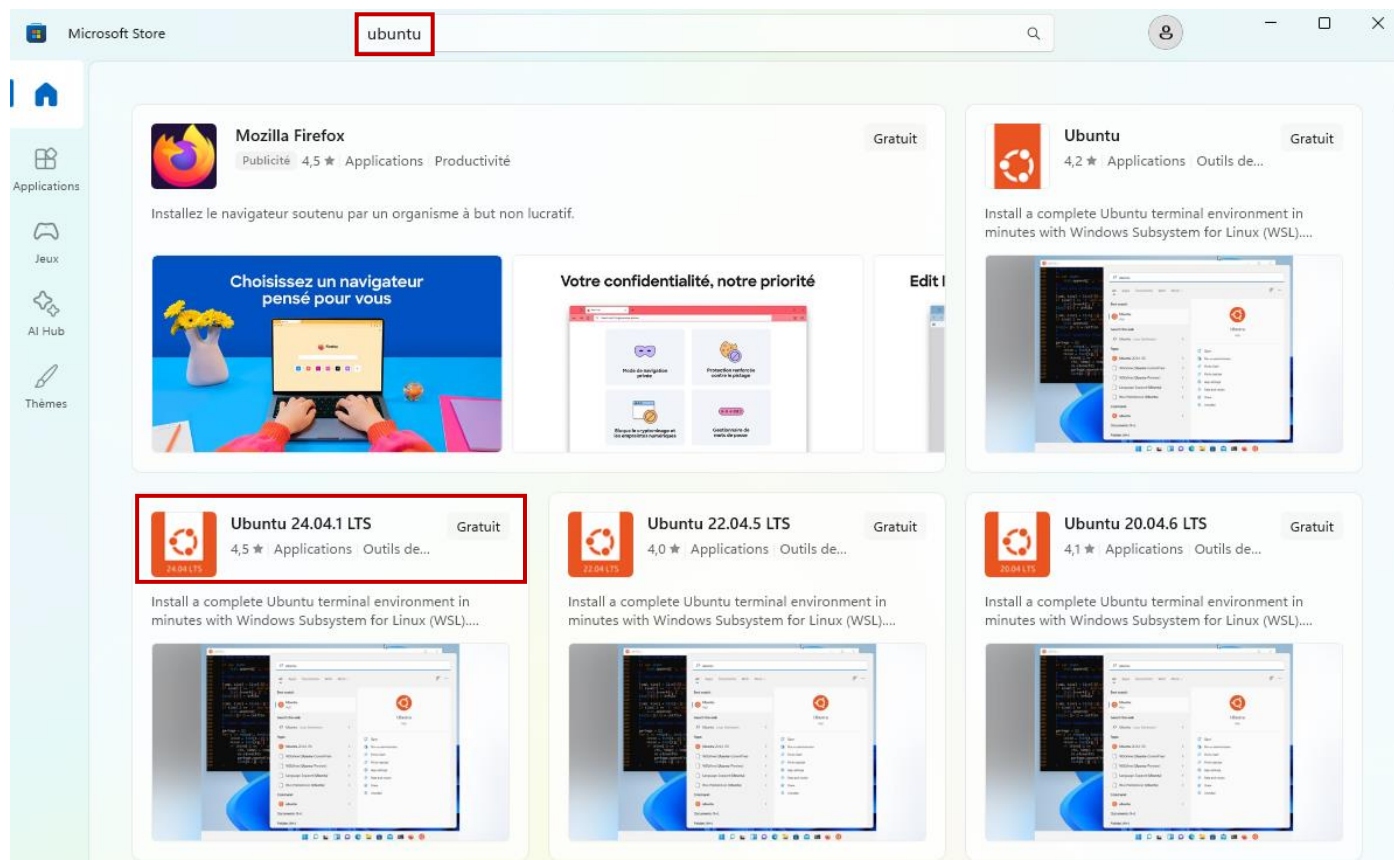


➤ J'ouvre maintenant l'autre WSL (le petit) :

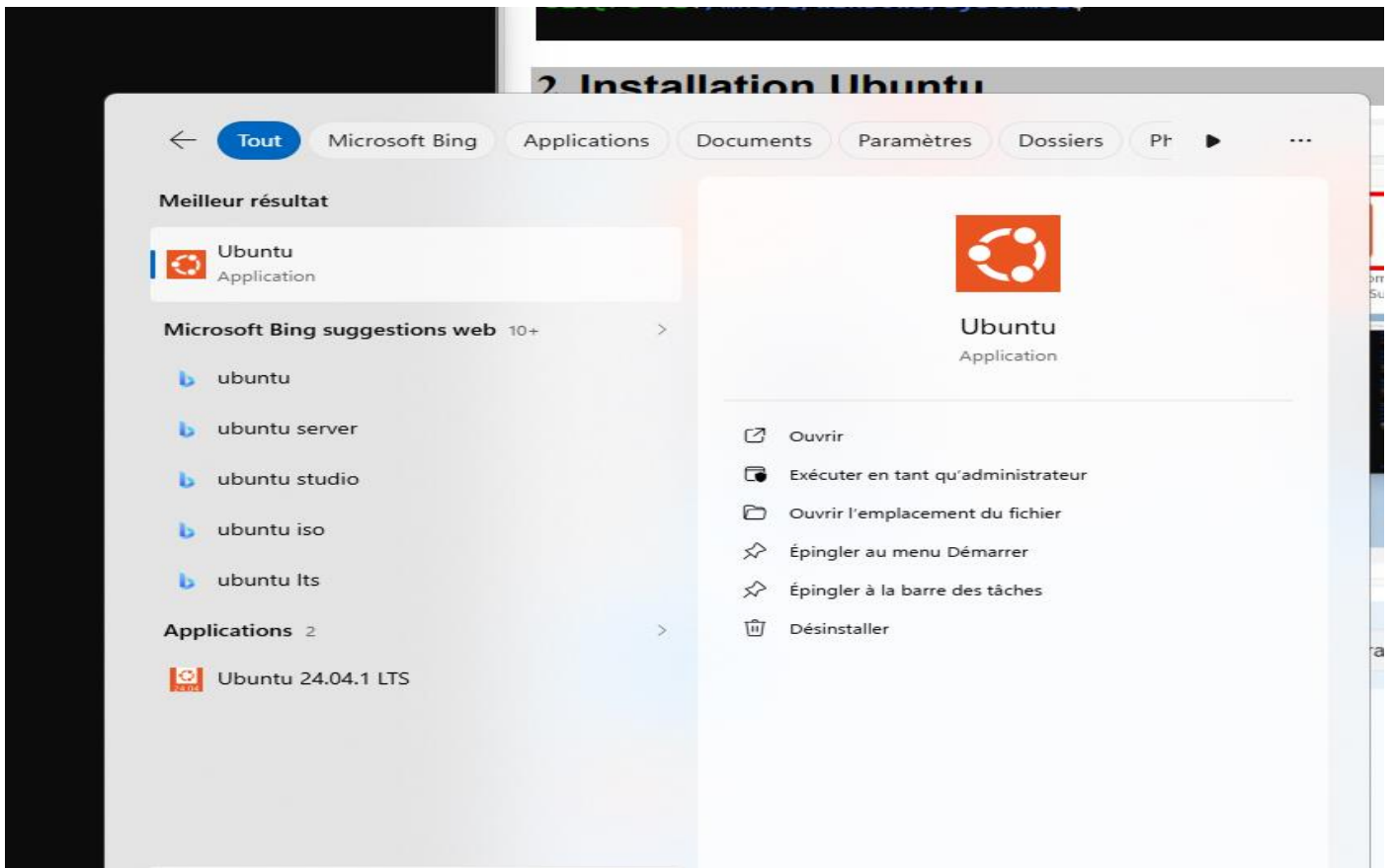


## 2. Installation Ubuntu.

- Je me rends dans le Microsoft store puis je tape Ubuntu dans la barre de recherche puis je clique sur le 24.04 LTS :



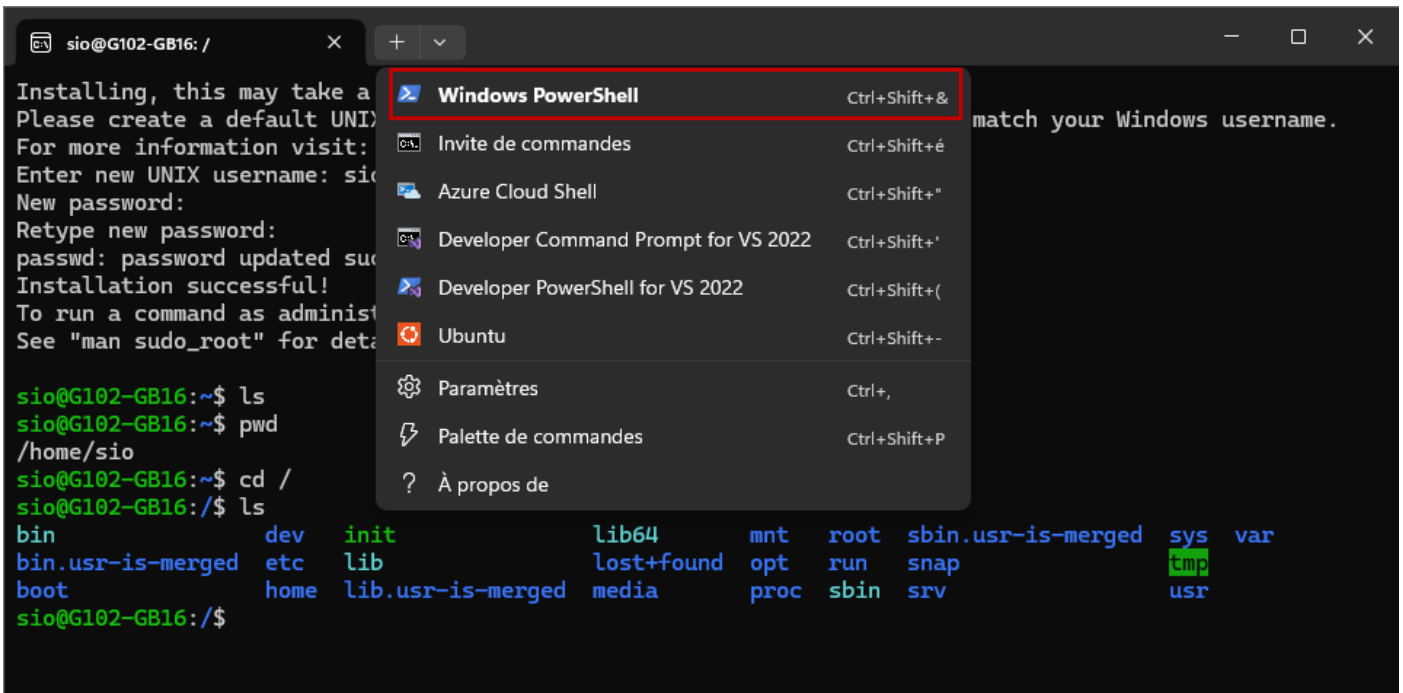
- Après l'installation je l'exécute :



```
sio@G102-GB16: /
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: sio
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.

sio@G102-GB16:~$ ls
sio@G102-GB16:~$ pwd
/home/sio
sio@G102-GB16:~$ cd /
sio@G102-GB16:/$ ls
bin          dev          init          lib64        mnt          root         sbin.usr-is-merged  sys  var
bin.usr-is-merged  etc          lib           lost+found   opt          run          snap                tmp
boot         home        lib.usr-is-merged  media        proc         sbin         srv                 usr
sio@G102-GB16:/$
```

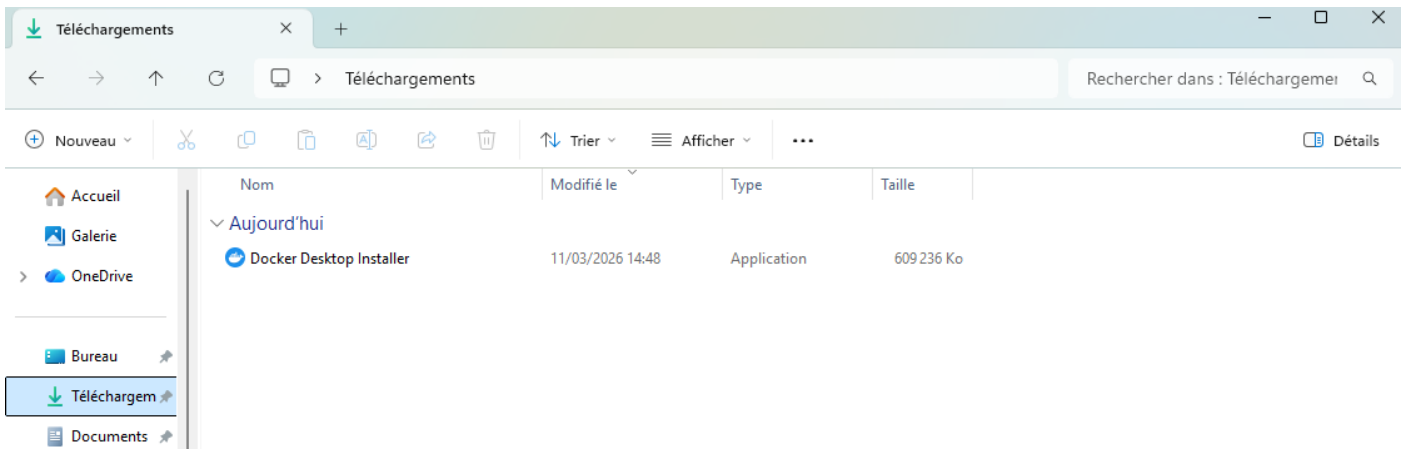
➤ Je choisis PowerShell :



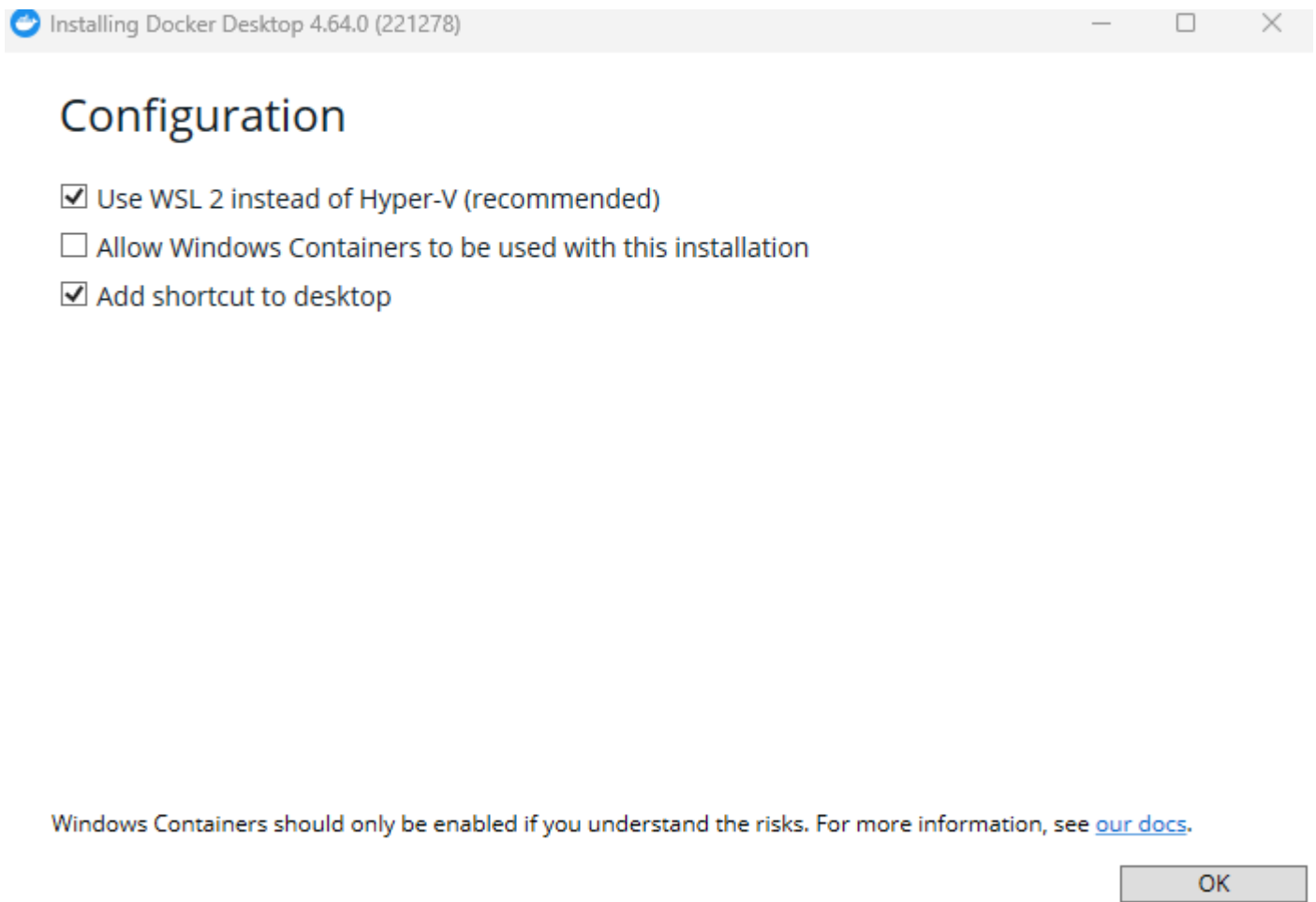
### 3. Installation Docker.

➤ Je me rends sur le site de Docker, j'installe l'exécutable sur ma machine :





➤ Je lance l'installation et je coche Use WSL 2 instead of Hyper-V :



# Docker Desktop 4.64.0

## Unpacking files...

```
Unpacking file: resources/cli-plugins/docker-buildx.exe
Unpacking file: resources/cli-plugins/docker-ai.exe
Unpacking file: resources/cli-plugins/docker-agent.exe
Unpacking file: resources/bin/local-sandboxsd.exe
Unpacking file: resources/bin/kubectl.exe
Unpacking file: resources/bin/hub-tool.exe
Unpacking file: resources/bin/extension-admin.exe
Unpacking file: resources/bin/docker.exe
Unpacking file: resources/bin/docker-credential-wincred.exe
Unpacking file: resources/bin/docker-credential-ecr-login.exe
Unpacking file: resources/bin/docker-credential-desktop.exe
Unpacking file: resources/bin/docker-compose
Unpacking file: resources/bin/docker
Unpacking file: resources/bin/cagent.exe
```

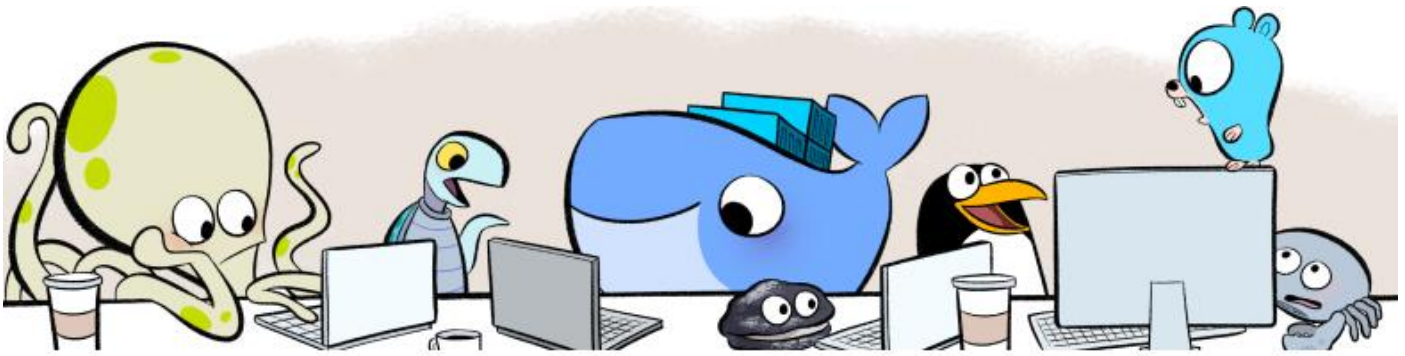
# Docker Desktop 4.64.0

## Installation succeeded

You must log out of Windows to complete installation.

Close and log out

- Docker est maintenant installé je l'exécute en tant qu'administrateur :



## Docker Subscription Service Agreement

By selecting **accept**, you agree to the [Subscription Service Agreement](#), the [Docker Data Processing Agreement](#), the [Data Privacy Policy](#) and the [Docker AI Supplemental Terms](#).

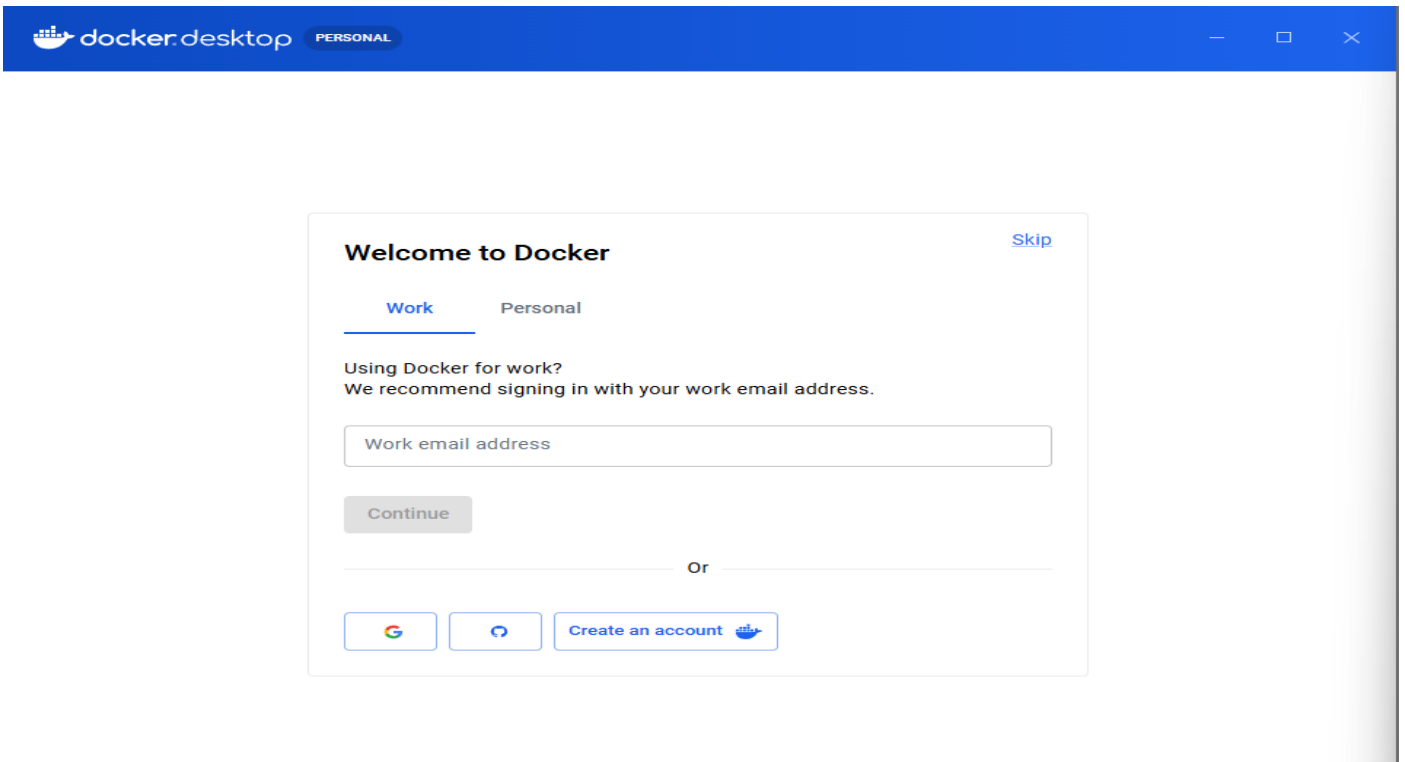
Commercial use of Docker Desktop at a company of more than 250 employees OR more than \$10 million in annual revenue requires a paid subscription (Pro, Team, or Business). [See subscription details](#)

[View Full Terms](#)

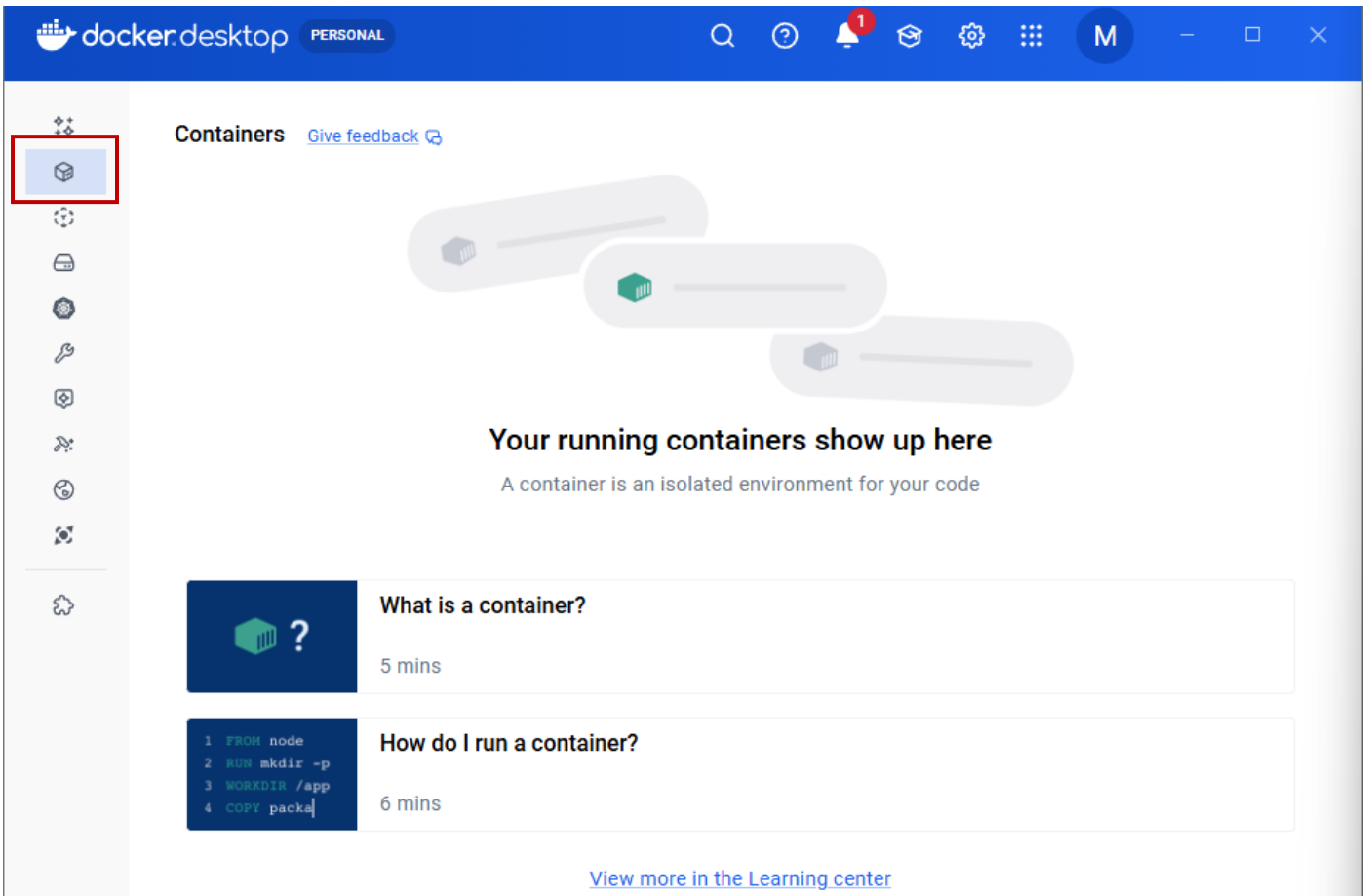
[Accept](#)

[Close](#)

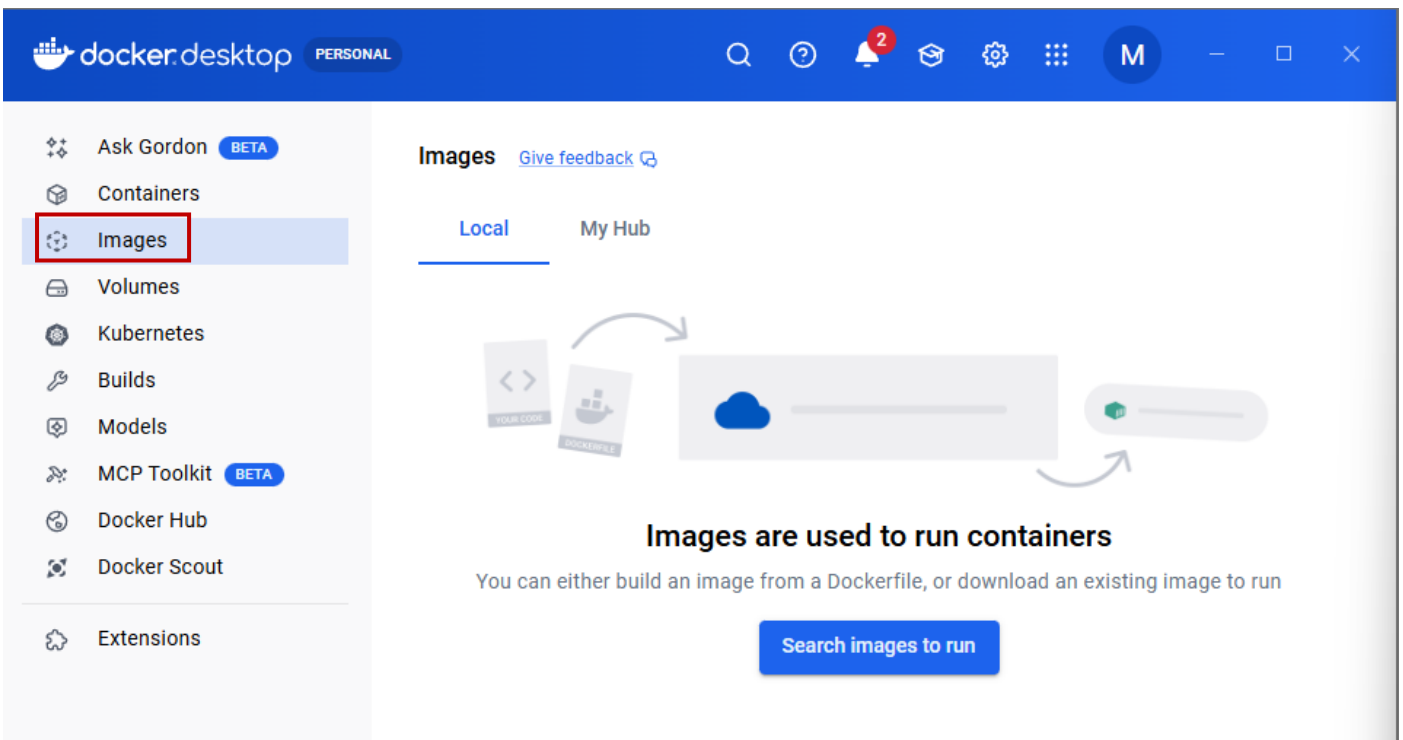
- Je créer un compte et je me connecte dessus :



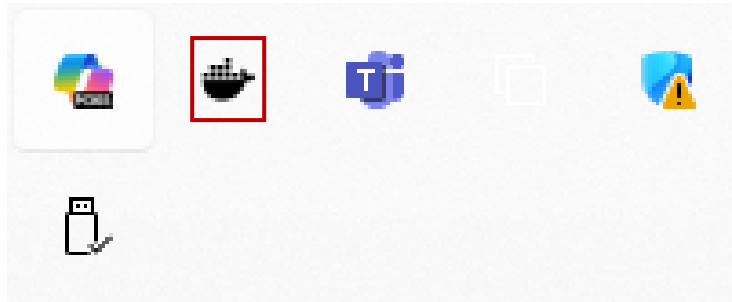
➤ Arriver sur l'interface je me rends dans l'onglet Containers :



➤ Ensuite je me rends dans l'onglet Images :



- Je remarque que Docker est présent dans les tâches en cours d'exécutions :



## 4. Premier test.

- Dans le terminal Ubuntu la commande `docker --version` ne fonctionne pas :

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

sio@G102-GB16:~$ docker --version

The command 'docker' could not be found in this WSL 2 distro.
We recommend to activate the WSL integration in Docker Desktop settings.

For details about using Docker Desktop with WSL 2, visit:
https://docs.docker.com/go/wsl2/

sio@G102-GB16:~$
```

- Tandis que sur un terminal PowerShell cela fonctionne très bien :

A screenshot of a Windows PowerShell terminal window. The window title is 'Windows PowerShell'. The terminal output shows the command `docker --version` being executed successfully, returning the text 'Docker version 29.2.1, build a5c7197'. The command and its output are highlighted with a red rectangular box. The terminal also displays the Windows PowerShell copyright notice and a link to the PowerShell website.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Installez la dernière version de PowerShell pour de nouvelles fonctionnalités et améliorations ! https
://aka.ms/PSWindows

PS C:\Users\Administrateur> docker --version
Docker version 29.2.1, build a5c7197
PS C:\Users\Administrateur>
```

➤ Cela fonctionne aussi sur une page de cmd normal :

```
Microsoft Windows [version 10.0.26200.7922]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\Administrateur> docker version
Client:
Version:      29.2.1
API version:  1.53
Go version:   go1.25.6
Git commit:   a5c7197
Built:        Mon Feb  2 17:20:16 2026
OS/Arch:      windows/amd64
Context:      desktop-linux

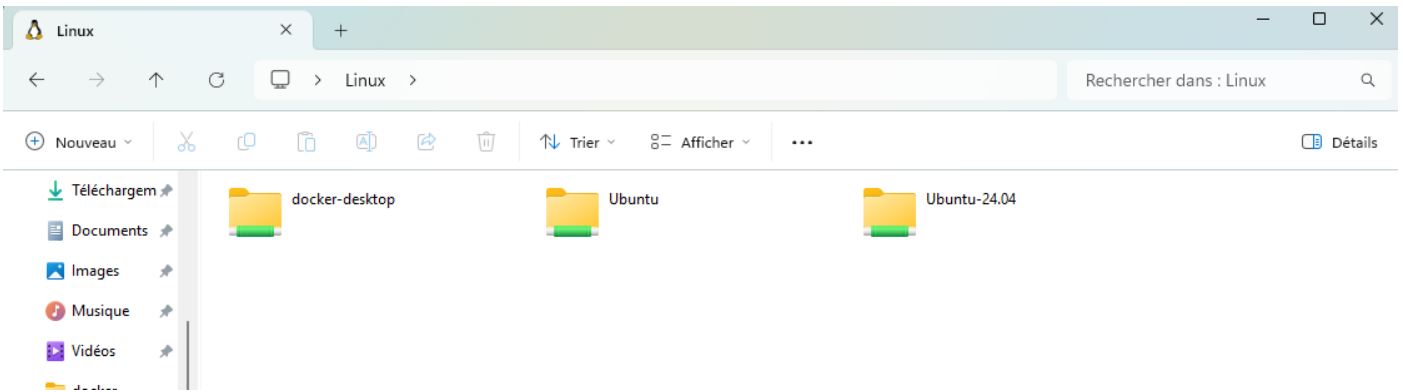
Server: Docker Desktop 4.64.0 (221278)
Engine:
Version:      29.2.1
API version:  1.53 (minimum version 1.44)
Go version:   go1.25.6
Git commit:   6bc6209
Built:        Mon Feb  2 17:17:24 2026
OS/Arch:      linux/amd64
Experimental: false
containerd:
Version:      v2.2.1
GitCommit:    dea7da592f5d1d2b7755e3a161be07f43fad8f75
runc:
Version:      1.3.4
GitCommit:    v1.3.4-0-gd6d73eb8
docker-init:
Version:      0.19.0
GitCommit:    de40ad0

C:\Users\Administrateur>
```

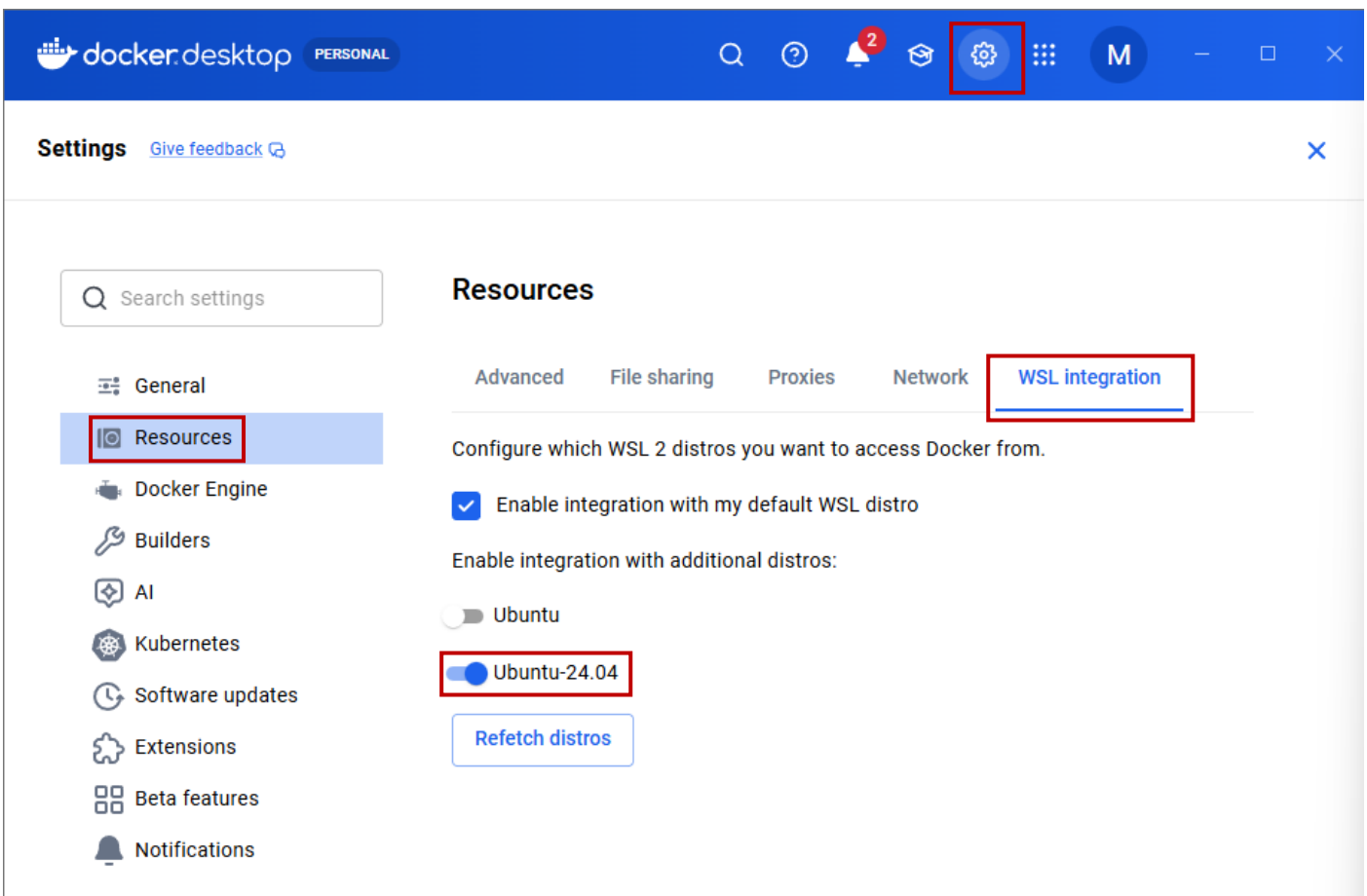
➤ Je liste les process de wsl avec des détails :

```
PS C:\Users\Administrateur> wsl -l -v
NAME          STATE      VERSION
* Ubuntu      Running    2
  Ubuntu-24.04 Running    2
  docker-desktop Running    2
PS C:\Users\Administrateur> wsl -l
Distributions du Sous-système Windows pour Linux :
Ubuntu (par défaut)
Ubuntu-24.04
docker-desktop
PS C:\Users\Administrateur>
```

➤ Dans l'explorateur de fichier je remarque la présence du logo de Linux :



➤ Dans les paramètres de Docker je me rends dans l'onglet Ressources, puis dans WSL intégration et j'active Ubuntu 24.04 :



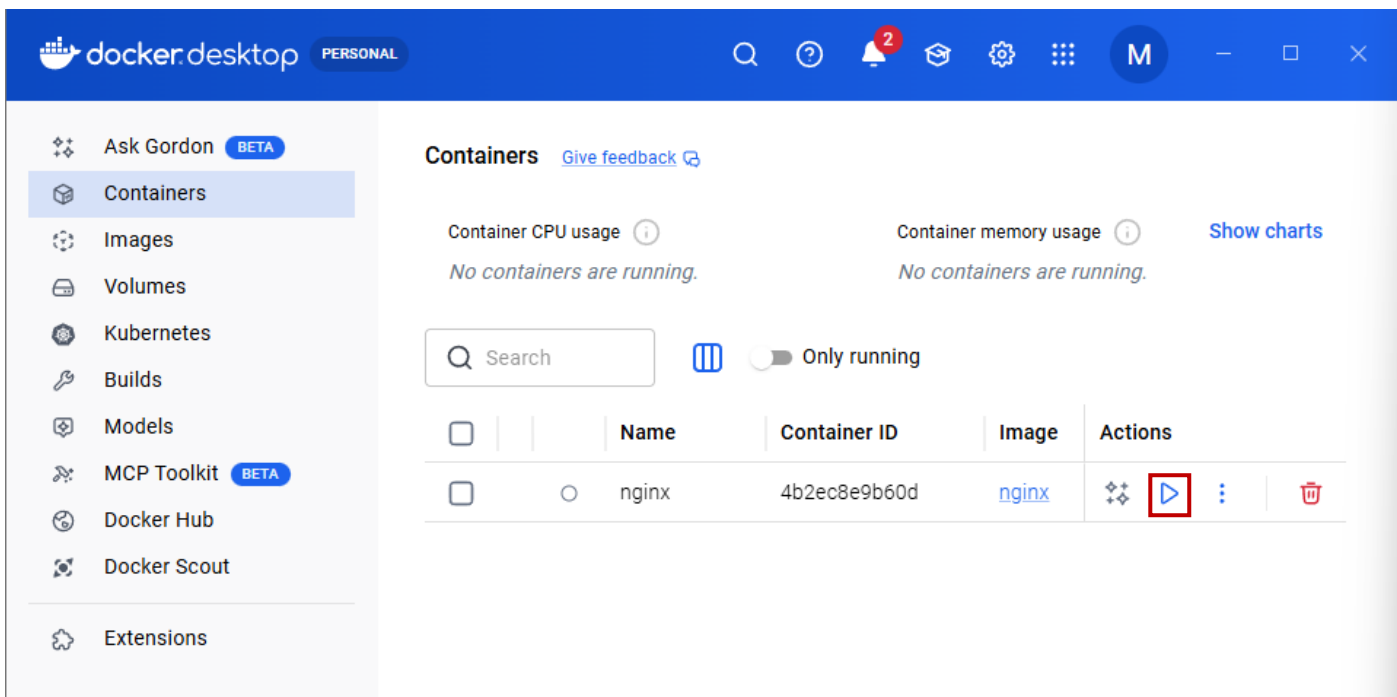
## 5. Deuxième test.

- Je lance un conteneur nginx avec son image :

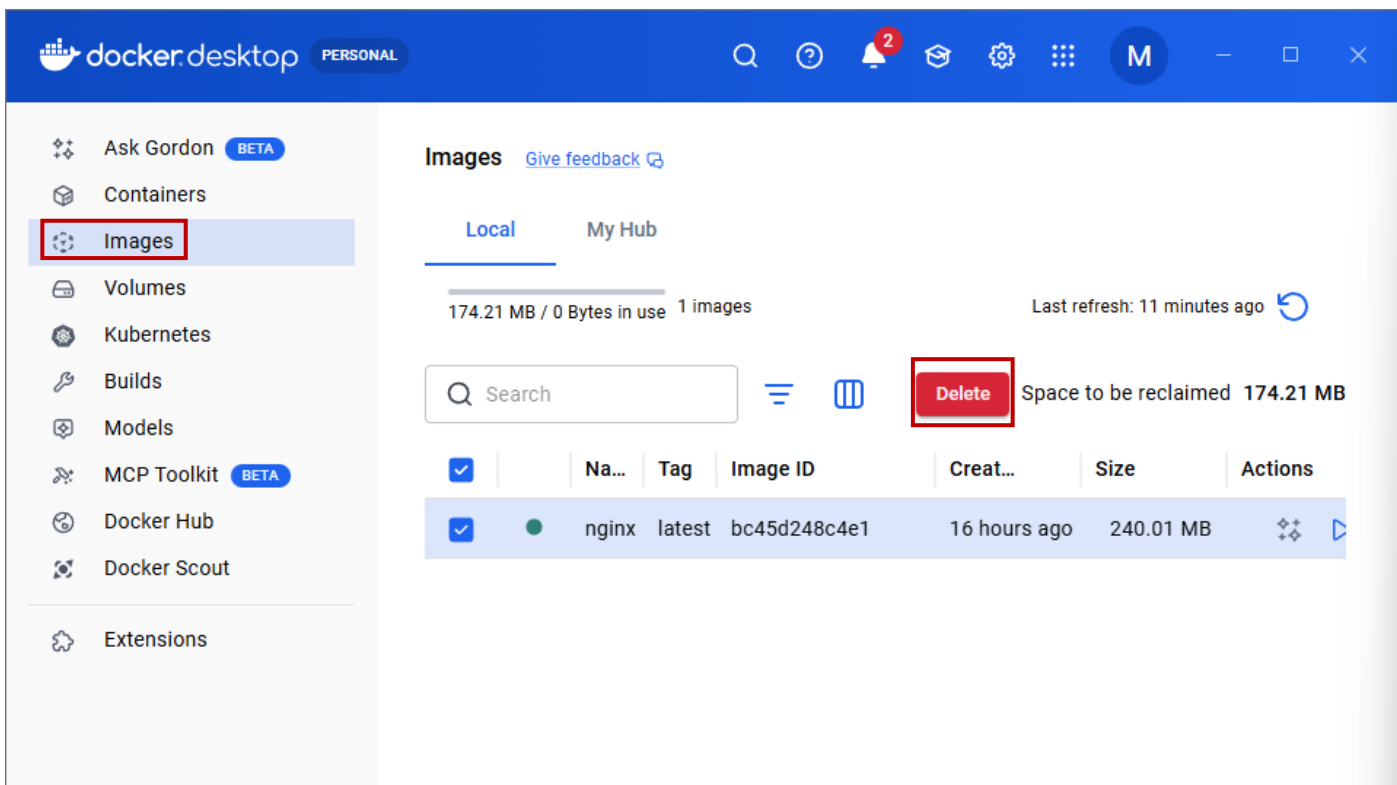
```
sio@G102-GB16: ~  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
sio@G102-GB16:~$ docker run --name nginx -p 8080:80 nginx  
Unable to find image 'nginx:latest' locally  
latest: Pulling from library/nginx  
9eef040df109: Pull complete  
206356c42440: Pull complete  
75ald70aee50: Pull complete  
a9d395129dce: Pull complete  
df9da45c1db2: Pull complete  
79697674b897: Pull complete  
18a071c04bd1: Pull complete  
23abb0f9ce55: Download complete  
d99947bc9177: Download complete  
Digest: sha256:bc45d248c4e1d1709321de61566eb2b64d4f0e32765239d66573666be7f13349  
Status: Downloaded newer image for nginx:latest  
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration  
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/  
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh  
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf  
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf  
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh  
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh  
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh  
/docker-entrypoint.sh: Configuration complete; ready for start up  
2026/03/11 14:01:44 [notice] 1#1: using the "epoll" event method  
2026/03/11 14:01:44 [notice] 1#1: nginx/1.29.6  
2026/03/11 14:01:44 [notice] 1#1: built by gcc 14.2.0 (Debian 14.2.0-19)  
2026/03/11 14:01:44 [notice] 1#1: OS: Linux 6.6.87.2-microsoft-standard-WSL2  
2026/03/11 14:01:44 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576  
2026/03/11 14:01:44 [notice] 1#1: start worker processes  
2026/03/11 14:01:44 [notice] 1#1: start worker process 29  
2026/03/11 14:01:44 [notice] 1#1: start worker process 30  
2026/03/11 14:01:44 [notice] 1#1: start worker process 31  
2026/03/11 14:01:44 [notice] 1#1: start worker process 32  
2026/03/11 14:01:44 [notice] 1#1: start worker process 33  
2026/03/11 14:01:44 [notice] 1#1: start worker process 34  
2026/03/11 14:01:44 [notice] 1#1: start worker process 35  
2026/03/11 14:01:44 [notice] 1#1: start worker process 36  
2026/03/11 14:01:44 [notice] 1#1: start worker process 37  
2026/03/11 14:01:44 [notice] 1#1: start worker process 38  
2026/03/11 14:01:44 [notice] 1#1: start worker process 39  
2026/03/11 14:01:44 [notice] 1#1: start worker process 40  
2026/03/11 14:01:44 [notice] 1#1: start worker process 41  
2026/03/11 14:01:44 [notice] 1#1: start worker process 42  
2026/03/11 14:01:44 [notice] 1#1: start worker process 43  
2026/03/11 14:01:44 [notice] 1#1: start worker process 44
```

Remarque : Docker télécharge dans un premier temps l'image nginx. Il crée ensuite une instance de cette image qui est le conteneur.



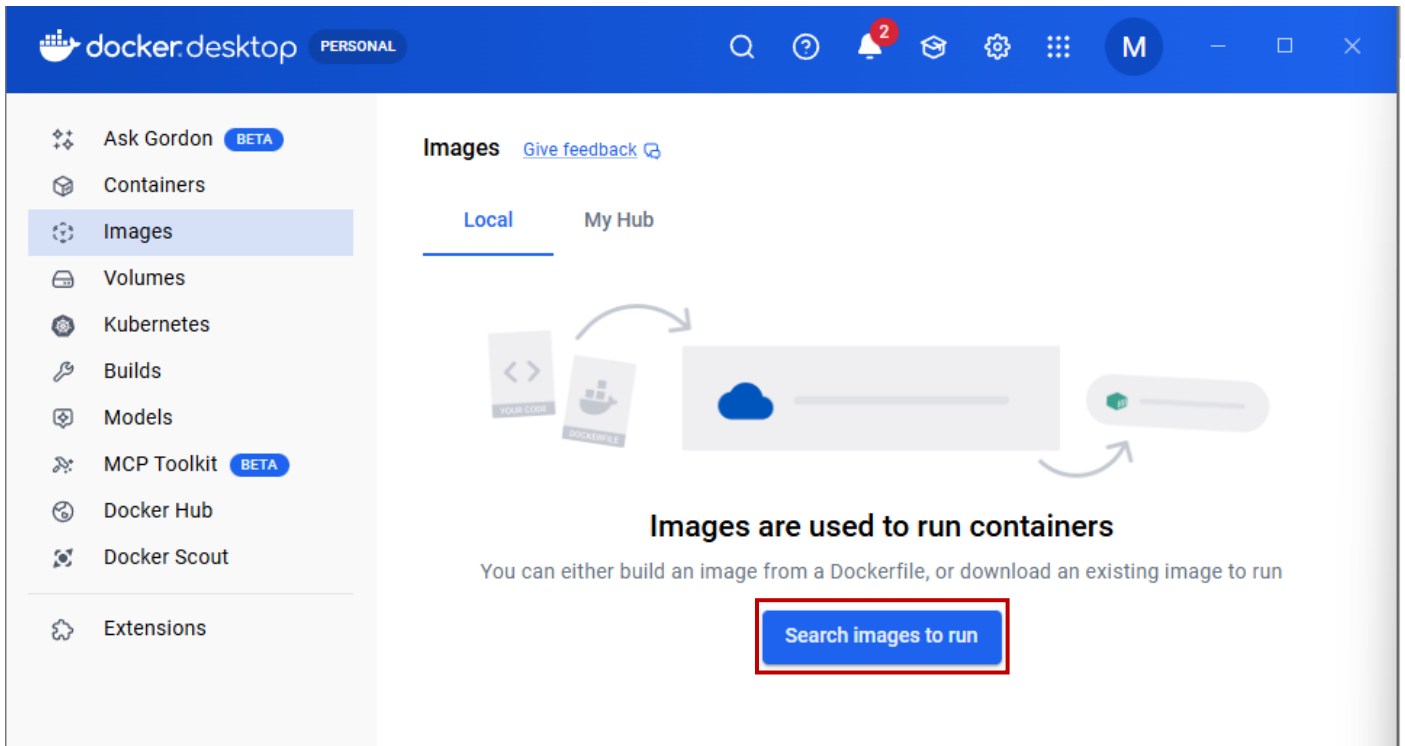


➤ Après avoir éteint le container, je me rends dans l'onglet Images et je supprime l'image :

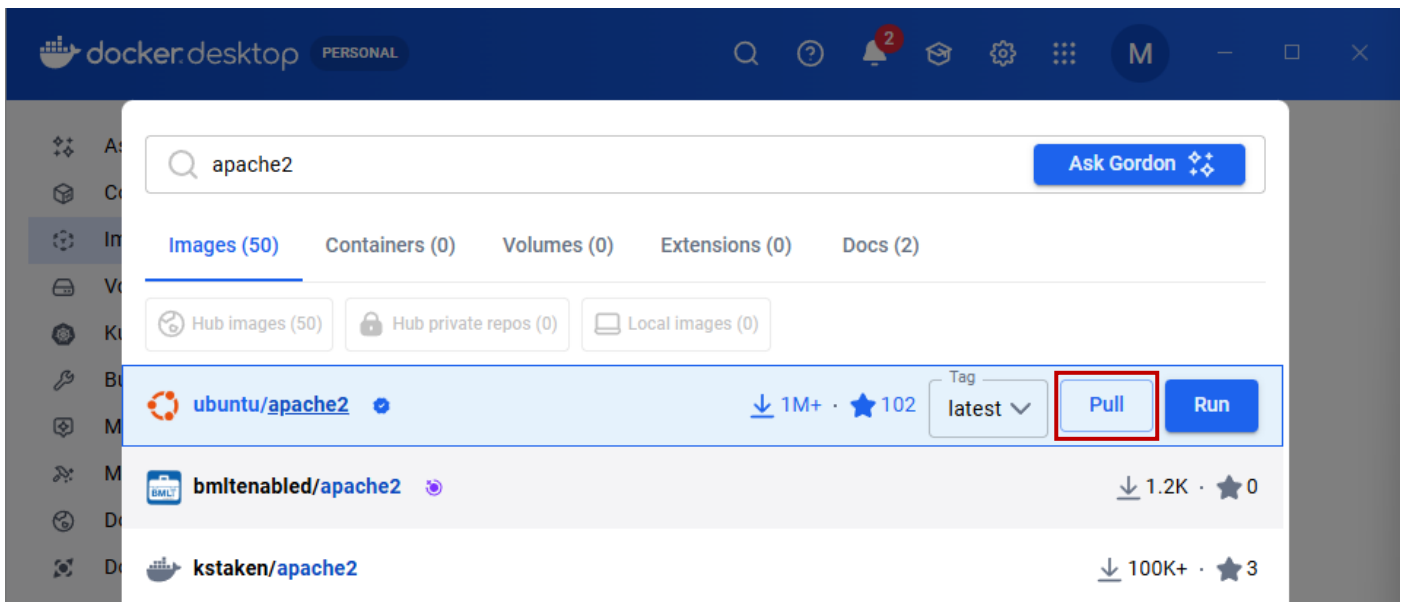


## 6. Troisième test.

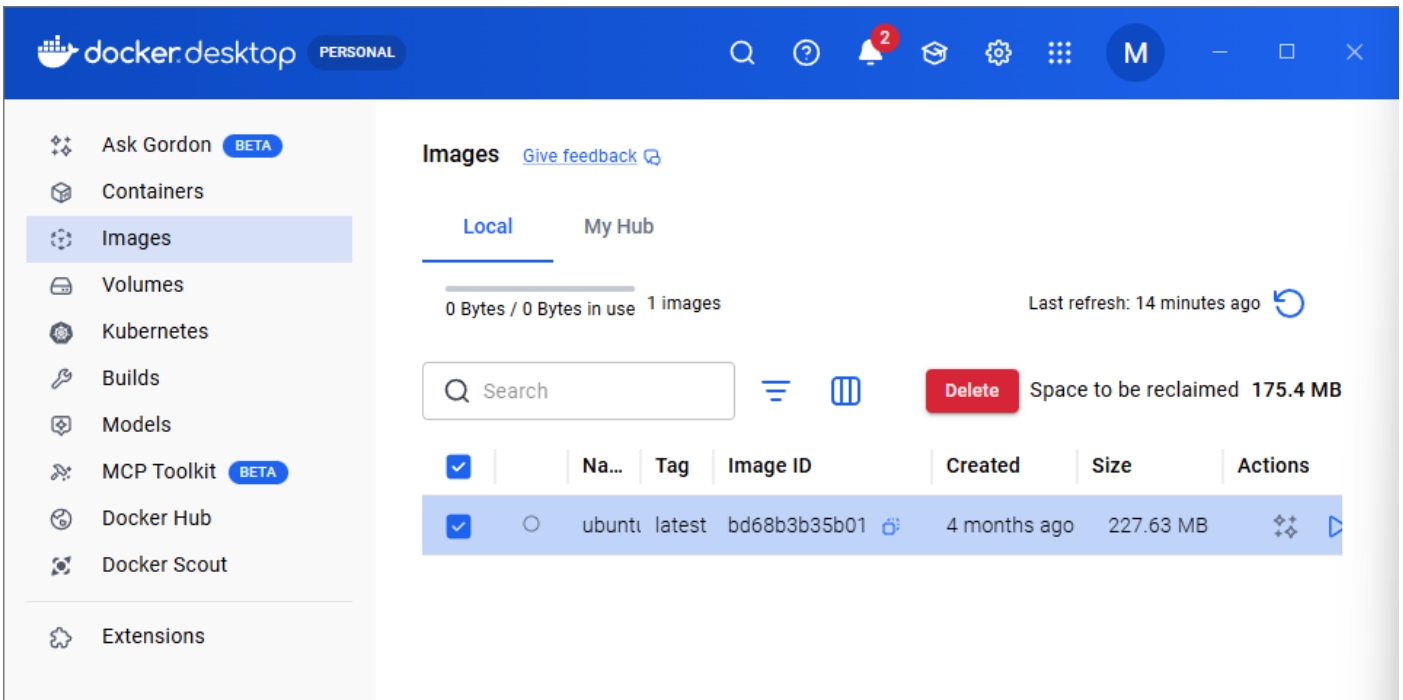
- Dans l'onglet Images je peux faire une recherche d'image en local :



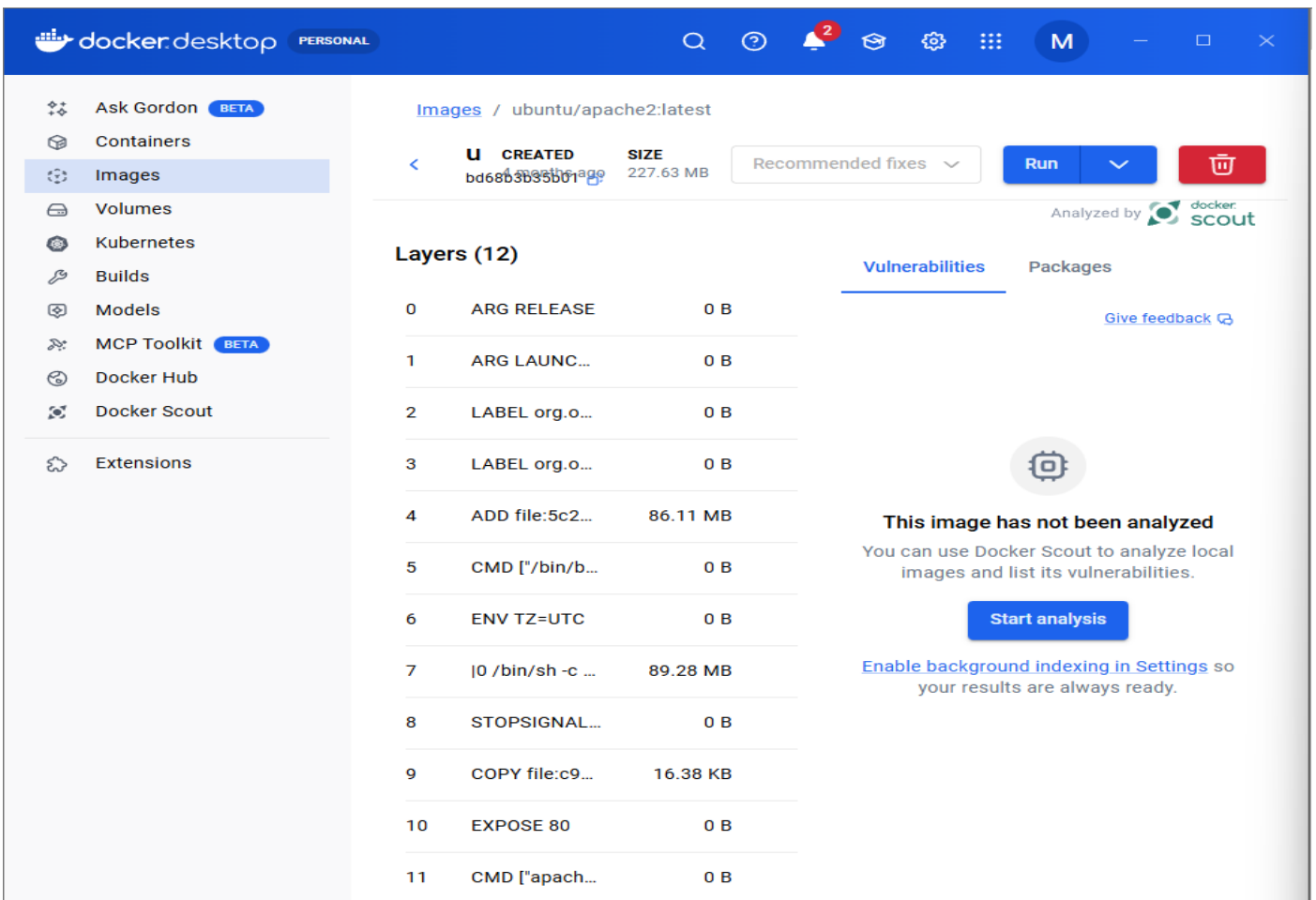
- Je tape apache2 dans la barre de recherche, puis sur pull :



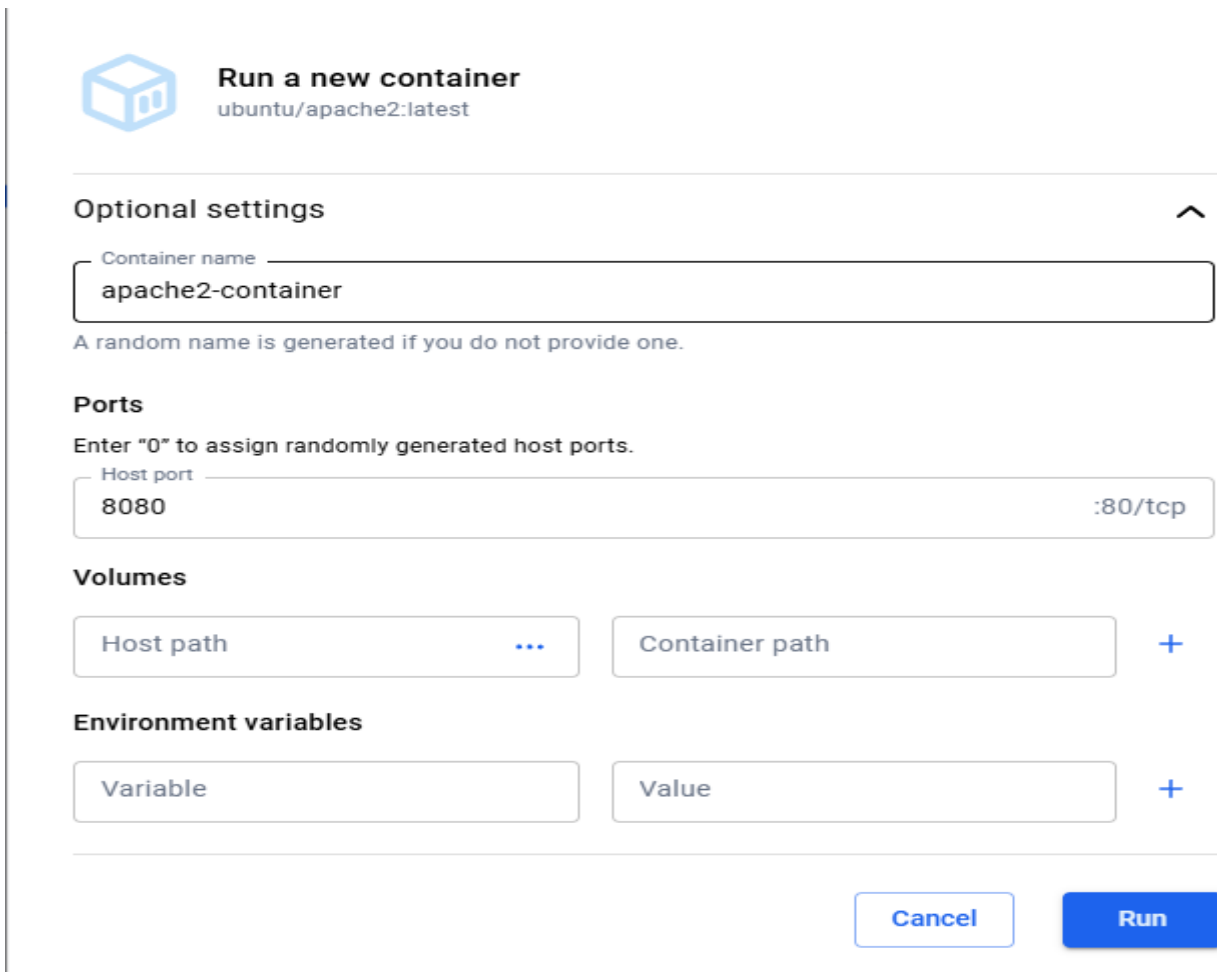
➤ Elle est maintenant présente dans l'onglet Images :



➤ J'appuis sur run :



➤ Je renseigne son nom et son port :



**Run a new container**  
ubuntu/apache2:latest

**Optional settings**

Container name

A random name is generated if you do not provide one.

**Ports**  
Enter "0" to assign randomly generated host ports.  
Host port  
 :80/tcp

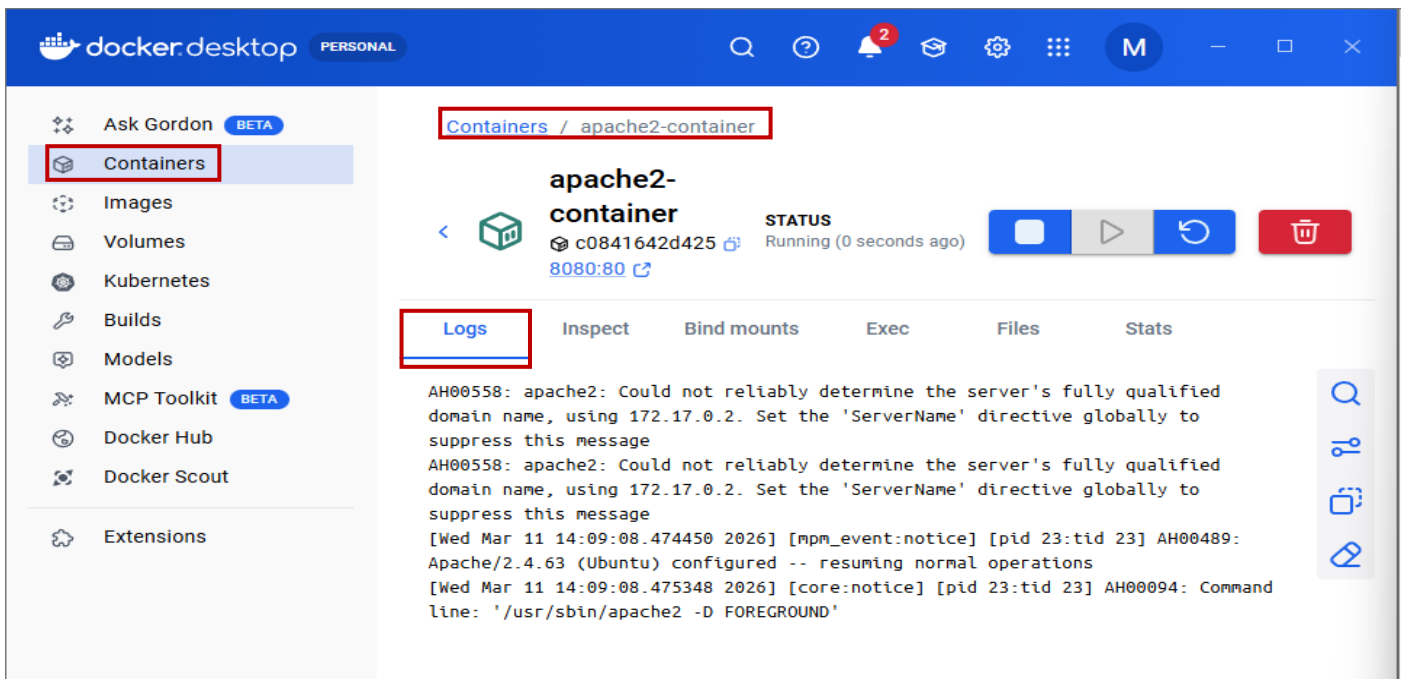
**Volumes**

Host path  ... Container path  +

**Environment variables**

Variable  Value  +

➤ Je me rends dans les logs de ce conteneur :



docker.desktop PERSONAL

Containers / apache2-container

**apache2-container**  
c0841642d425 Running (0 seconds ago)  
8080:80

**Logs** Inspect Bind mounts Exec Files Stats

```
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress this message
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress this message
[Wed Mar 11 14:09:08.474450 2026] [mpm_event:notice] [pid 23:tid 23] AH00489: Apache/2.4.63 (Ubuntu) configured -- resuming normal operations
[Wed Mar 11 14:09:08.475348 2026] [core:notice] [pid 23:tid 23] AH00094: Command line: '/usr/sbin/apache2 -D FOREGROUND'
```

➤ Je me rends dans l'onglet Exec :

The screenshot shows the Docker Desktop interface. On the left is a sidebar with navigation options: Ask Gordon (BETA), Containers (selected), Images, Volumes, Kubernetes, Builds, Models, MCP Toolkit (BETA), Docker Hub, Docker Scout, and Extensions. The main area displays the 'apache2-container' details, including its ID 'c0841642d425' and status 'Running (19 seconds ago)'. Below this are tabs for Logs, Inspect, Bind mounts, Exec (highlighted with a red box), Files, and Debug mode. The 'Exec' tab shows a terminal session with the following commands and output:

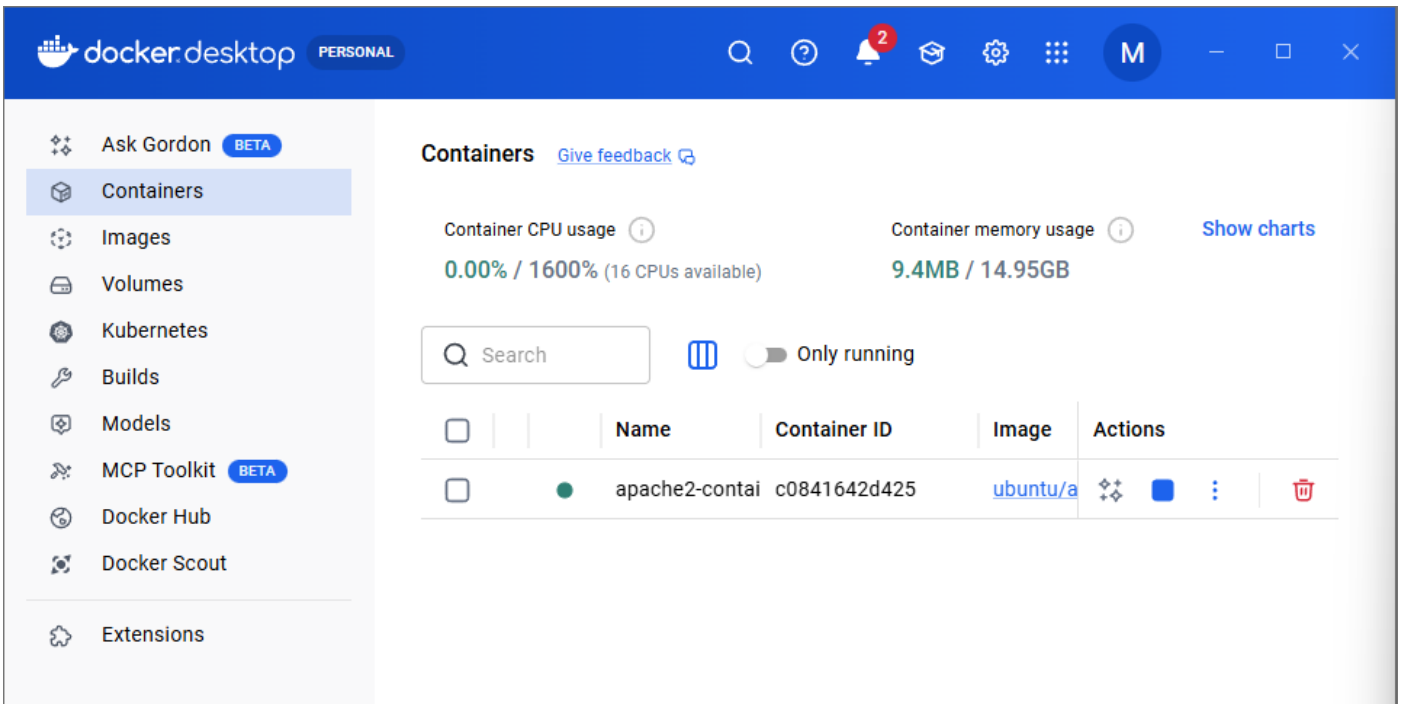
```
# pwd
/  
# cd /etc/apache2  
# ls  
apache2.conf  conf-enabled  magic          mods-enabled  sites-available  
conf-available  envvars      mods-available  ports.conf    sites-enabled  
# cd /var/www/html  
# ls  
index.html  
#
```

➤ Puis dans les Statistiques :

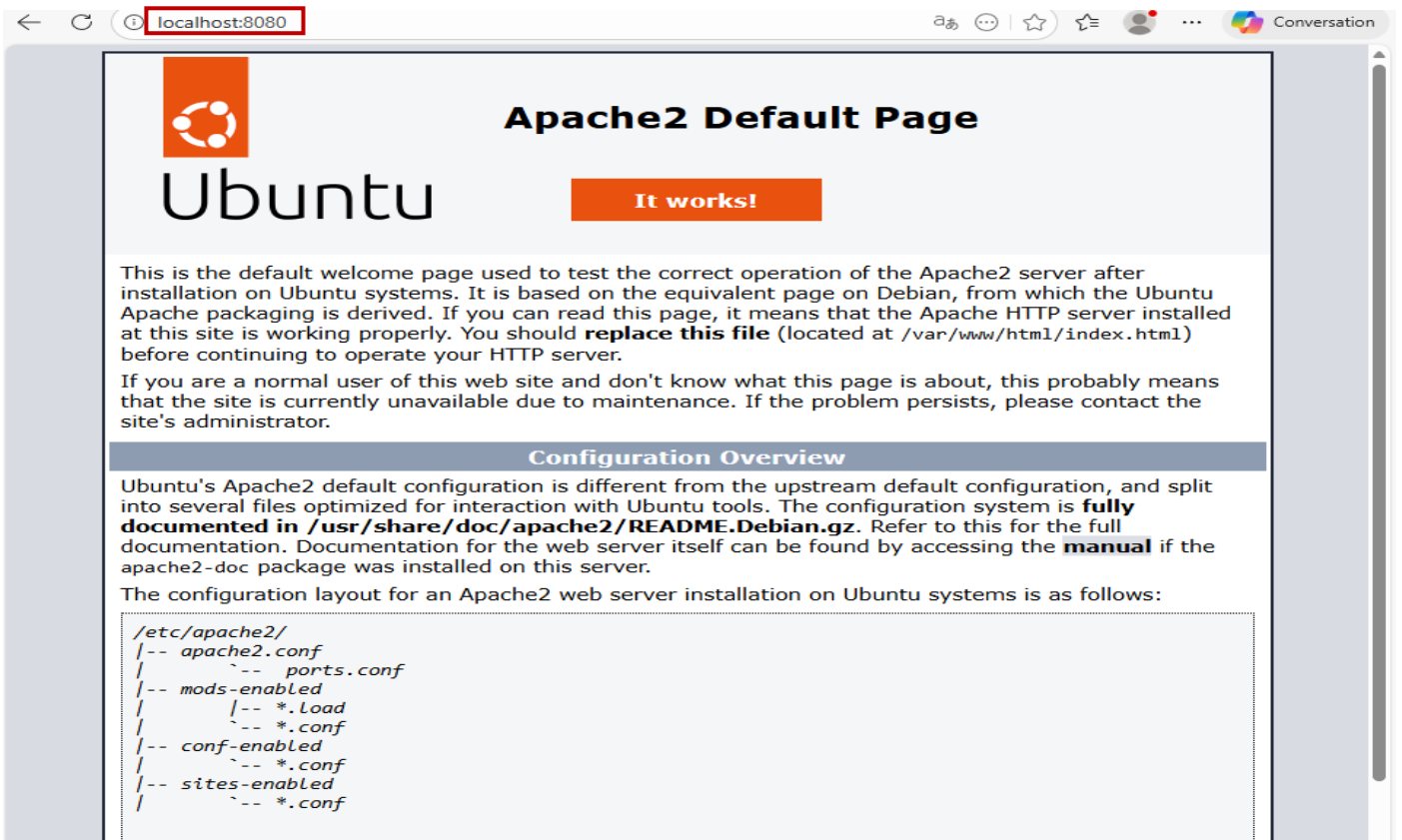
The screenshot shows the Docker Desktop interface with the 'Stats' tab selected for the 'apache2-container'. The container is shown as 'Running (1 minute ago)'. The 'Stats' tab displays two graphs:

- CPU usage:** 0.01%. The graph shows a sharp increase in CPU usage starting around the 00:00:00 mark, reaching 0.01% by 00:00:10.
- Memory usage:** 9.46MB / 15.31GB. The graph shows memory usage starting at 0B and increasing to 9.46MB by 00:00:10.

➤ Je me rends dans l'onglet Containers, le conteneur est en ligne :



➤ Sur mon navigateur je vais en localhost sur le port attribué que j'ai attribué tout à l'heure :



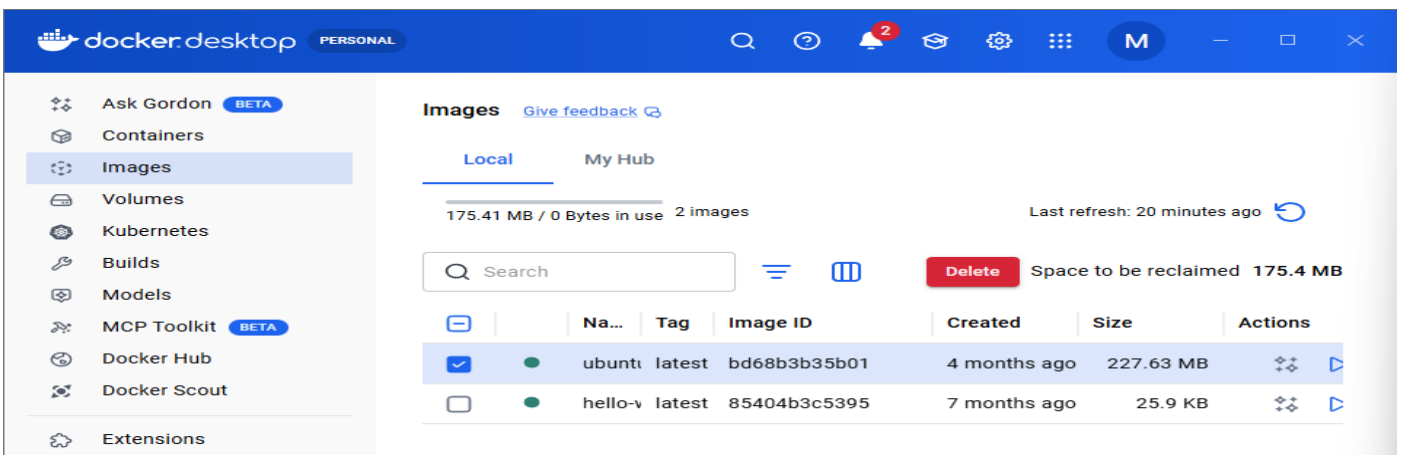
J'arrête et je supprime le conteneur puis l'image.

## 7. Quatrième test.

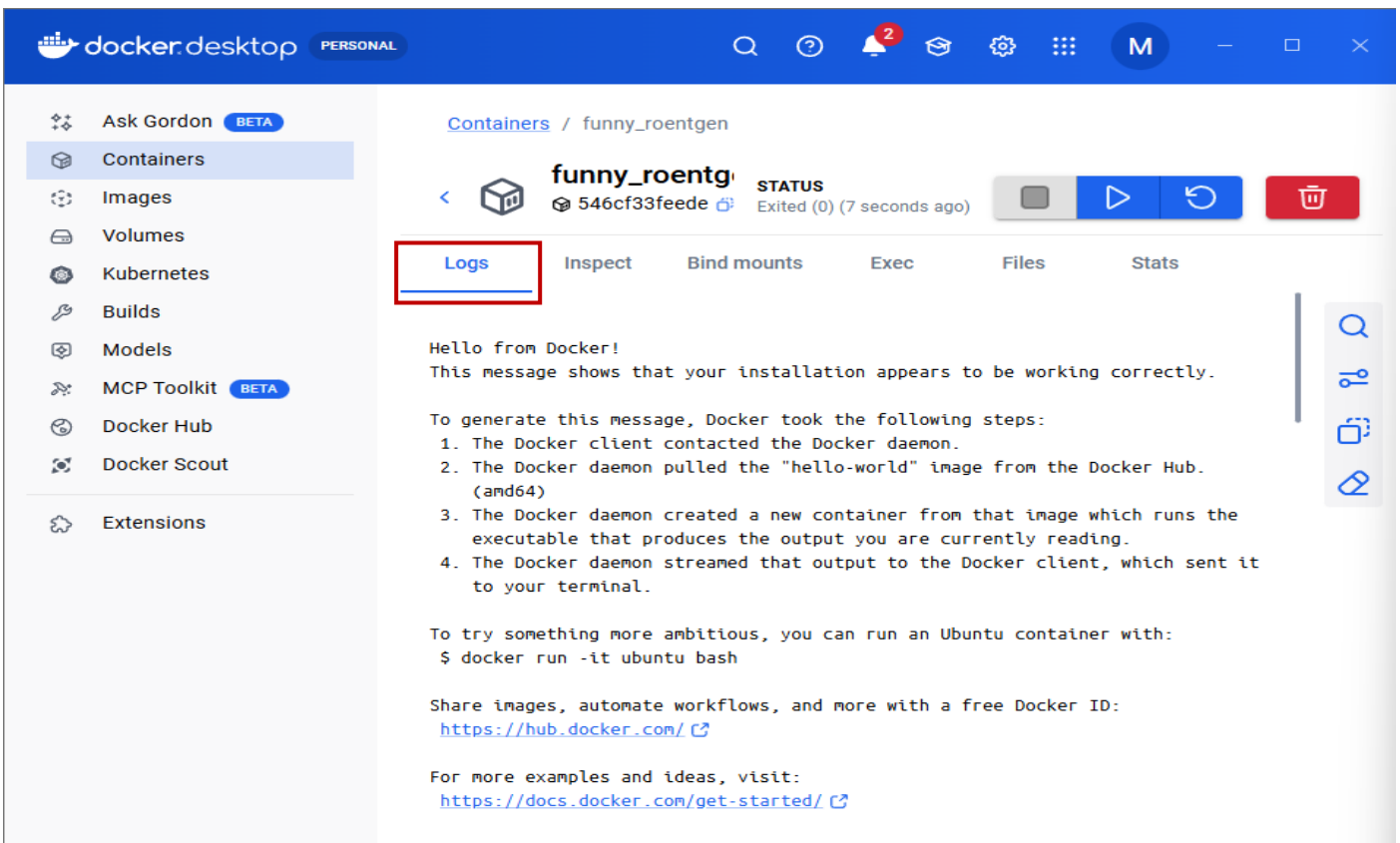
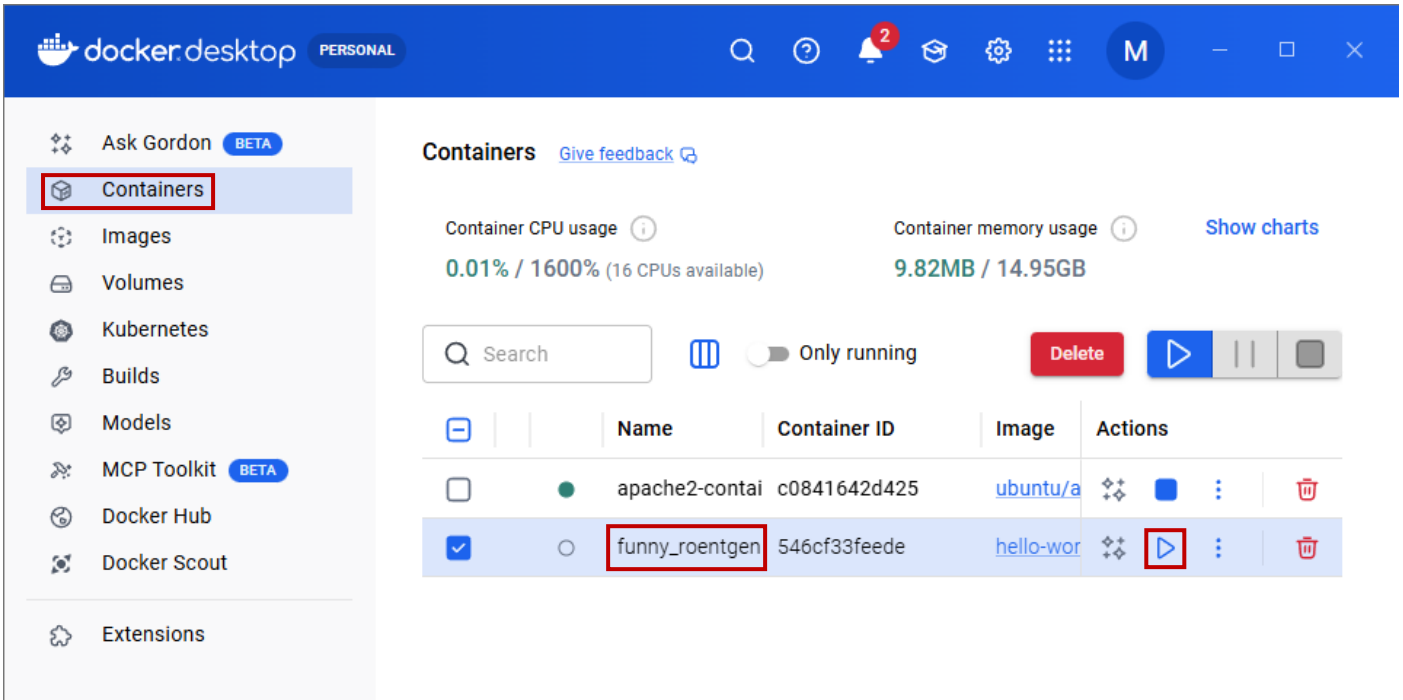
- Dans le terminal Ubuntu je tape la commande docker run hello-world :

```
sio@G102-GB16: ~  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
sio@G102-GB16:~$ docker run hello-world  
Unable to find image 'hello-world:latest' locally  
latest: Pulling from library/hello-world  
17eec7bbc9d7: Pull complete  
ea52d2000f90: Download complete  
Digest: sha256:85404b3c53951c3ff5d40de0972b1bb21fafa2e8daa235355baf44f33db9dbdd  
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/  
  
sio@G102-GB16:~$
```

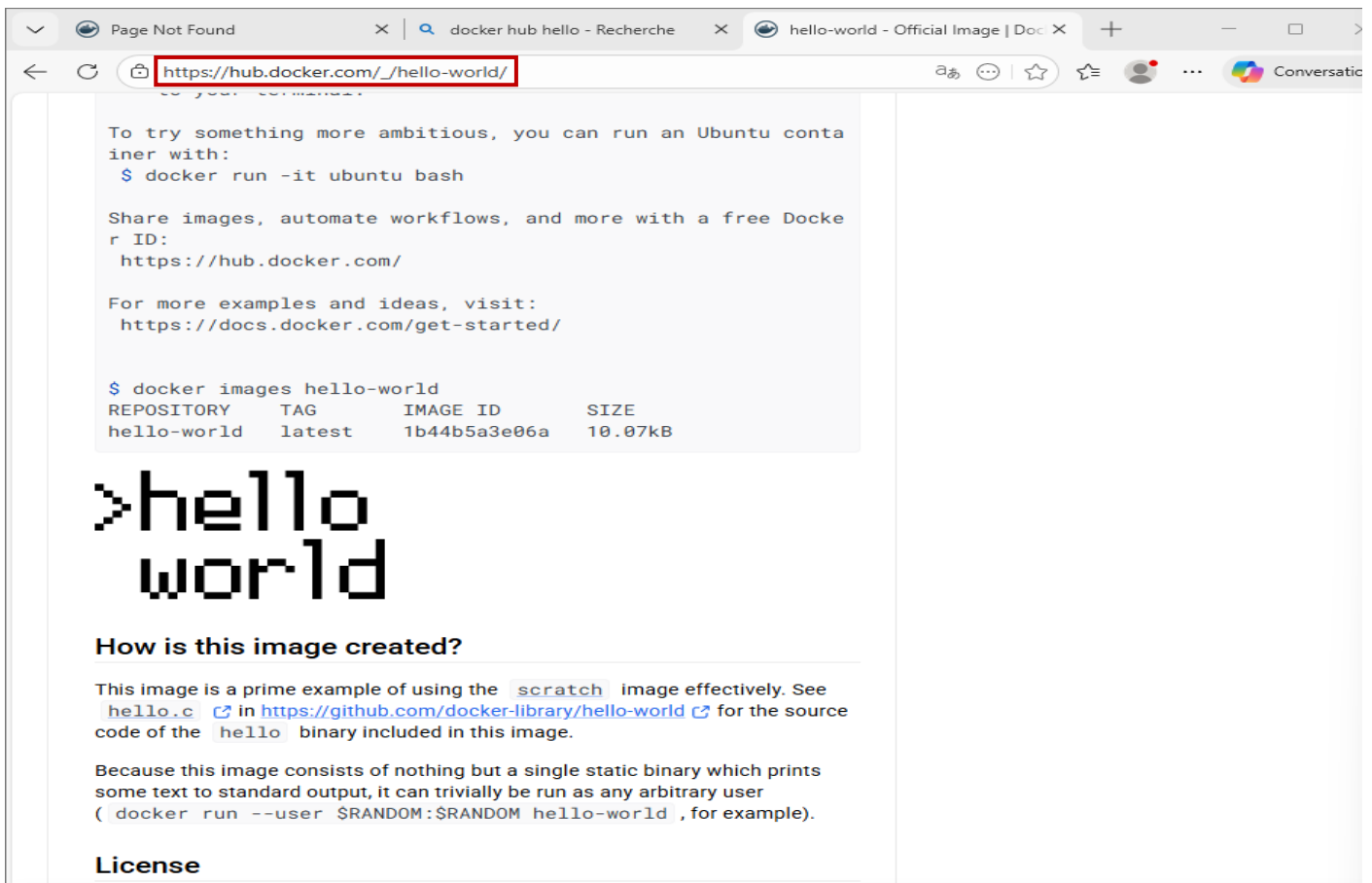
- Dans Docker je me rends dans l'onglet Images :



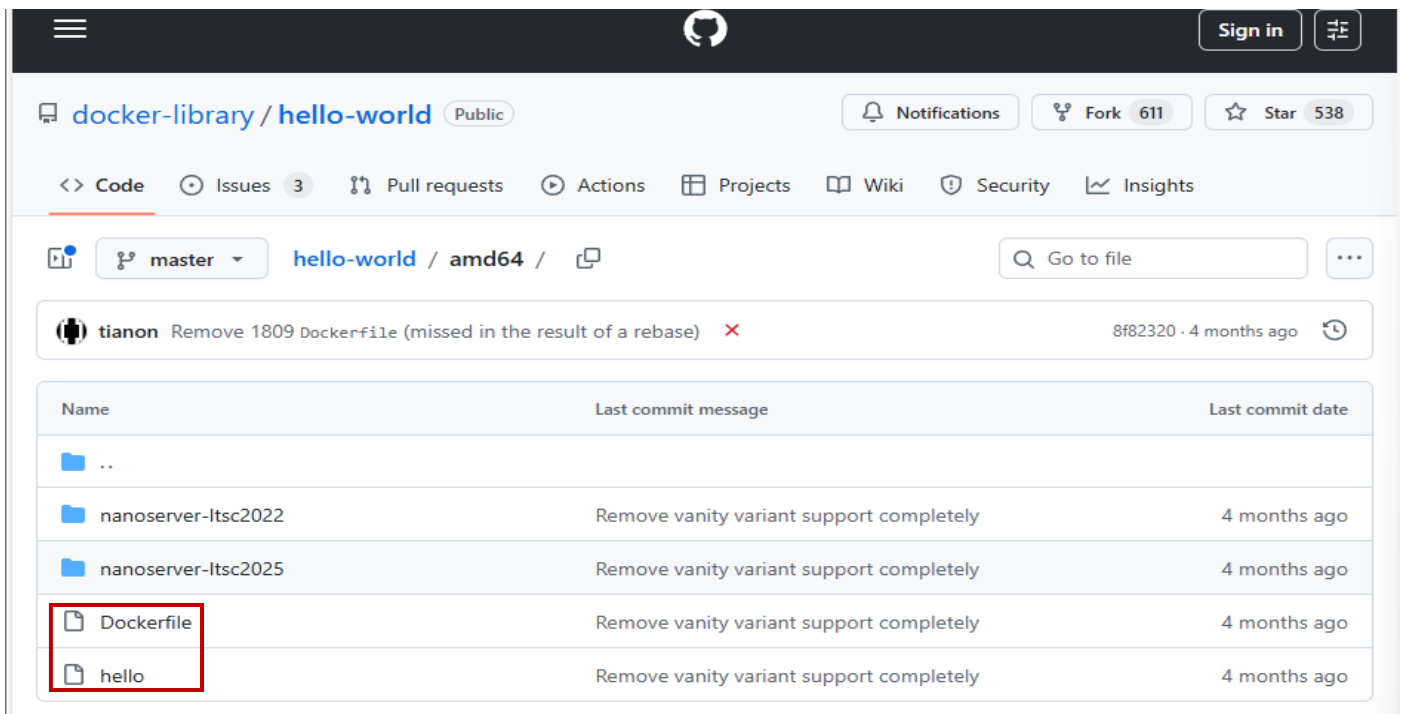
- Dans l'onglet Containers je constate la présence d'un nouveau container, je me rend dans les logs :



➤ Je me rends sur le hub de docker pour voir d'où vient le hello-world :



➤ Voici les fichiers qui servent à créer l'image hello-world :



## 8. Cinquième test : mode interactif d'utilisation d'un conteneur.

- Commande docker images : liste toutes les images sur le cache local de la machine :

```
INFO In Use
IMAGE          ID          DISK USAGE  CONTENT SIZE  EXTRA
hello-world:latest  85404b3c5395  25.9kB     9.52kB       U
ubuntu/apache2:latest  bd68b3b35b01  228MB     52.2MB       U
sio@G102-GB16:~$
```

- Je lance un conteneur en mode interactif qui lancera un shell bash qui interprètera les commandes :

```
sio@G102-GB16:~$ docker run -it ubuntu bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
01d7766a2e4a: Pull complete
fd8cda969ed2: Download complete
Digest: sha256:d1e2e92c075e5ca139d51a140fff46f84315c0fdce203eab2807c7e495eff4f9
Status: Downloaded newer image for ubuntu:latest
root@567d6e8a4056:/#
```

Option -i : permet d'interagir avec le conteneur.

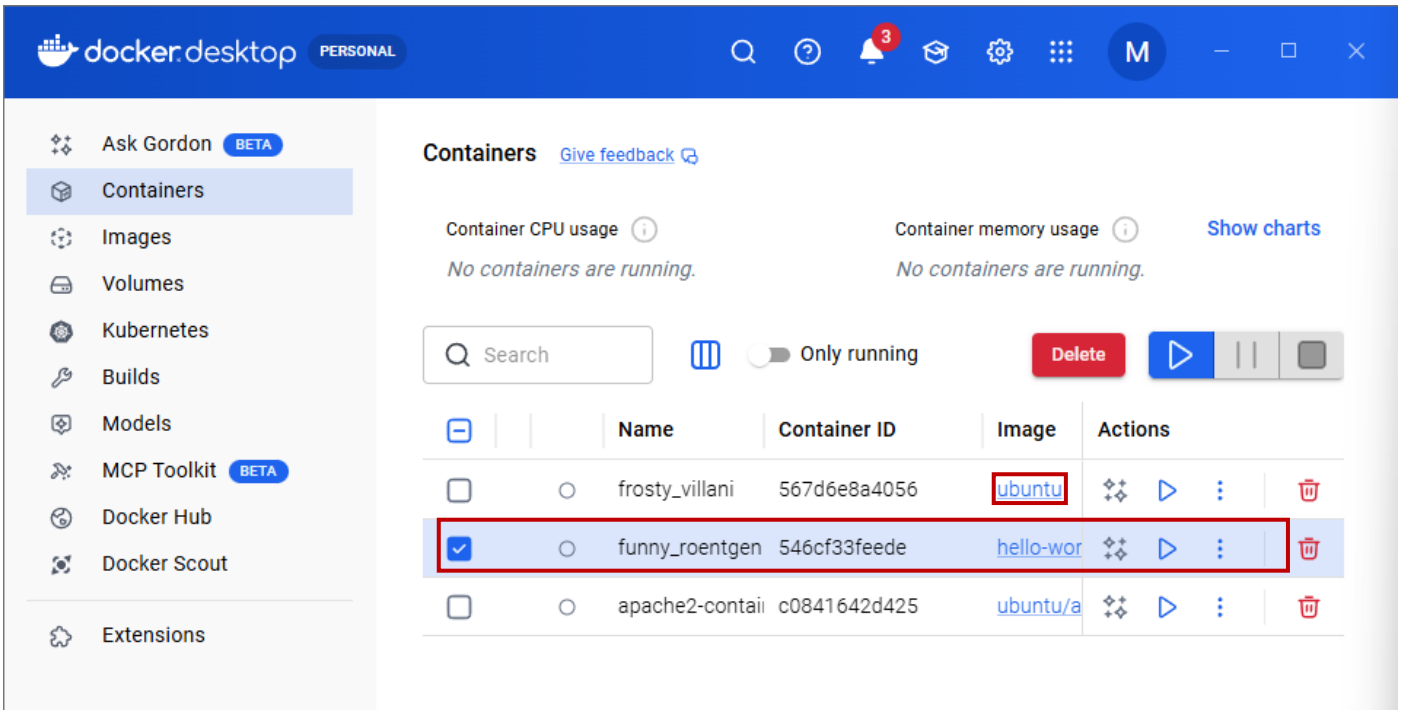
Option -t : permet de saisir du texte (des commandes) dans une console

- Dans docker je me rends dans l'onglet Images, Le conteneur Ubuntu est en marche (sur mon screen il est en cours d'installation) :

The screenshot shows the Docker Desktop interface. On the left is a sidebar with navigation options: Ask Gordon (BETA), Containers, Images (selected), Volumes, Kubernetes, Builds, Models, MCP Toolkit (BETA), Docker Hub, Docker Scout, and Extensions. The main area is titled 'Images' and has tabs for 'Local' and 'My Hub'. A progress bar at the top indicates '263.01 MB / 347.08 MB in use' and '3 images'. Below this is a search bar and a refresh button. A table lists the local images:

<input type="checkbox"/>	Na...	Tag	Image ID	Created	Size	Actions
<input type="checkbox"/>	●	ubuntu latest	d1e2e92c075e	1 month ago	119.26 MB	⚙️ ▶️
<input type="checkbox"/>	●	ubuntu latest	bd68b3b35b01	4 months ago	227.63 MB	⚙️ ▶️
<input type="checkbox"/>	●	hello-v latest	85404b3c5395	8 months ago	25.9 KB	⚙️ ▶️

➤ J'éteins celui d'avant :



➤ Je tape la commande docker ps (depuis la console de la machine Ubuntu ou depuis le terminal Windows) permet de lister les conteneurs en cours d'exécution :

```
Installez la dernière version de PowerShell pour de nouvelles fonctionnalités et améliorations ! https://aka.ms/PSWindows
PS C:\WINDOWS\system32> docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
PS C:\WINDOWS\system32>
```

➤ Je tape quelques commandes :

```
sio@G102-GB16:~$ top
top - 15:20:27 up 4 min, 2 users, load average: 0.00, 0.03, 0.00
Tasks: 31 total, 1 running, 30 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 99.9 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 15676.2 total, 13277.6 free, 1076.8 used, 1532.2 buff/cache
MiB Swap: 4096.0 total, 4096.0 free, 0.0 used, 14599.4 avail Mem

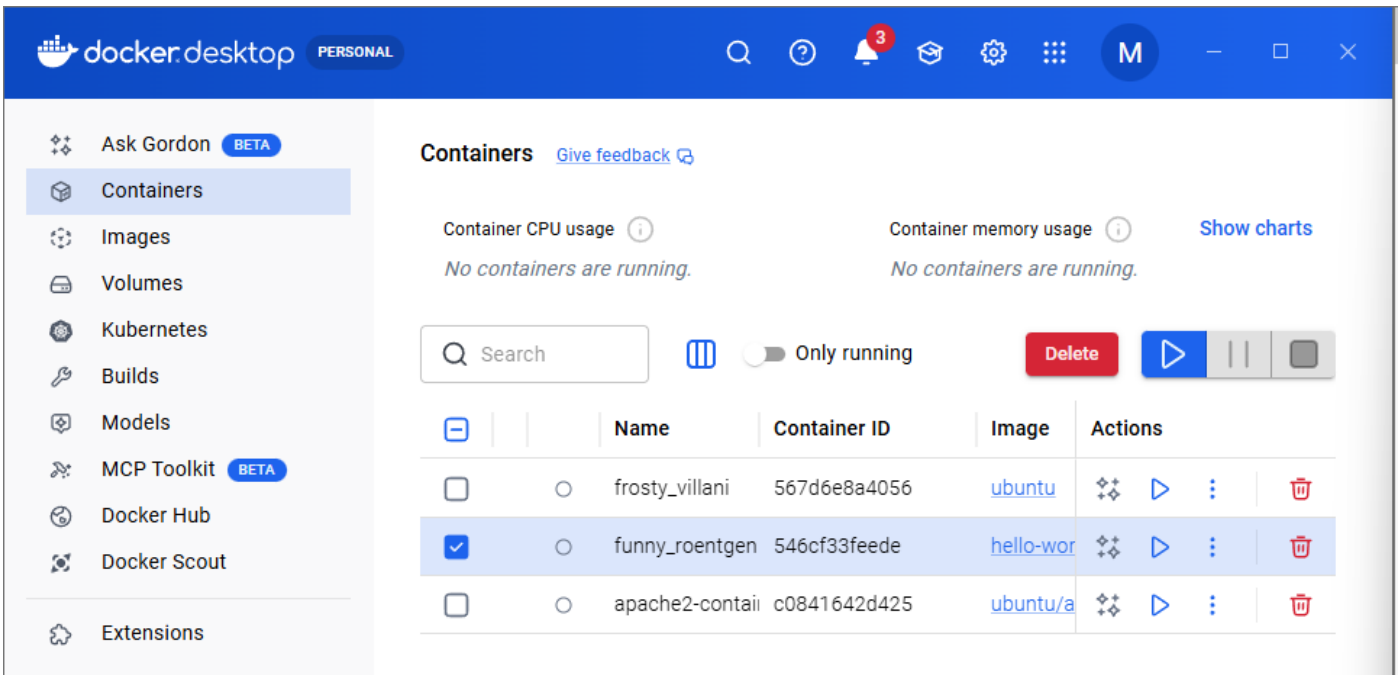
  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
    1 root        0   0  21808 11712 8896  S   0.0   0.1   0:00.62  systemd
    2 root        0   0   3120  2048  2048  S   0.0   0.0   0:00.00  init-systemd(Ub
    7 root        0   0   3136  1792  1792  S   0.0   0.0   0:00.00  init
   56 root       19  -1  42228 15488 14592  S   0.0   0.1   0:00.16  systemd-journal
  101 root        0   0  25120  5760  4736  S   0.0   0.0   0:00.22  systemd-udevd
  184 systemd+   20   0  21452 12544 10368  S   0.0   0.1   0:00.07  systemd-resolve
  185 systemd+   20   0  91020  7424  6656  S   0.0   0.0   0:00.06  systemd-timesyn
  191 root        0   0   4236  2432  2304  S   0.0   0.0   0:00.00  cron
```

- Je créer un fichier, et je vérifie sa présence :

```
sio@G102-GB16:~$ touch fichier_test_persistence
sio@G102-GB16:~$ ls
fichier_test_persistence
```

- Je supprime /home, puis je vérifie sa suppression puis je stoppe le conteneur :

```
sio@G102-GB16:~$ ls
sio@G102-GB16:~$ touch fichier_test
sio@G102-GB16:~$ ls
fichier_test
sio@G102-GB16:~$ rm fichier_test
sio@G102-GB16:~$ ls
sio@G102-GB16:~$
```



- Je vérifie que le conteneur n'est plus en marche :

```
PS C:\WINDOWS\system32> docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
PS C:\WINDOWS\system32> docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
567d6e8a4056   ubuntu   "bash"    2 weeks ago   Exited (129) 2 weeks ago   frosty_villani
546cf33feede   hello-world   "/hello"  2 weeks ago   Exited (0) 2 weeks ago   funny_roentgen
c0841642d425   ubuntu/apache2:latest   "apache2-foreground"  2 weeks ago   Exited (255) 8 minutes ago   0.0.0.0:8080->80/tcp   apache2-container
PS C:\WINDOWS\system32>
```

- Je redémarre le conteneur et j'y accède avec l'option -ai :

```
sio@G102-GB16:~$ docker start 567d
567d
sio@G102-GB16:~$ docker stop 567d
567d
sio@G102-GB16:~$ docker start -ai 567d
root@567d6e8a4056:/# ls
bin  dev  home  lib64  mnt  proc  run  srv  tmp  var
boot  etc  lib  media  opt  root  sbin  sys  usr
root@567d6e8a4056:/# exit
exit
sio@G102-GB16:~$
```

- Voici une autre manière d'accéder au conteneur :

```
sio@G102-GB16:~$ docker start 567d
567d
sio@G102-GB16:~$ docker exec -it 567d bash
root@567d6e8a4056:/# exit
exit
sio@G102-GB16:~$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
567d6e8a4056   ubuntu   "bash"    2 weeks ago   Up 15 seconds   frosty_villani
sio@G102-GB16:~$ docker stop 567d
567d
sio@G102-GB16:~$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
sio@G102-GB16:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
567d6e8a4056   ubuntu   "bash"    2 weeks ago   Exited (137) 9 seconds ago
546cf33feede   hello-world   "/hello"   2 weeks ago   Exited (0) 2 weeks ago
c0841642d425   ubuntu/apache2:latest   "apache2-foreground"   2 weeks ago   Exited (255) 10 minutes ago
o  0.0.0.0:8080->80/tcp   apache2-container
```

- Je supprime en ligne de commande les 2 containers :

```
sio@G102-GB16:~$ docker rm frosty_villani
frosty_villani
sio@G102-GB16:~$ docker rm funny_roentgen
funny_roentgen
sio@G102-GB16:~$ docker rm apache2-container
apache2-container
sio@G102-GB16:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
sio@G102-GB16:~$
```

```
sio@G102-GB16:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS          NAMES
960ebab22f7e  ubuntu   "/bin/bash"            3 minutes ago   Exited (0) 2 minutes ago          test_modif
ication
sio@G102-GB16:~$ docker rm test_modification
test_modification
sio@G102-GB16:~$
```

- Je lance un nouveau conteneur à partir de l'image créée avec la commande docker commit et je vérifie que les modifications sont bien présentes :

```
sio@G102-GB16:~$ docker run -it --rm --name test ubuntu:1.0
root@4f191277853f:/# ls /home
ubuntu
root@4f191277853f:/# exit
exit
sio@G102-GB16:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED          STATUS          PORTS          NAMES
sio@G102-GB16:~$
```

## 10. Première approche des volumes.

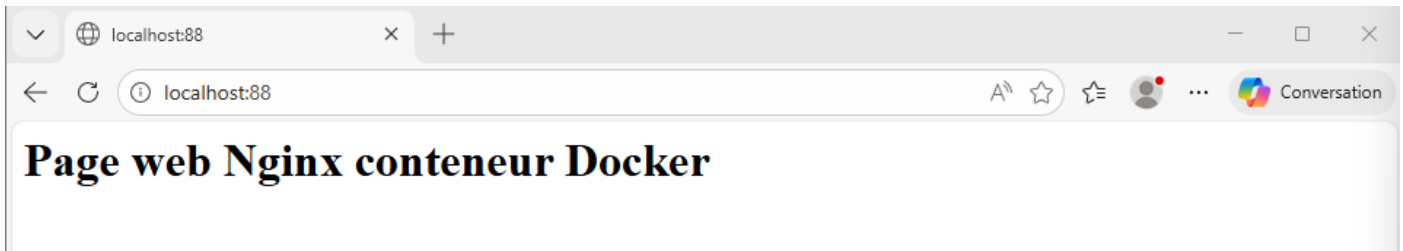
- Les données sont sur le système local : je crée un répertoire web dans /home/sio de la machine ubuntu ainsi qu'une page html :

```
sio@G102-GB16:~$ pwd
/home/sio
sio@G102-GB16:~$ mkdir web
sio@G102-GB16:~$ cd web
sio@G102-GB16:~/web$ nano index.html
sio@G102-GB16:~/web$
```

```
GNU nano 7.2                                index.html
<html><
<body>
<h1>Page web Nginx conteneur Docker</h1>
</body>
</html>
```

- Je crée un conteneur qui va pointer sur le répertoire web du FS local accessible en lecture ( je n'ai pas le screen de la création) :

- Je teste l'accès aux données :



- J'affiche les logs du conteneur web :

```
sio@G102-GB16:~$ docker logs web
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2026/03/25 14:37:15 [notice] 1#1: using the "epoll" event method
2026/03/25 14:37:15 [notice] 1#1: nginx/1.29.7
2026/03/25 14:37:15 [notice] 1#1: built by gcc 14.2.0 (Debian 14.2.0-19)
2026/03/25 14:37:15 [notice] 1#1: OS: Linux 6.6.87.2-microsoft-standard-WSL2
2026/03/25 14:37:15 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2026/03/25 14:37:15 [notice] 1#1: start worker processes
2026/03/25 14:37:15 [notice] 1#1: start worker process 29
2026/03/25 14:37:15 [notice] 1#1: start worker process 30
2026/03/25 14:37:15 [notice] 1#1: start worker process 31
2026/03/25 14:37:15 [notice] 1#1: start worker process 32
2026/03/25 14:37:15 [notice] 1#1: start worker process 33
2026/03/25 14:37:15 [notice] 1#1: start worker process 34
2026/03/25 14:37:15 [notice] 1#1: start worker process 35
2026/03/25 14:37:15 [notice] 1#1: start worker process 36
2026/03/25 14:37:15 [notice] 1#1: start worker process 37
2026/03/25 14:37:15 [notice] 1#1: start worker process 38
2026/03/25 14:37:15 [notice] 1#1: start worker process 39
2026/03/25 14:37:15 [notice] 1#1: start worker process 40
2026/03/25 14:37:15 [notice] 1#1: start worker process 41
2026/03/25 14:37:15 [notice] 1#1: start worker process 42
2026/03/25 14:37:15 [notice] 1#1: start worker process 43
2026/03/25 14:37:15 [notice] 1#1: start worker process 44
172.17.0.1 -- [25/Mar/2026:14:38:47 +0000] "GET / HTTP/1.1" 200 71 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/146.0.0.0 Safari/537.36 Edg/146.0.0.0" "-"
2026/03/25 14:38:47 [error] 29#29: *1 open() "/usr/share/nginx/html/favicon.ico" failed (2: No such file or directory), client: 172.17.0.1, server: localhost, request: "GET /favicon.ico HTTP/1.1", host: "localhost:88", referrer: "http://localhost:88/"
172.17.0.1 -- [25/Mar/2026:14:38:47 +0000] "GET /favicon.ico HTTP/1.1" 404 555 "http://localhost:88/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/146.0.0.0 Safari/537.36 Edg/146.0.0.0" "-"
sio@G102-GB16:~$
```

- J'arrête et je supprime le conteneur web ainsi que le mappage de volume :

```
sio@G102-GB16:~$ docker rm -fv web
web
sio@G102-GB16:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
sio@G102-GB16:~$
```

```
sio@G102-GB16:~$ cd web
sio@G102-GB16:~/web$ ls
index.html
sio@G102-GB16:~/web$
```

## 11. Gestion des volumes en écriture.

- Je récupère l'image mongodb-community-server avec la commande docker pull :

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

sio@G102-GB16:~$ docker pull mongodb/mongodb-community-server:latest
latest: Pulling from mongodb/mongodb-community-server
ee6c644e0c20: Pull complete
2e4f7f4523be: Pull complete
d68094177302: Pull complete
fe2fbbbc7ec0: Pull complete
4f4fb700ef54: Pull complete
f3e0282bbb14: Pull complete
6ad2e513b847: Pull complete
568a1e397067: Pull complete
96c832531c38: Pull complete
a9af3f48621c: Pull complete
6989f79dc9b5: Pull complete
Digest: sha256:7266392360c651bf99ad725eb78ee17d723e51a8f4cb7c3382aa89e7b2c3eee3
Status: Downloaded newer image for mongodb/mongodb-community-server:latest
docker.io/mongodb/mongodb-community-server:latest
sio@G102-GB16:~$
```

- Je lance un conteneur en mode serveur à partir de l'image mongodb-community-server :

```
sio@G102-GB16:~$ docker run -d -p 27017:27017 -e MONGODB_INITDB_ROOT_USERNAME=sio -e MONGODB_INITDB_ROOT_PASSWORD=password1234 --name mongodb mongodb/mongodb-community-server:latest
35bb51835cd5d85bc54d9e4ccfe0e9c69e298b366f30341821cfe07db19a960e
sio@G102-GB16:~$
```

- J'installe MongoDB Shell sur la machine physique pour pouvoir me connecter au serveur MongoDB :

MongoDB Atlas

MongoDB Enterprise Advanced

MongoDB Community Edition

Tools

MongoDB Atlas Terraform Provider

**MongoDB Shell**

MongoDB Compass (GUI)

Atlas CLI

Atlas Kubernetes Operator

MongoDB CLI for Cloud Manager  
and Ops Manager

Mongosync

MongoDB Relational Migrator

**Note:** MongoDB Shell is an open source (Apache 2.0), standalone product developed separately from the MongoDB Server.

**Learn more**

Version

2.8.1



Platform

Windows x64 (10+)



Package

msi

**Download** ↓

Copy link

**More Options** ⋮

- Je me connecte au serveur MongoDB :

```
PS C:\Users\Administrateur> mongosh mongodb://localhost:27017
```

- Je créer ma base de données et j'y inscrit des noms et je teste son bon fonctionnement :

```
Current Mongosh Log ID: 69c3f674ce67a67e95114029
Connecting to:      mongodb://localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000
&appName=mongosh+2.8.1
Using MongoDB:     6.0.6
Using Mongosh:     2.8.1
```

For mongosh info see: <https://www.mongodb.com/docs/mongodb-shell/>

To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (<https://www.mongodb.com/legal/privacy-policy>).  
You can opt-out by running the `disableTelemetry()` command.

```
-----
  The server generated these startup warnings when booting
  2026-03-23T14:56:29.032+01:00: Access control is not enabled for the database. Read and write access
  s to data and configuration is unrestricted
-----
```

```
test> use("ma_bdd");
switched to db ma_bdd
ma_bdd> db.createCollection("étudiants");
{ ok: 1 }
ma_bdd> db.etudiants.insertOne({"nom":"smet", "prenom":"jp"});
{
  acknowledged: true,
  insertedId: ObjectId('69c3f6eece67a67e9511402a')
}
ma_bdd> db.etudiants.find();
[
  {
    _id: ObjectId('69c3f6eece67a67e9511402a'),
    nom: 'smet',
    prenom: 'jp'
  }
]
ma_bdd> exit
PS C:\Users\Administrateur>
```

➤ J'inspecte mongodb avec docker :

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

```
sio@G102-GB16:~$ docker inspect mongodb
[
  {
    "Id": "35bb51835cd5d85bc54d9e4ccfe0e9c69e298b366f30341821cfe07db19a960e",
    "Created": "2026-03-25T14:47:37.85819296Z",
    "Path": "python3",
    "Args": [
      "/usr/local/bin/docker-entrypoint.py",
      "mongod"
    ],
    "State": {
      "Status": "running",
      "Running": true
    }
  }
]
```

```
"Mounts": [
  {
    "Type": "volume",
    "Name": "f891423f09ab2a2f661814659cbb3fd7e7ff03161a06146d2ef0bf7fc72a2b7f",
    "Source": "/var/lib/docker/volumes/f891423f09ab2a2f661814659cbb3fd7e7ff03161a06146d2ef0bf7fc72a2b7f/_data",
    "Destination": "/data/configdb",
    "Driver": "local",
    "Mode": "",
    "RW": true,
    "Propagation": ""
  },
  {
    "Type": "volume",
    "Name": "a0b811de3f993eae4f13412b42203003eef71914a2ed69b82371ddcd46831164",
    "Source": "/var/lib/docker/volumes/a0b811de3f993eae4f13412b42203003eef71914a2ed69b82371ddcd46831164/_data",
    "Destination": "/data/db",
    "Driver": "local",
    "Mode": "",
    "RW": true,
    "Propagation": ""
  }
]
```

```
"Env": [
  "MONGODB_INITDB_ROOT_USERNAME=sio",
  "MONGODB_INITDB_ROOT_PASSWORD=password1234",
  "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin",
  "HOME=/data/db",
  "GLIBC_TUNABLES=glibc.pthread.rseq=0"
],
"Cmd": [
  "mongod"
],
"Image": "mongodb/mongodb-community-server:latest",
"Volumes": {
  "/data/configdb": {},
  "/data/db": {}
},
}
```

➤ J'accède au conteneur puis aux données et arrête le conteneur :

```
sio@G102-GB16:~$ docker exec -it mongodb bash
mongodb@35bb51835cd5:/ $ cd /data
mongodb@35bb51835cd5:/data$ ls
configdb db
mongodb@35bb51835cd5:/data$ cd db
mongodb@35bb51835cd5:~$ ls
WiredTiger diagnostic.data
WiredTiger.lock index-3ba44be8-5f2f-4028-b070-3f3a469c95d1.wt
WiredTiger.turtle index-7b29566e-3438-4ec4-abc5-792481d37caa.wt
WiredTiger.wt index-b06f0c68-92aa-4092-947b-1946949a0ff1.wt
WiredTigerHS.wt index-bea7426f-95b5-4a58-a280-80d6e9a90840.wt
_mdb_catalog.wt index-e00083de-2bb1-4469-b39f-f29f7787c822.wt
_tmp index-f7003d95-71eb-40d6-92de-064e7cb93ded.wt
collection-818fb0c6-4dc6-494e-9723-c4acf0fa4761.wt journal
collection-8ccac561-a320-4190-b48d-53e8ff1568e1.wt mongod.lock
collection-d40fce1f-29e4-451a-b7c2-c1b59a77b341.wt sizeStorer.wt
collection-dfe18728-3ed3-49e4-a12c-e69384eeeff9.wt storage.bson
mongodb@35bb51835cd5:~$ exit
exit
sio@G102-GB16:~$
```

- Je supprime le conteneur : les données vont être perdues :

```
sio@G102-GB16:~$ docker rm -f mongodb
mongodb
sio@G102-GB16:~$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS        NAMES
sio@G102-GB16:~$
```

- Je souhaite bénéficier d'une persistance des données MongoDB dans Docker. Je crée un volume nommé à l'aide de la commande docker volume create :

```
sio@G102-GB16:~$ docker volume create mon_volume_mongodb
mon_volume_mongodb
sio@G102-GB16:~$
```

```
sio@G102-GB16:~$ docker volume ls
DRIVER      VOLUME NAME
local      a0b811de3f993eae4f13412b42203003eef71914a2ed69b82371ddcd46831164
local      f891423f09ab2a2f661814659cbb3fd7e7ff03161a06146d2ef0bf7fc72a2b7f
local      mon_volume_mongodb
sio@G102-GB16:~$
```

- Puis je l'inspecte :

```
sio@G102-GB16:~$ docker volume inspect mon_volume_mongodb
[
  {
    "CreatedAt": "2026-04-01T10:15:38Z",
    "Driver": "local",
    "Labels": null,
    "Mountpoint": "/var/lib/docker/volumes/mon_volume_mongodb/_data",
    "Name": "mon_volume_mongodb",
    "Options": null,
    "Scope": "local"
  }
]
sio@G102-GB16:~$
```

- Je lance un conteneur avec un mappage de volume (volume nommé:/volume du conteneur) :

```
sio@G102-GB16:~$ docker run -d --name mongodb -p 27017:27017 -e MONGODB_INITDB_ROOT_USERNAME=sio -e MONGODB_INITDB_ROOT_PASSWORD=password1234 -v mon_volume_mongodb:/data/db mongodb/mongodb-community-server:latest
18c4c99a454e23465ed6b7c3fcc9211ffc32380c310925be4f1d96973919965a
sio@G102-GB16:~$
```

```

mongodb:localhost:27017> use("ma_bdd");
switched to db ma_bdd
ma_bdd> db.createCollection("etudiants");
MongoServerError[NamespaceExists]: Collection ma_bdd.etudiants already exists.
ma_bdd> db.etudiants.insertOne({"nom":"smet", "prenom":"jp"});
{
  acknowledged: true,
  insertedId: ObjectId('69ccf254879aeea51411402a')
}
ma_bdd> db.etudiants.find();
[
  {
    _id: ObjectId('69c3f6eece67a67e9511402a'),
    nom: 'smet',
    prenom: 'jp'
  },
  {
    _id: ObjectId('69ccf254879aeea51411402a'),
    nom: 'smet',
    prenom: 'jp'
  }
]
ma_bdd> exit

```

```

sio@G102-GB16:~$ docker ps
CONTAINER ID   IMAGE                                COMMAND                                  CREATED        STAT
US            PORTS                                NAMES                                     5 minutes ago Up 5
18c4c99a454e  mongodb/mongodb-community-server:latest "python3 /usr/local/..."             5 minutes ago Up 5
0.0.0.0:27017->27017/tcp, [::]:27017->27017/tcp  mongodb
sio@G102-GB16:~$

```

- J'arrête le conteneur et je le supprime :

```

sio@G102-GB16:~$ docker stop mongodb
mongodb
sio@G102-GB16:~$ docker rv -v mongodb
Docker version 29.2.1, build a5c7197
sio@G102-GB16:~$ docker ps -a
CONTAINER ID   IMAGE                                COMMAND                                  CREATED        STAT
US            PORTS                                NAMES                                     6 minutes ago Exit
18c4c99a454e  mongodb/mongodb-community-server:latest "python3 /usr/local/..."             6 minutes ago Exit
ed (137) 12 seconds ago                mongodb
sio@G102-GB16:~$ docker rv -v mongodb
Docker version 29.2.1, build a5c7197

```

- Après sa suppression, je redémarre un conteneur mongo :

```

mongoserver1101: Authentication failed.
PS C:\Users\Administrateur> mongosh mongodb://localhost:27017
Current Mongosh Log ID: 69ccf489a3ce7987cb114029
Connecting to:      mongodb://localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000
&appName=mongosh+2.8.1
Using MongoDB:      6.0.6
Using Mongosh:      2.8.1
mongosh 2.8.2 is available for download: https://www.mongodb.com/try/download/shell

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

-----
  The server generated these startup warnings when booting
  2026-03-31T15:25:15.465+02:00: Access control is not enabled for the database. Read and write access
  to data and configuration is unrestricted
-----

test> use("ma_bdd")
switched to db ma_bdd
ma_bdd> db.etudiants.find();
[
  {
    _id: ObjectId('69c3f6eece67a67e9511402a'),
    nom: 'smet',
    prenom: 'jp'
  },
  {
    _id: ObjectId('69ccf254879aeaa51411402a'),
    nom: 'smet',
    prenom: 'jp'
  }
]
ma_bdd> exit

```

- Je supprime le conteneur :

```

sio@G102-GB16:~$ docker rm -fv mongodb
mongodb
sio@G102-GB16:~$

```

- Je relance un conteneur et j'ajoute une deuxième ligne pour vérifier la persistance :

```

sio@G102-GB16:~$ docker run -d --name mongodb -p 27017:27017 -e MONGODB_INITDB_ROOT_USERNAME=sio -e MONGODB_INITDB_ROOT_PASSWORD=password1234 -v mon_volume_mongodb:/data/db mongodb/mongodb-community-server:latest
47983bf0b3d24430bf84bca64ac5117d79732ed8a91a3336ba14676c33b07366
sio@G102-GB16:~$

```

```

test> use("ma_bdd");
switched to db ma_bdd
ma_bdd> db.etudiants.insertOne({"nom":"hallyday", "prenom":"johnny"});
{
  acknowledged: true,
  insertedId: ObjectId('69ccf5adccc1034a8f11402a')
}
ma_bdd> db.etudiants.find();
[
  {
    _id: ObjectId('69c3f6eece67a67e9511402a'),
    nom: 'smet',
    prenom: 'jp'
  },
  {
    _id: ObjectId('69ccf254879aeea51411402a'),
    nom: 'smet',
    prenom: 'jp'
  },
  {
    _id: ObjectId('69ccf5adccc1034a8f11402a'),
    nom: 'hallyday',
    prenom: 'johnny'
  }
]
ma_bdd> exit

```

- Je supprime le conteneur :

```

sio@G102-GB16:~$ docker rm -fv mongodb
mongodb
sio@G102-GB16:~$

```

- Je relance un conteneur et je constate la persistance des données :

```

sio@G102-GB16:~$ docker run -d --name mongodb -p 27017:27017 -e MONGODB_INITDB_ROOT_USERNAME=sio -e MONGODB_INITDB_ROOT_PASSWORD=password1234 -v mon_volume_mongodb:/data/db mongodb/mongodb-community-server:latest
e022bcea88dbf163f14ae350b4ff112fde09cdf24d41448768f90676408657ef
sio@G102-GB16:~$

```

```
PS C:\Users\Administrateur> mongosh mongodb://localhost:27017
Current Mongosh Log ID: 69ccf61cc1e40850dd114029
Connecting to:      mongodb://localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000
&appName=mongosh+2.8.1
Using MongoDB:      6.0.6
Using Mongosh:      2.8.1
mongosh 2.8.2 is available for download: https://www.mongodb.com/try/download/shell

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

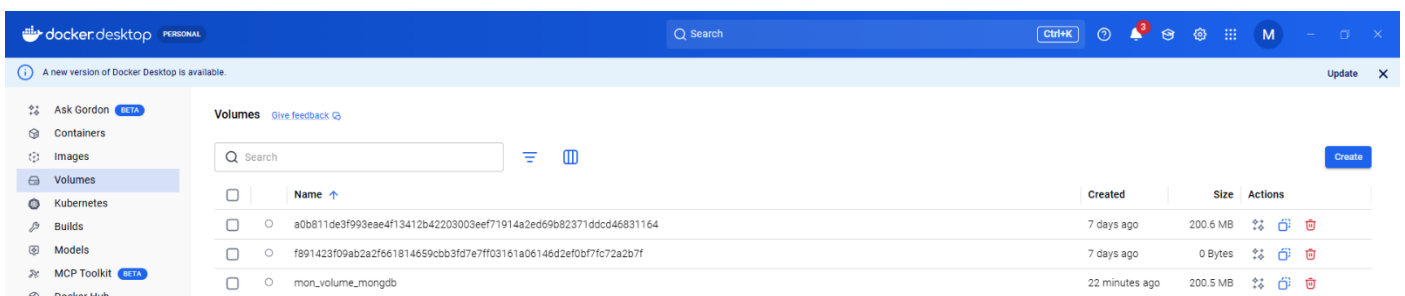
-----
The server generated these startup warnings when booting
2026-03-31T15:25:15.465+02:00: Access control is not enabled for the database. Read and write access
to data and configuration is unrestricted
-----

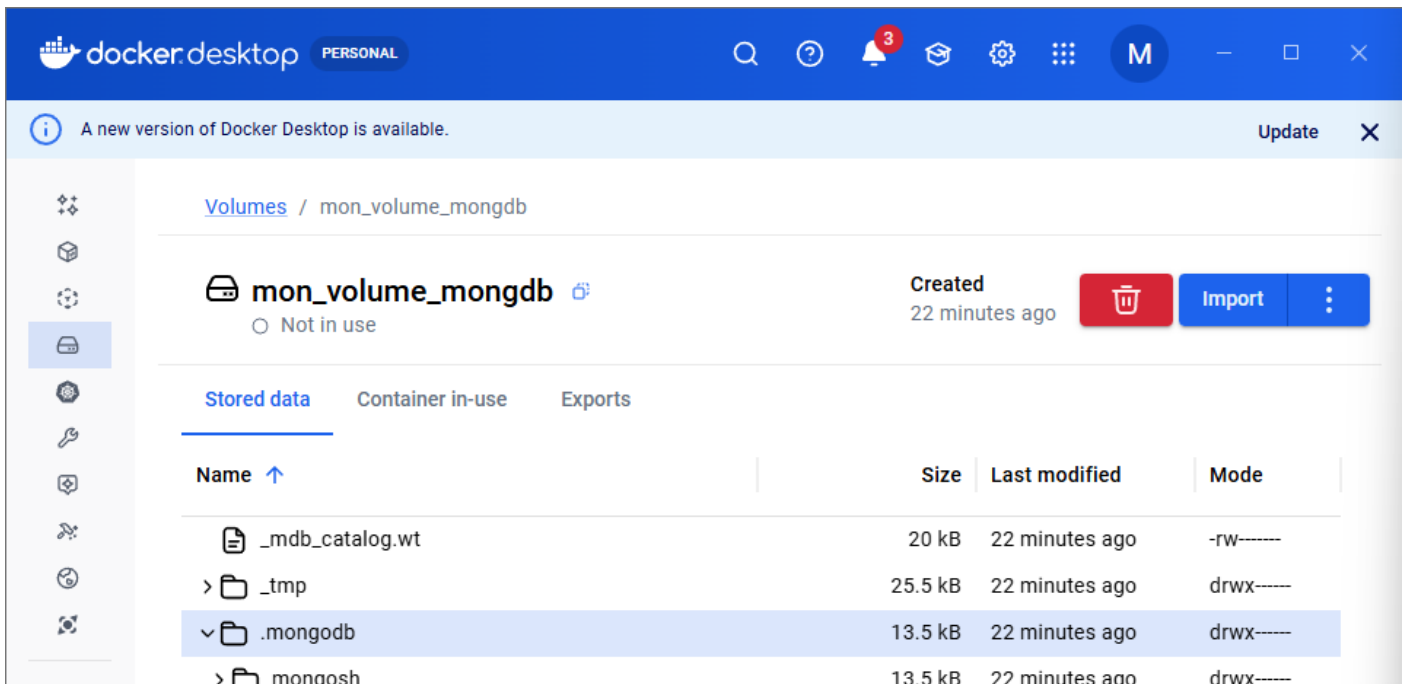
test> use("ma_bdd");
switched to db ma_bdd
ma_bdd> db.etudiants.find();
[
  {
    _id: ObjectId('69c3f6eece67a67e9511402a'),
    nom: 'smet',
    prenom: 'jp'
  },
  {
    _id: ObjectId('69ccf254879aeaa51411402a'),
    nom: 'smet',
    prenom: 'jp'
  },
  {
    _id: ObjectId('69ccf5adccc1034a8f11402a'),
    nom: 'hallyday',
    prenom: 'johnny'
  }
]
ma_bdd> exit
```

➤ Je supprime le conteneur :

```
sio@G102-GB16:~$ docker rm -fv mongodb
mongodb
sio@G102-GB16:~$
```

➤ Sur Docker je me rends sur l'onglet Volumes et je constate la présence du volume mongodb :





- Les montages liés comportent cependant des risques car ils exposent les chemins d'accès hôtes aux conteneurs, je créer le dossier data :

```
sio@G102-GB16:~$ mkdir data
sio@G102-GB16:~$
```

- Je lance un conteneur que j'appelle mongodb2 :

```
sio@G102-GB16:~$ docker run -d --name mongodb2 -p 27017:27017 -e MONGODB_INITDB_ROOT_USERNAME=sio -e M
ONGODB_INITDB_ROOT_PASSWORD=password1234 -v mon_volume_mongodb:/data/db mongodb/mongodb-community-serv
er:latest
14a250d1a755341f49a4ee93e993ce59bb4336f7c5dc66858ec1ad0284f20c40
sio@G102-GB16:~$
```

```

PS C:\Users\Administrateur> mongosh mongodb://localhost:27017
Current Mongosh Log ID: 69ccf6e5165906a1d9114029
Connecting to:      mongodb://localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000
&appName=mongosh+2.8.1
Using MongoDB:      6.0.6
Using Mongosh:      2.8.1
mongosh 2.8.2 is available for download: https://www.mongodb.com/try/download/shell

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

-----
  The server generated these startup warnings when booting
  2026-03-31T15:25:15.465+02:00: Access control is not enabled for the database. Read and write acces
  s to data and configuration is unrestricted
  -----

test> use("ma_bdd");
switched to db ma_bdd
ma_bdd> db.etudiants.insertOne({"nom":"mitchell", "prenom":"eddy"});
{
  acknowledged: true,
  insertedId: ObjectId('69ccf71a165906a1d911402a')
}
ma_bdd> db.etudiants.find()
[
  {
    _id: ObjectId('69c3f6eece67a67e9511402a'),
    nom: 'smet',
    prenom: 'jp'
  },
  {
    _id: ObjectId('69ccf254879aeea51411402a'),
    nom: 'smet',
    prenom: 'jp'
  },
  {
    _id: ObjectId('69ccf5adccc1034a8f11402a'),
    nom: 'hallyday',
    prenom: 'johnny'
  },
  {
    _id: ObjectId('69ccf71a165906a1d911402a'),
    nom: 'mitchell',
    prenom: 'eddy'
  }
]
ma_bdd>

```

➤ Je stoppe le conteneur et je le supprime :

```

PS C:\Users\Administrateur> docker ps
CONTAINER ID   IMAGE                                COMMAND                                  CREATED        STAT
US            PORTS                               NAMES
14a250d1a755  mongodb/mongodb-community-server:latest  "python3 /usr/local/..."  2 minutes ago  Up 2
minutes      0.0.0.0:27017->27017/tcp, [::]:27017->27017/tcp  mongodb2
PS C:\Users\Administrateur> docker stop mongodb2
mongodb2
PS C:\Users\Administrateur> docker rm -v mongodb2
mongodb2
PS C:\Users\Administrateur>

```

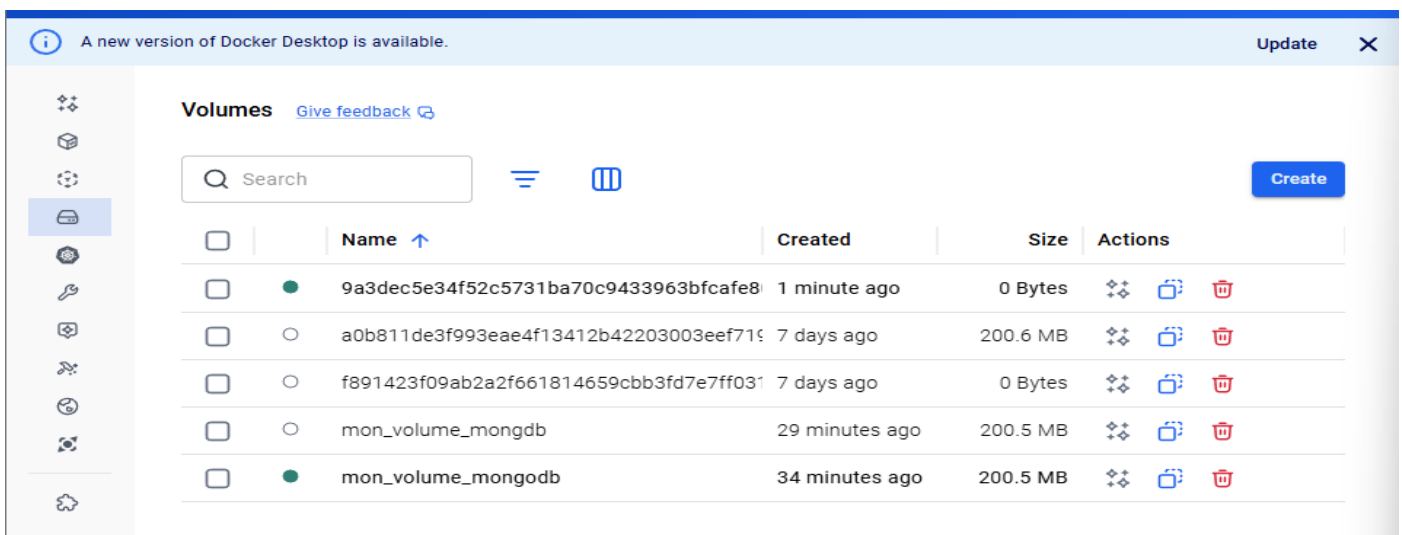
- Après suppression du conteneur, je relance une nouvelle instance de l'image :

```
sio@G102-GB16:~$ docker run -d --name mongodb2 -p 27017:27017 -e MONGODB_INITDB_ROOT_USERNAME=sio -e MONGODB_INITDB_ROOT_PASSWORD=password1234 -v mon_volume_mongodb:/data/db mongodb/mongodb-community-server:latest
a8f7c1b095d6a7770f1ca9555cd35c25a4b862c62159e6185a2c5896152b6ed7
sio@G102-GB16:~$
```

```
test> use("ma_bdd");
switched to db ma_bdd
ma_bdd> db.etudiants.find();
[
  {
    _id: ObjectId('69c3f6eece67a67e9511402a'),
    nom: 'smet',
    prenom: 'jp'
  },
  {
    _id: ObjectId('69ccf254879aeea51411402a'),
    nom: 'smet',
    prenom: 'jp'
  },
  {
    _id: ObjectId('69ccf5adccc1034a8f11402a'),
    nom: 'hallyday',
    prenom: 'johnny'
  },
  {
    _id: ObjectId('69ccf71a165906a1d911402a'),
    nom: 'mitchell',
    prenom: 'eddy'
  }
]
ma_bdd> exit
PS C:\Users\Administrateur>
```

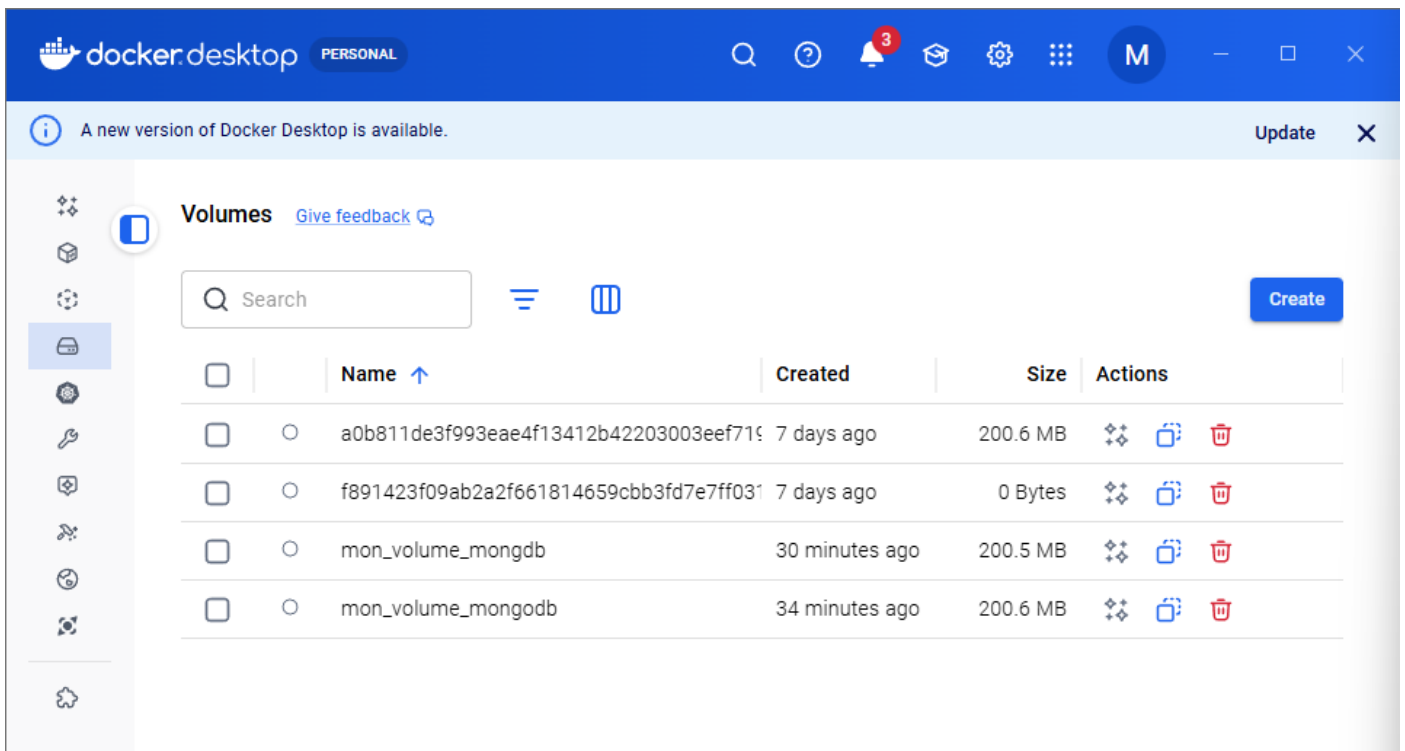
La persistance a bien été gérée

- Dans Docker je me rends dans l'onglet Volumes :



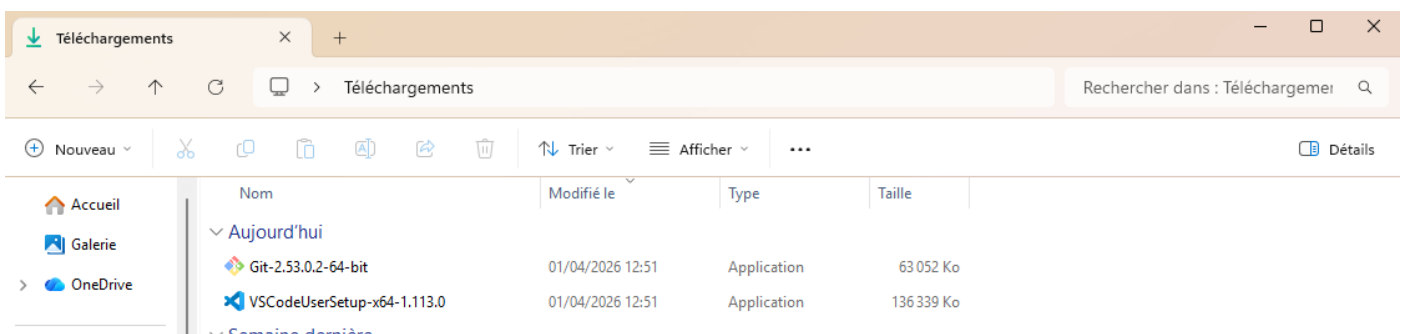
➤ Je supprime le conteneur :

```
sio@G102-GB16:~$ docker rm -fv mongodb2
mongodb2
sio@G102-GB16:~$
```

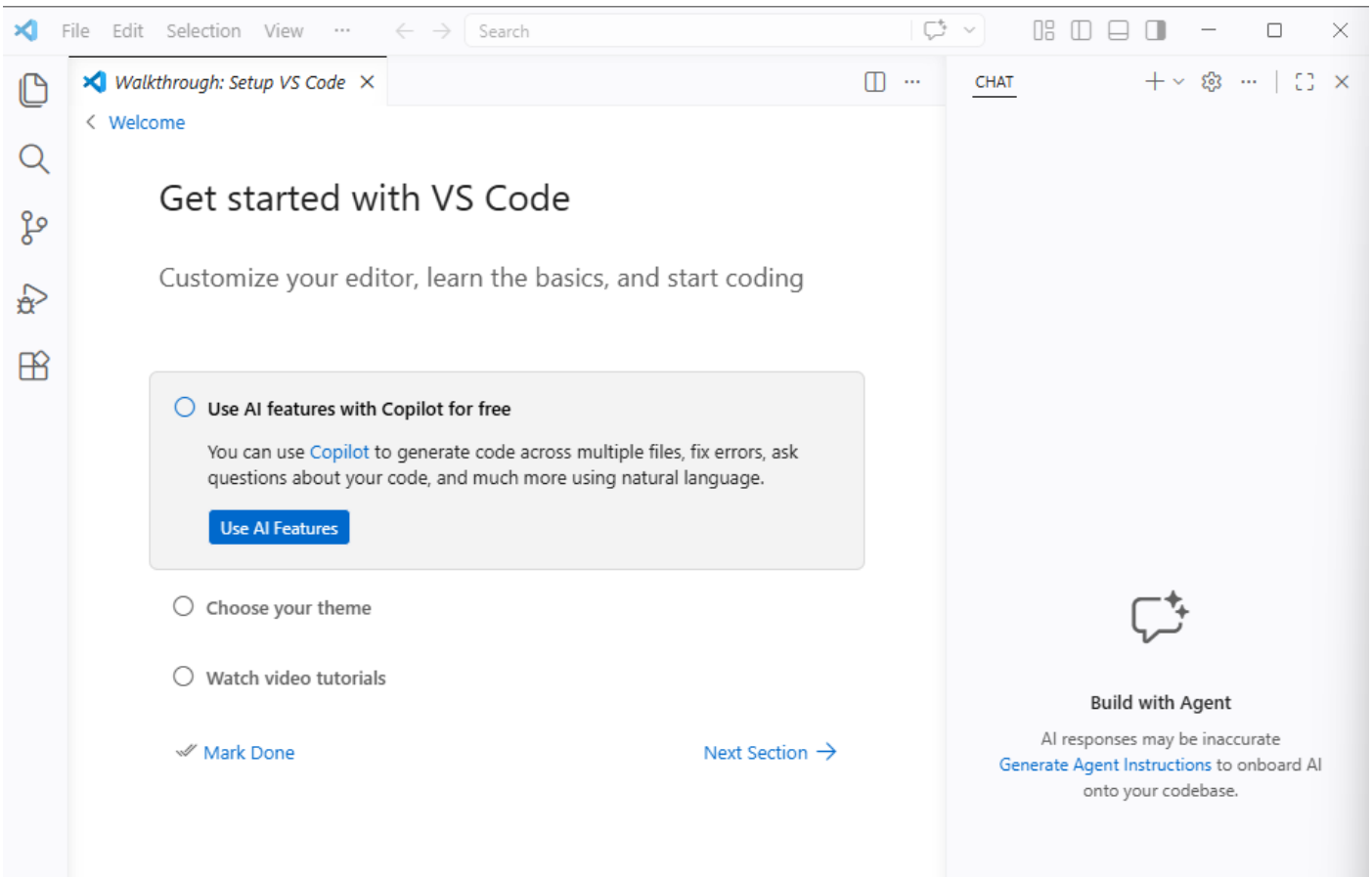


## 12. Utiliser un conteneur Docker et Visual Studio Code.

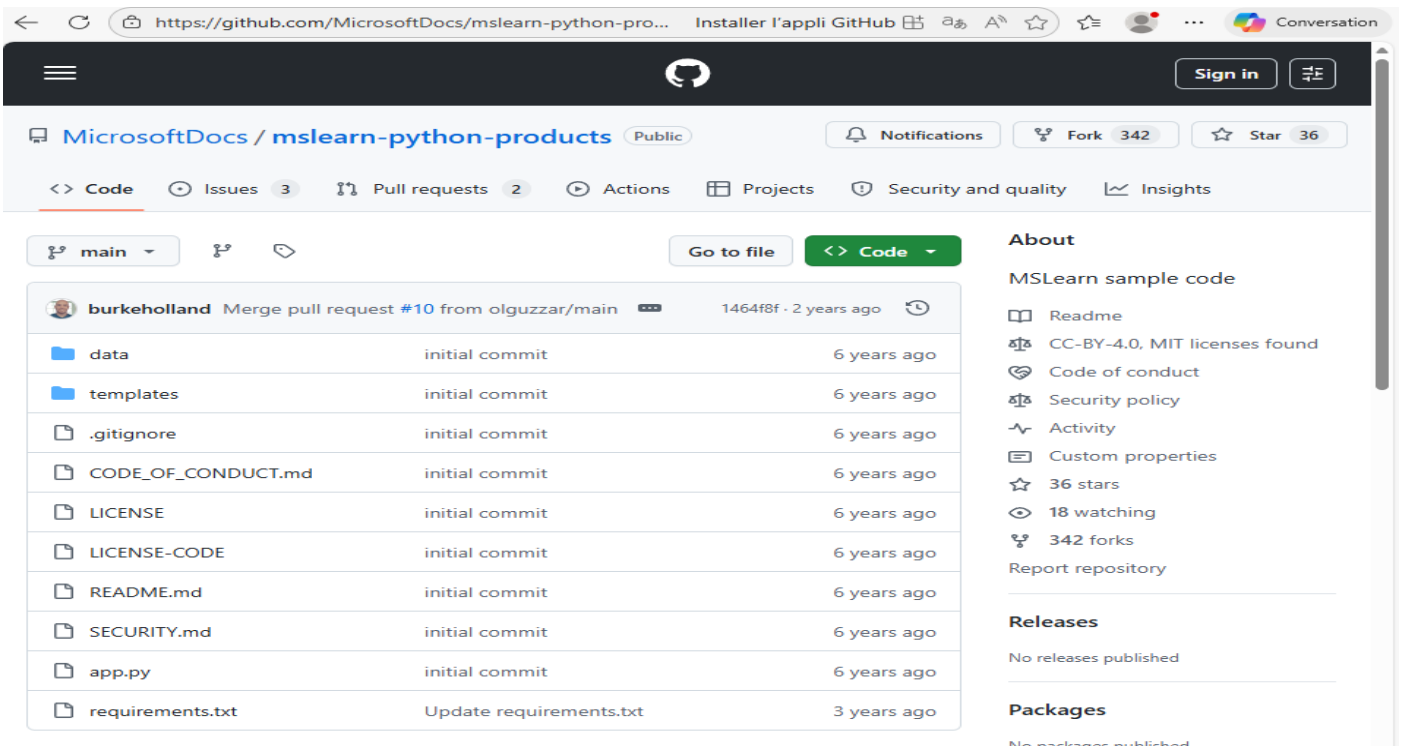
➤ J'installe VS Code :



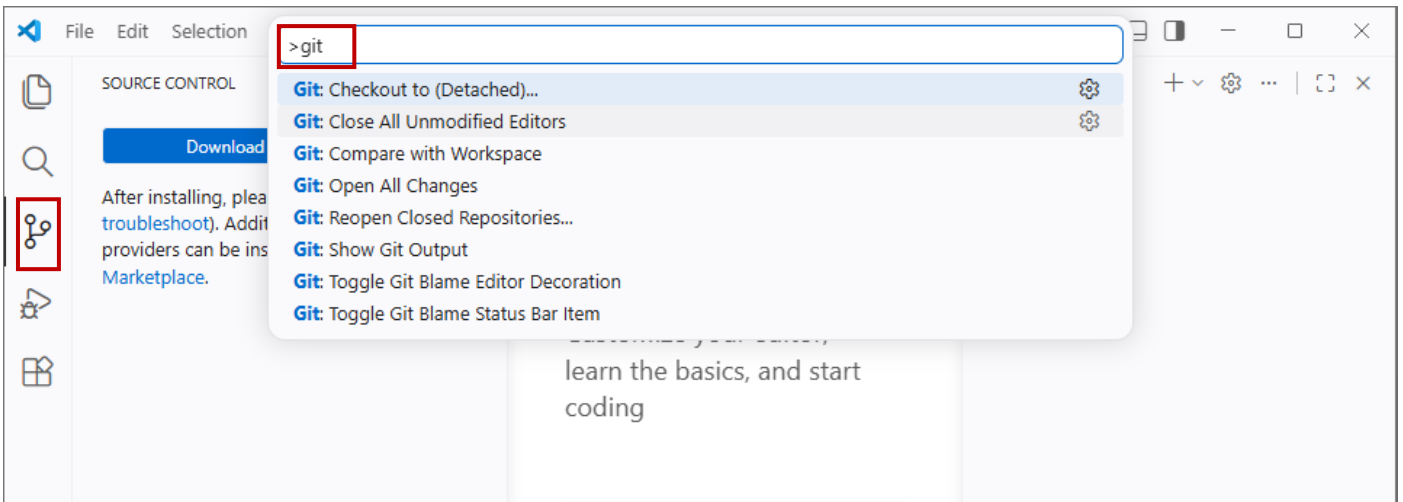
➤ J'ouvre une nouvelle instance de VS Code :



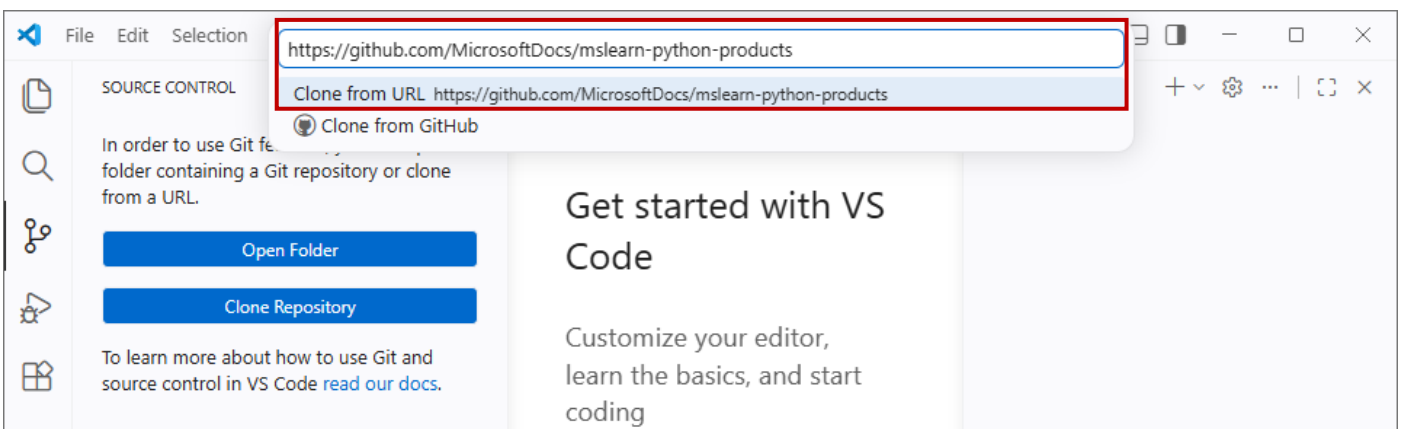
➤ Je copie l'URL du projet mslearn-python-products qui nous servira d'exemple :



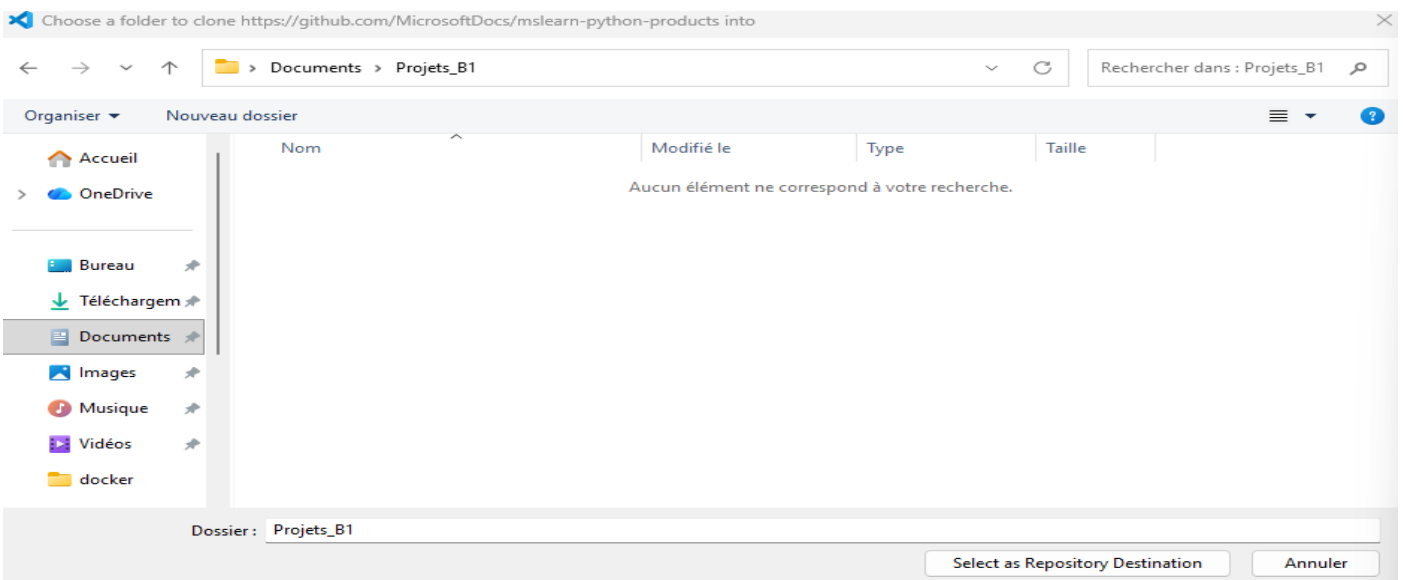
➤ Dans VS Code, je sélectionne Clone Repository pour rechercher Git : Clone :



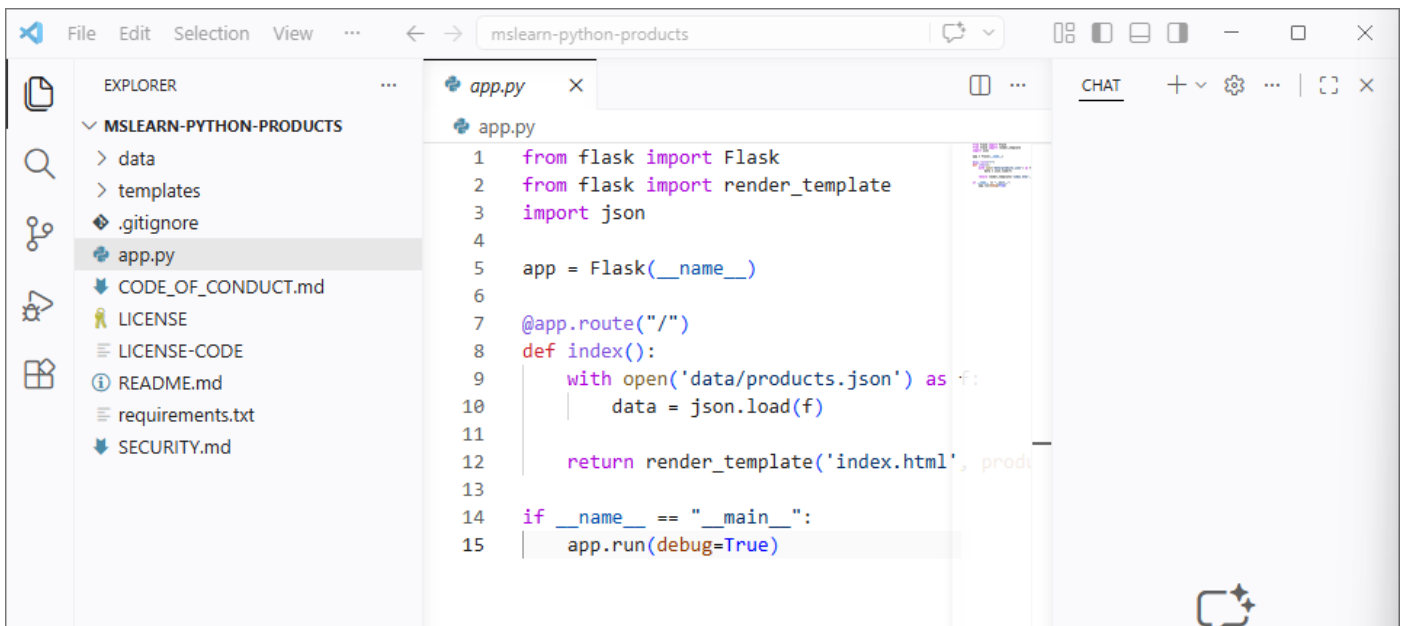
➤ Je colle l'URL :



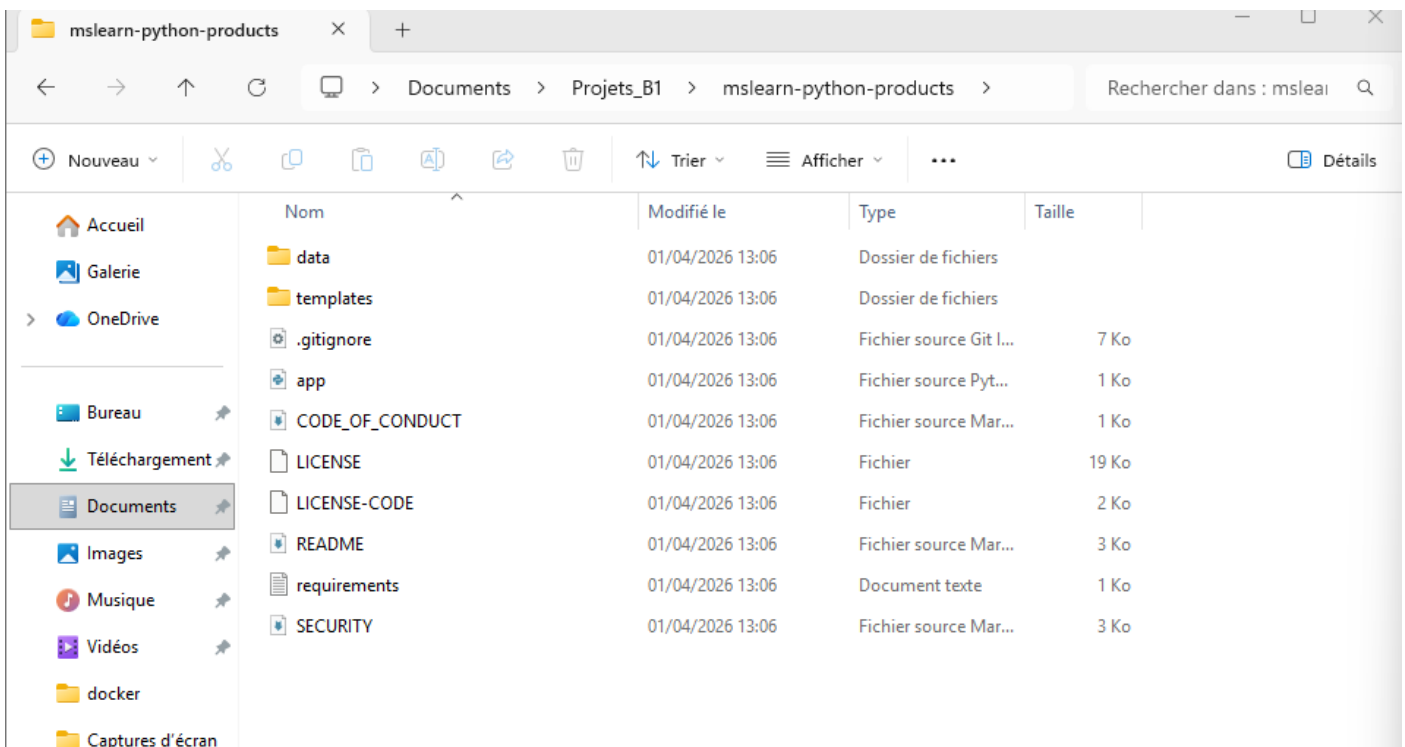
➤ Je sélectionne un emplacement sur mon disque où le projet va être cloné :



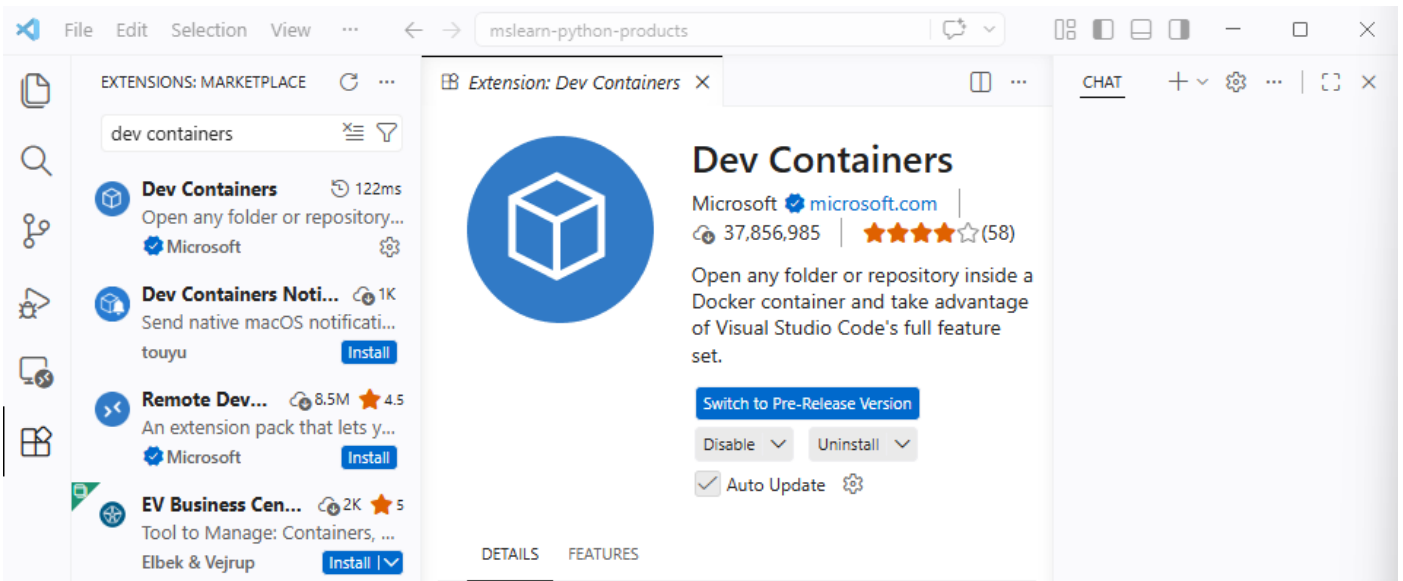
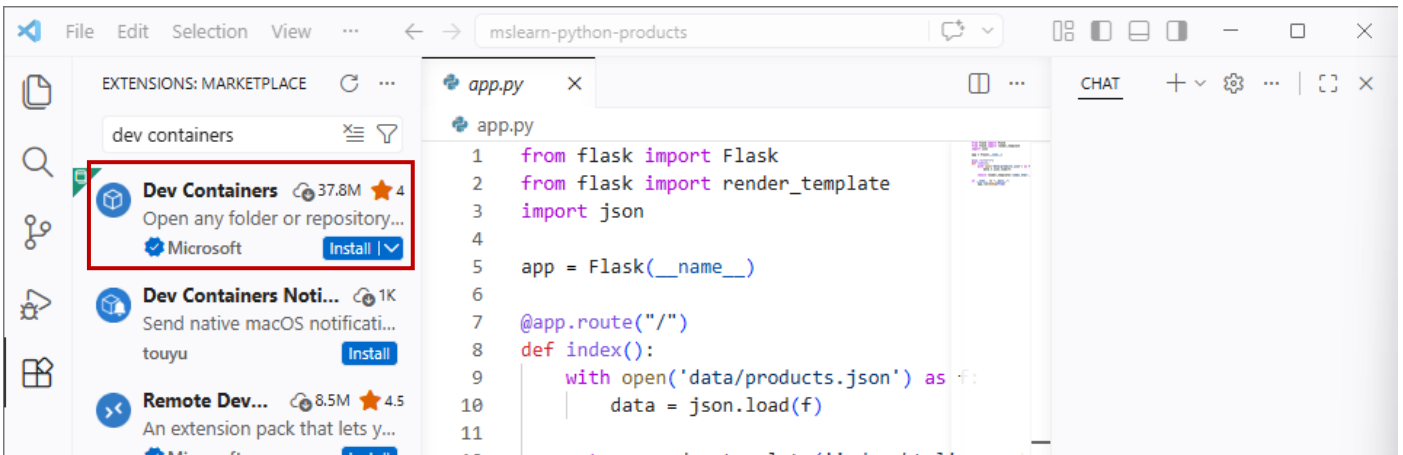
➤ Le projet est maintenant ouvert dans VS Code :



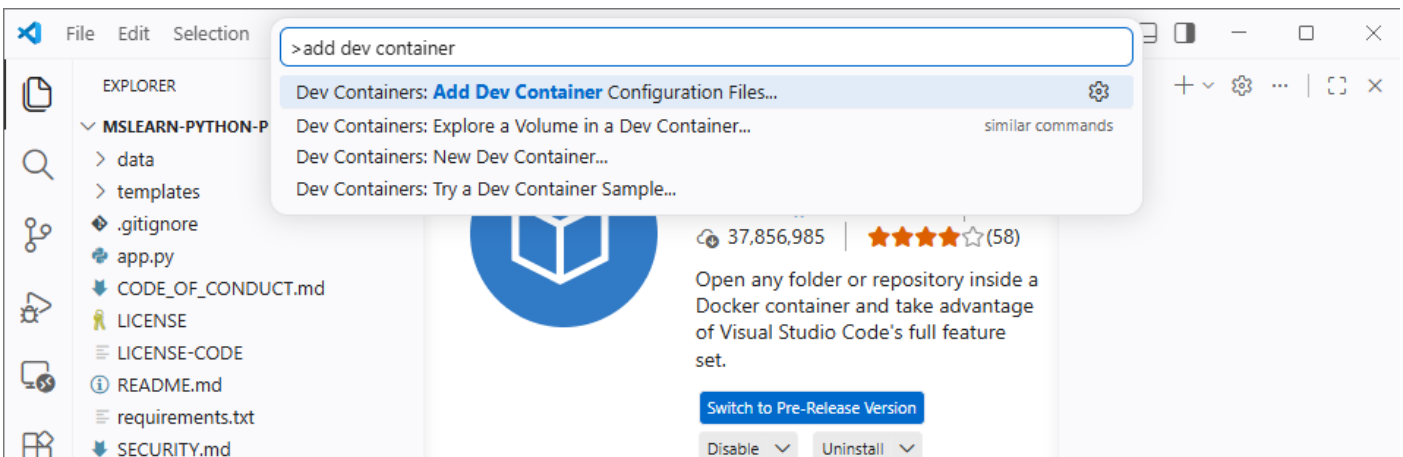
➤ Je me rends dans l'explorateur de fichier dans le répertoire du projet :



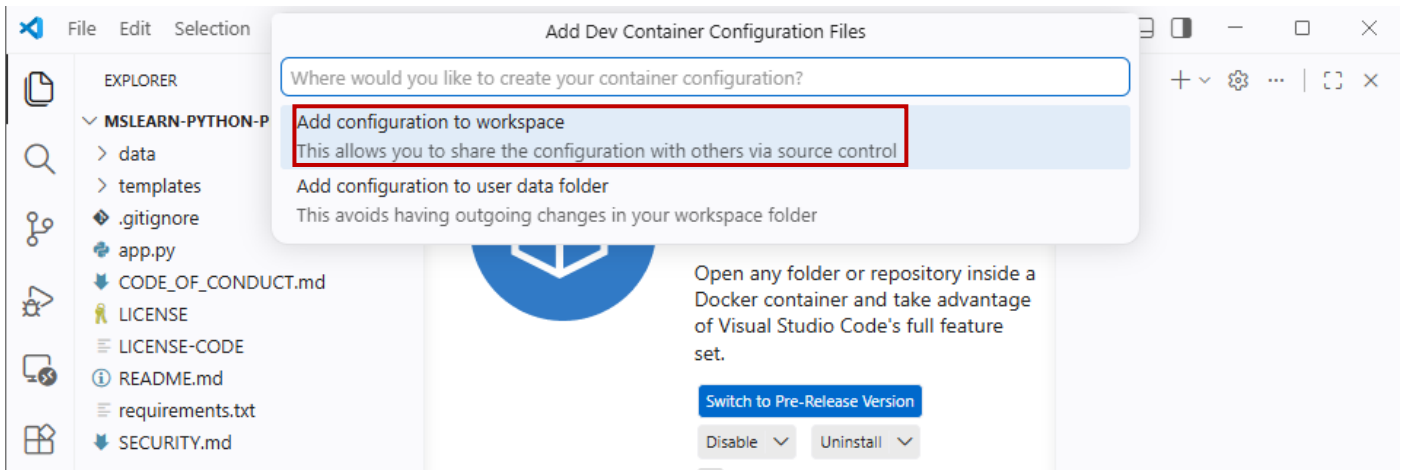
➤ J'installe l'extension Dev Containers :



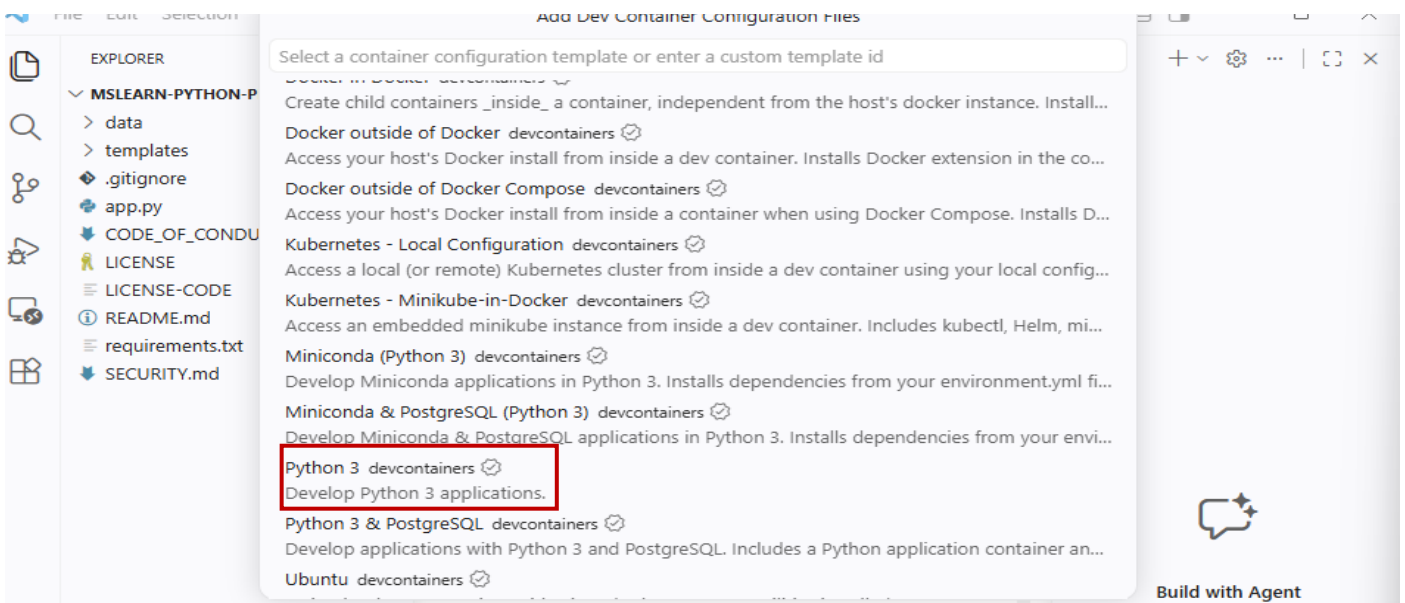
➤ Dans la palette de commandes je tape add dev container et je sélectionne Dev Containers: Add Dev Container Configuration Files :



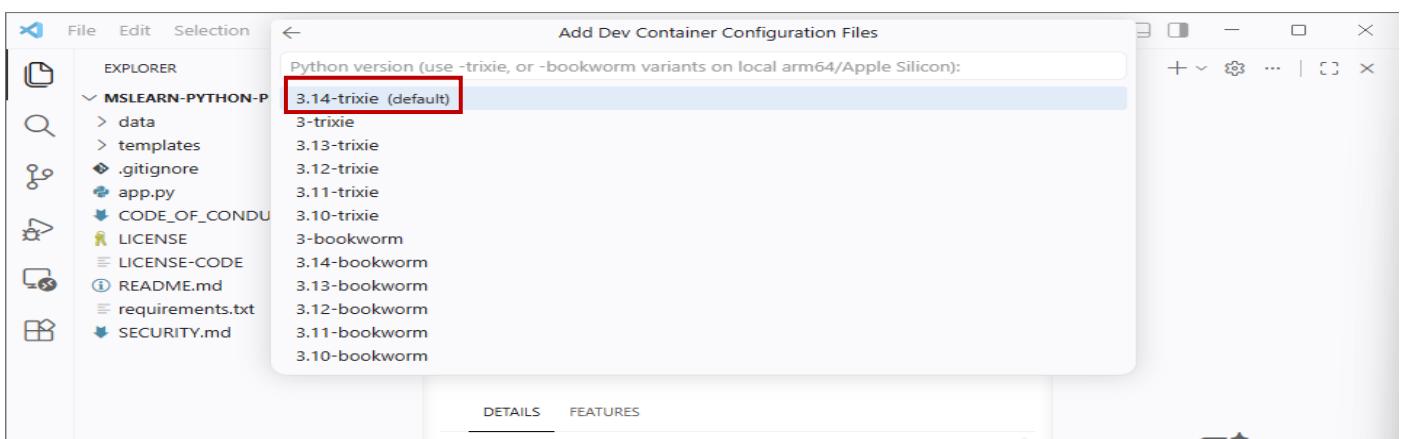
➤ Je sélectionne Add configuration to workspace :



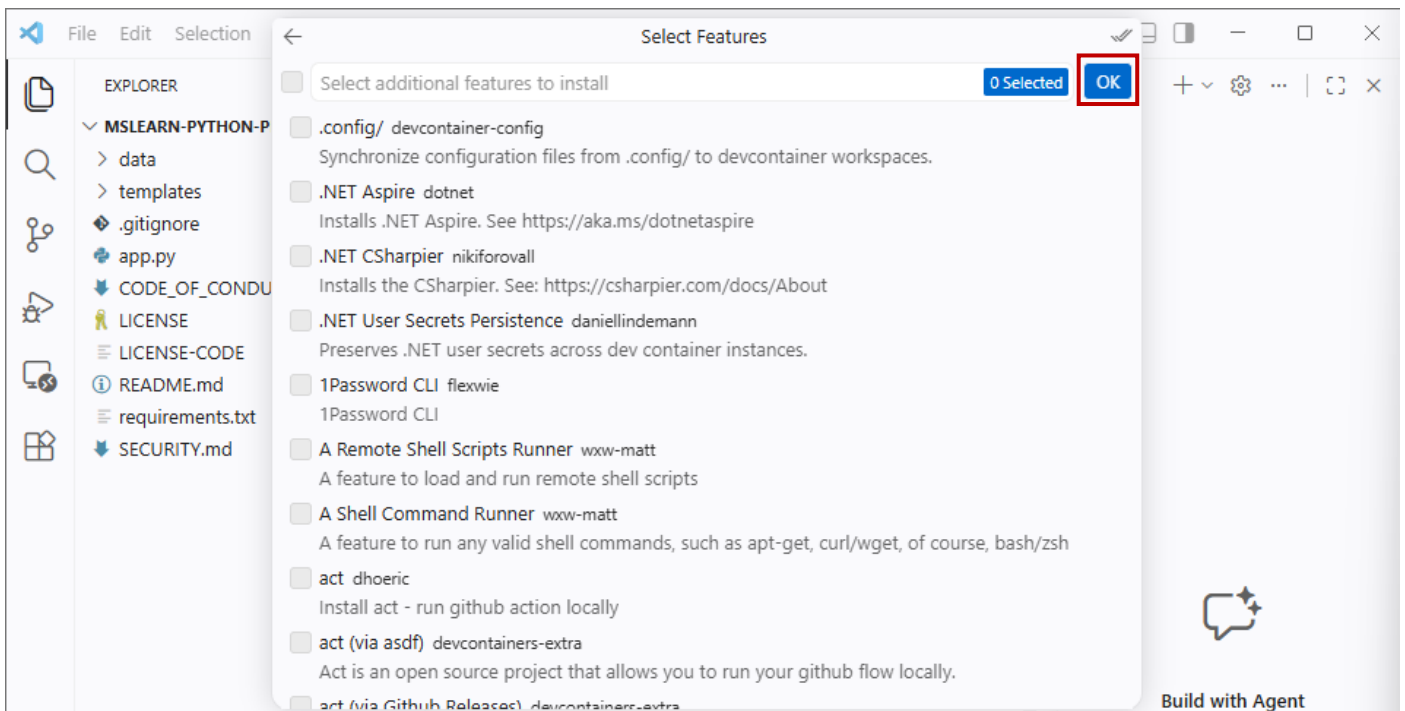
➤ Je choisis Python 3 :



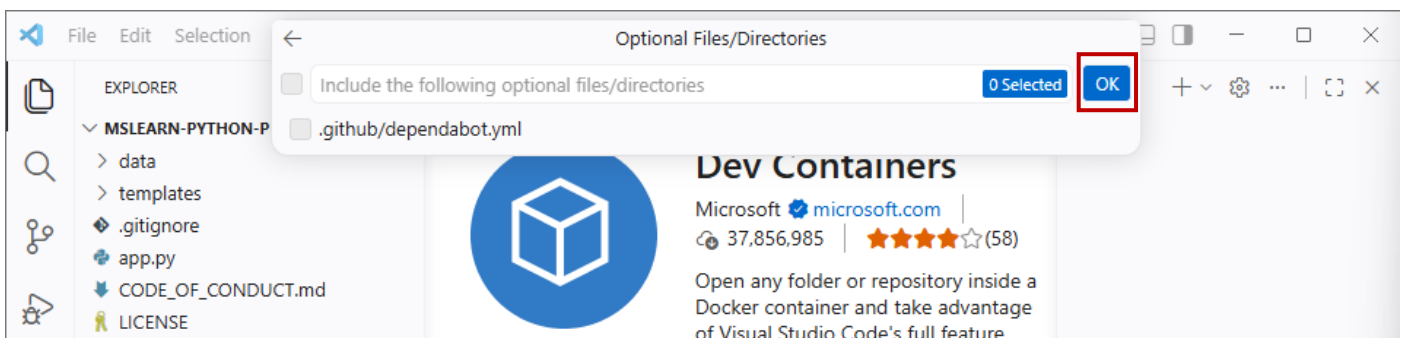
➤ Je laisse par défaut :



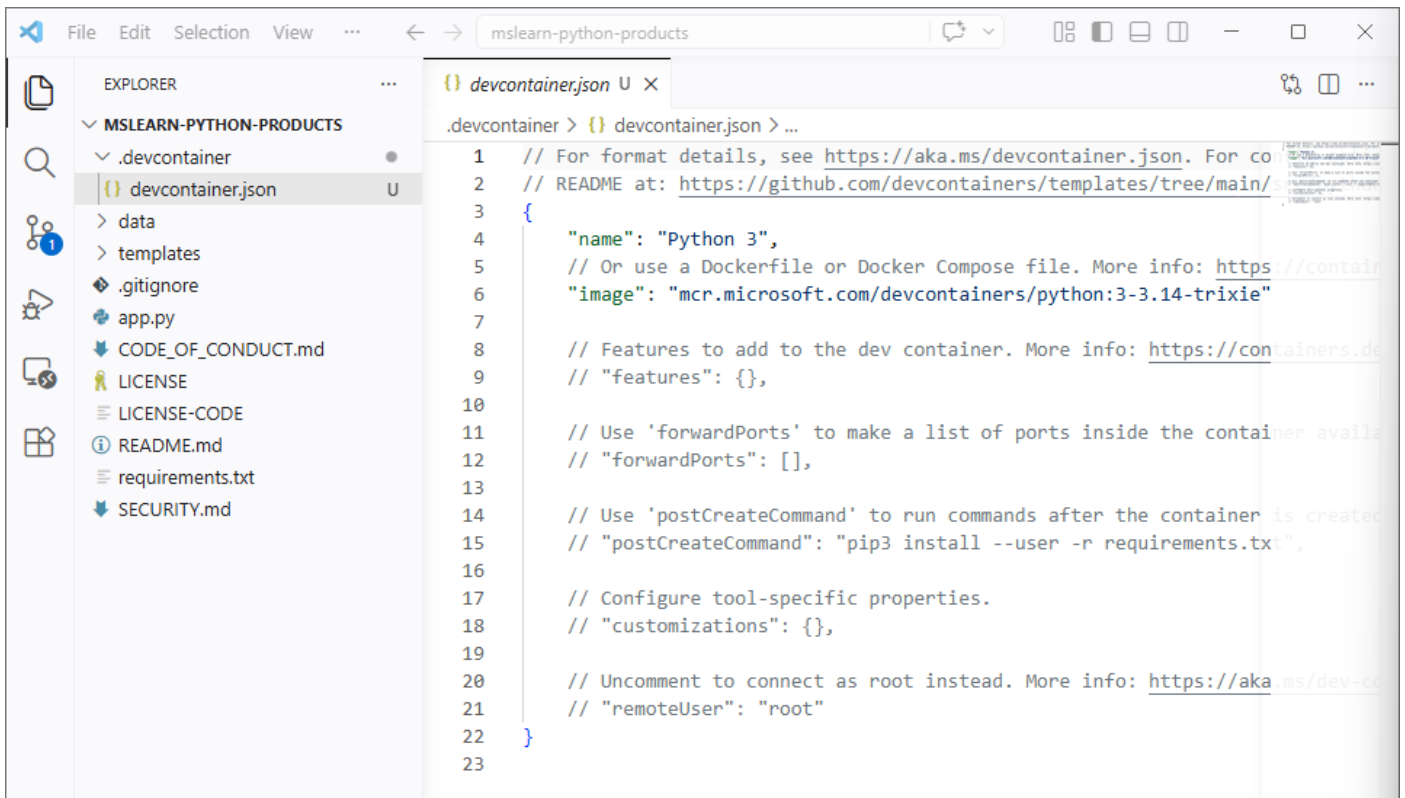
➤ Je sélectionne rien et je fais ok :



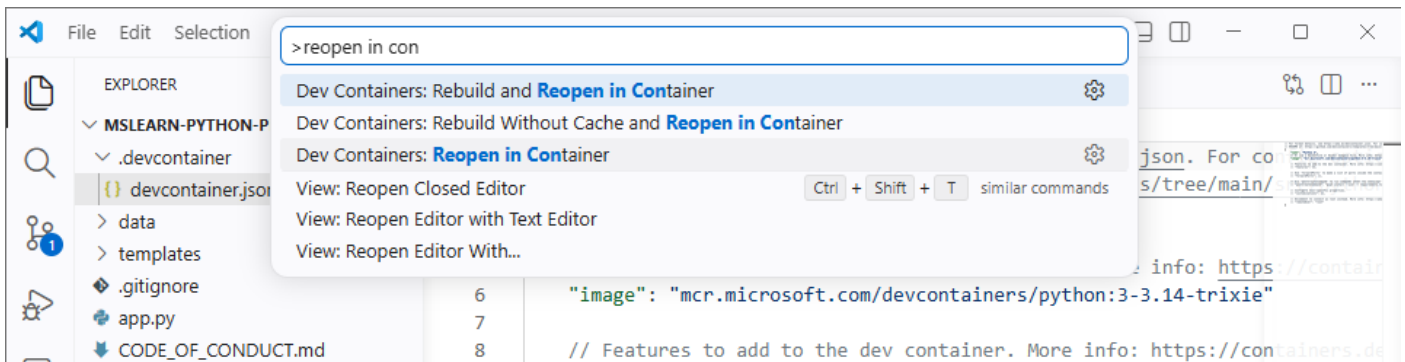
➤ Idem :



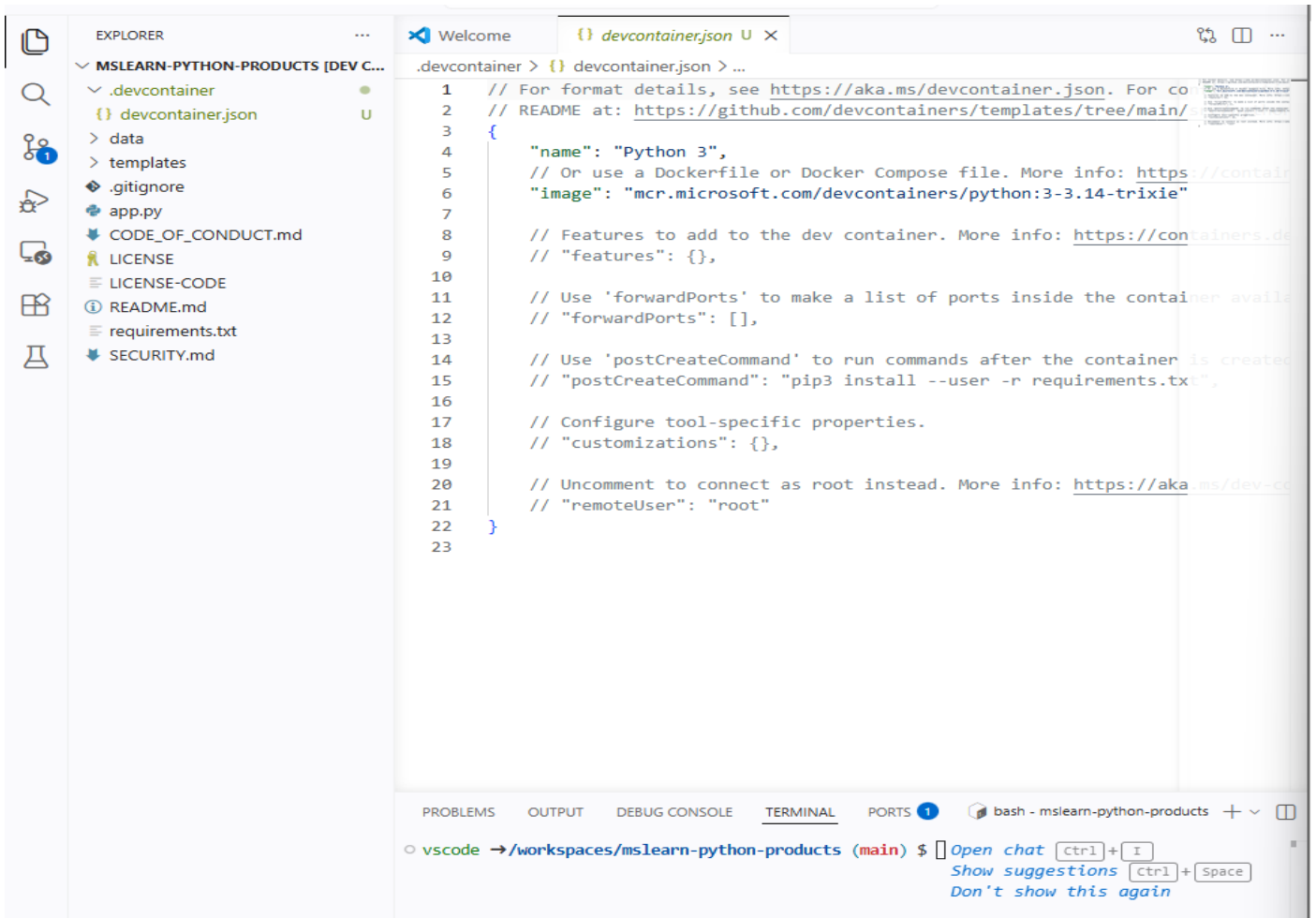
➤ Un nouveau dossier nommé « .devcontainer » a été ajouté au projet. Je développe ce dossier : il contient un fichier devcontainer.json :



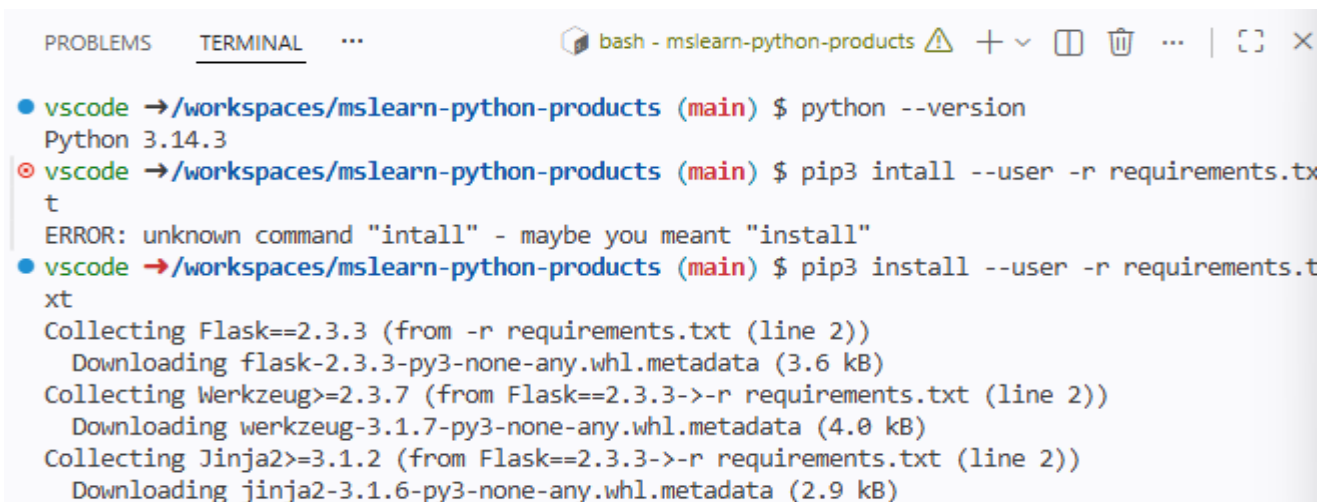
- J'appuis sur F1 pour ouvrir la palette de commandes, je tape reopen in container et je sélectionne Dev Containers: Reopen in Container dans la liste des options disponibles :



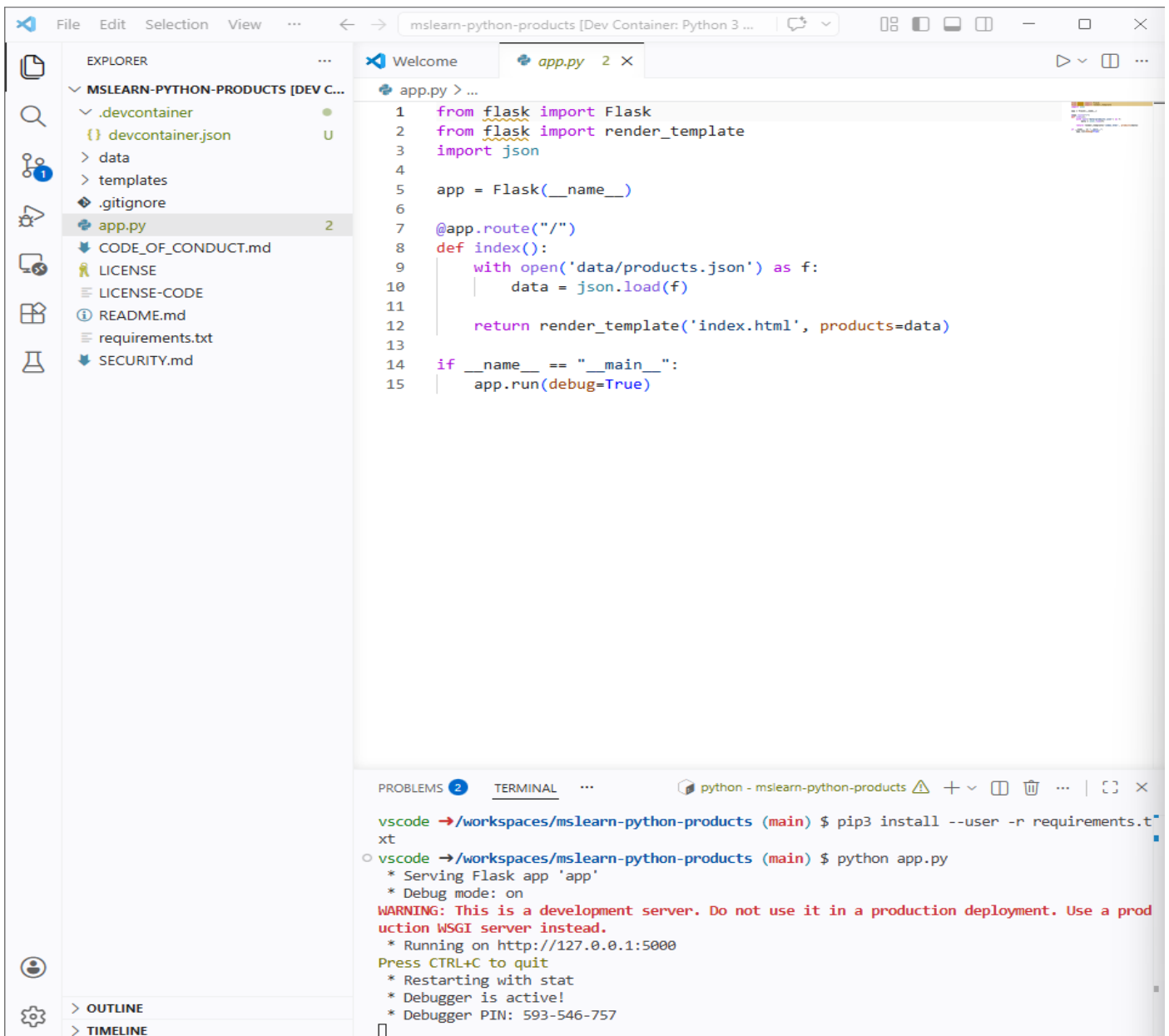
- J'examine l'indicateur distant en bas à gauche de VS Code : il affiche à présent « Dev Container: Python 3 » :



- Dans le terminal intégré à VS code j'exécute une première commande pour savoir si python est installé, puis j'exécute la commande `pip3 install --user -r requirements.txt` dans le terminal pour installer les dépendances Flask nécessaires à l'exécution du projet :



➤ J'entre la commande python app.py dans le terminal pour exécuter le projet :



The screenshot shows the Visual Studio Code interface. On the left, the Explorer view shows the project structure for 'MSLEARN-PYTHON-PRODUCTS [DEV C...]' with files like .devcontainer, data, templates, .gitignore, app.py, CODE\_OF\_CONDUCT.md, LICENSE, LICENSE-CODE, README.md, requirements.txt, and SECURITY.md. The main editor displays the code for 'app.py':

```
1 from flask import Flask
2 from flask import render_template
3 import json
4
5 app = Flask(__name__)
6
7 @app.route("/")
8 def index():
9     with open('data/products.json') as f:
10         data = json.load(f)
11
12     return render_template('index.html', products=data)
13
14 if __name__ == "__main__":
15     app.run(debug=True)
```

At the bottom, the Terminal view shows the execution of the application:

```
vscode →/workspaces/mslearn-python-products (main) $ pip3 install --user -r requirements.txt
vscode →/workspaces/mslearn-python-products (main) $ python app.py
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 593-546-757
```

➤ Je montre que l'application web Python avec Flask s'est bien exécutée sur votre machine en saisissant `http://127.0.0.1:5000` depuis votre navigateur :

localhost:5000

# Products

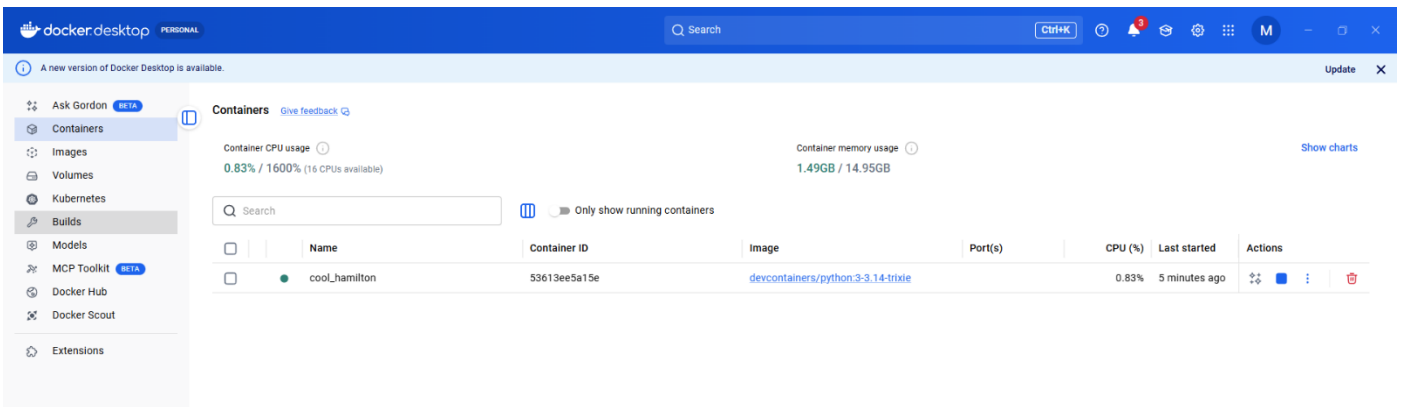
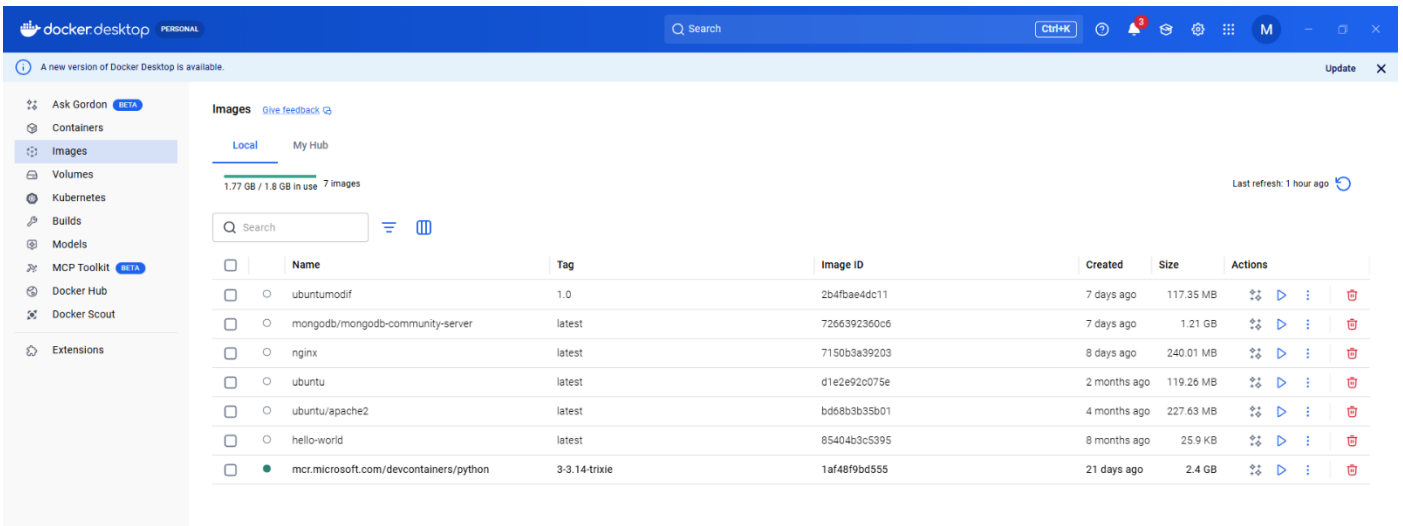
Name	Brand	Price	Units in Stock
Single red garden gnome	Home & Pro tools	56	98
Two red garden gnomes	Home & Pro tools	92	4
One sat gnome	Home & Pro tools	34	34
One sat on shoe gnome	Home & Pro tools	54	54
One barrow gnome	Home & Pro tools	29	23
One glasses gnome	Home & Pro tools	54	94
One smiling gnome	Home & Pro tools	43	45
Two singing gnomes	Home & Pro tools	65	78
Two sleeping gnomes	Home & Pro tools	32	12
Seven-pack gnomes	Home & Pro tools	2543	45
One afraid gnome	Home & Pro tools	39	88
One welcome gnome	Home & Pro tools	28	65
Two smiling gnomes	Home & Pro tools	76	87
Microwave 33.9 Cu. Ft. 92325 W	ElctroDrill	1740	58
Refrigerator 1.7 cu. ft. 110 watts	ProSaws	200	54
Oven 900 W	Drills Co	300	389
Washing machine 1200rpm	Home & Pro tools	400	45

➤ Puis dans VS Code :

```
mslearn-python-products [Dev Container: Python 3 ...]
Welcome {} products.json > ...
1 [
2   {
3     "name": "Single red garden gnome",
4     "price": 56,
5     "brand": {
6       "name": "Home & Pro tools"
7     },
8     "stockUnits": 98
9   },
10  {
11    "name": "Two red garden gnomes",
12    "price": 92,
13    "brand": {
14      "name": "Home & Pro tools"
15    },
16    "stockUnits": 4
17  },
18  {
19    "name": "One sat gnome",
20    "price": 34,
21    "brand": {
22      "name": "Home & Pro tools"
23    },
24    "stockUnits": 34
25  },
26  {
27    "name": "One sat on shoe gnome",
28    "price": 54,
29    "brand": {
30      "name": "Home & Pro tools"
31    },
32    "stockUnits": 54
33  },
34  ]
```

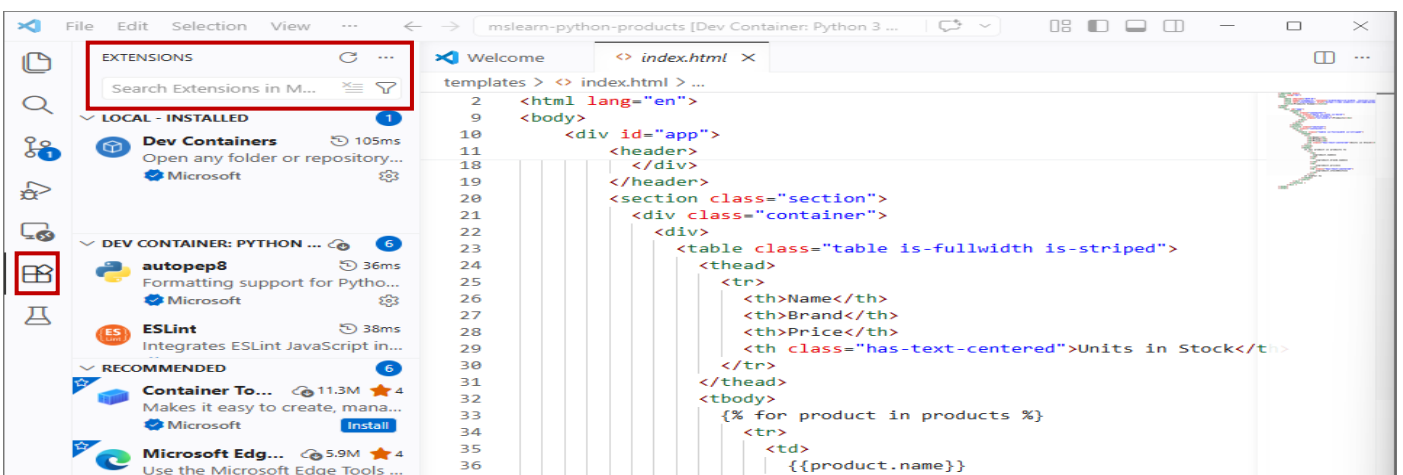
```
mslearn-python-products [Dev Container: Python 3 ...]
Welcome <> index.html > ...
2 <html lang="en">
9 <body>
10 <div id="app">
11 <header>
18 </div>
19 </header>
20 <section class="section">
21 <div class="container">
22 <div>
23 <table class="table is-fullwidth is-striped">
24 <thead>
25 <tr>
26 <th>Name</th>
27 <th>Brand</th>
28 <th>Price</th>
29 <th class="has-text-centered">Units in Stock</th>
30 </tr>
31 </thead>
32 <tbody>
33 <tr>
34 <td>
35 <td>
36 <td>
37 <td>
38 <td>
39 <td>
40 <td>
41 <td>
42 <td>
43 <td>
44 <td class="has-text-centered">
45 <td>
46 <td>
47 <td>
```

## ➤ Idem dans Docker

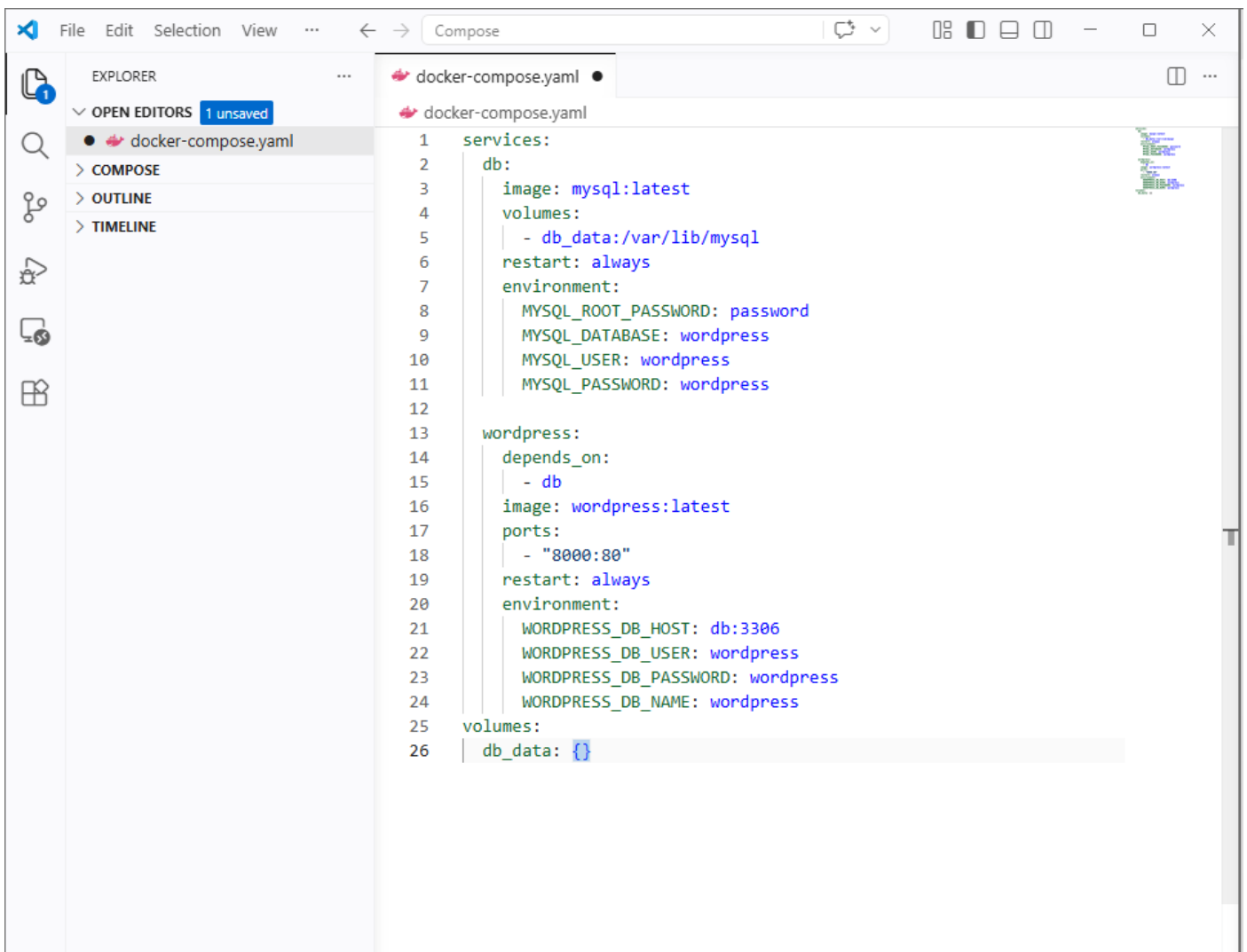
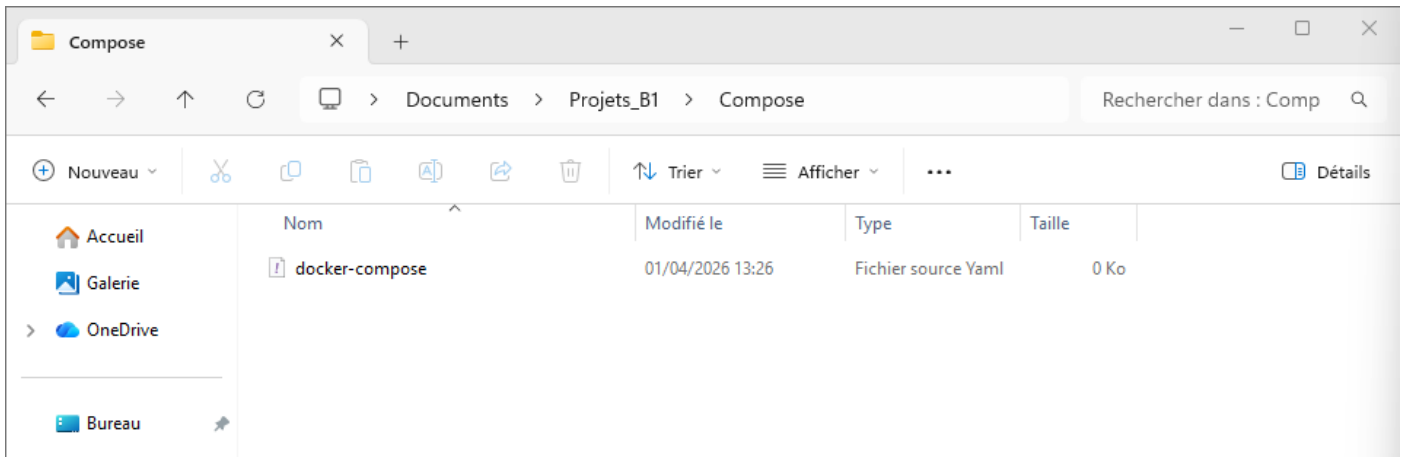


## 13. Monter une stack applicative avec Docker Compose.

➤ J'installe l'extension YAML dans VS Code :



- Afin de monter un blog Wordpress avec sa base de données, je créé dans un dossier le fichier docker-compose.yml suivant à l'aide de VS Code :



- Depuis le terminal, je lance la stack applicative avec la commande docker compose up :

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS powershell + v [ ] [ ] ... | [ ] [ ] X  
PS C:\Users\Administrateur\Documents\Projets_B1\Compose> PS docker compose up [ ]
```

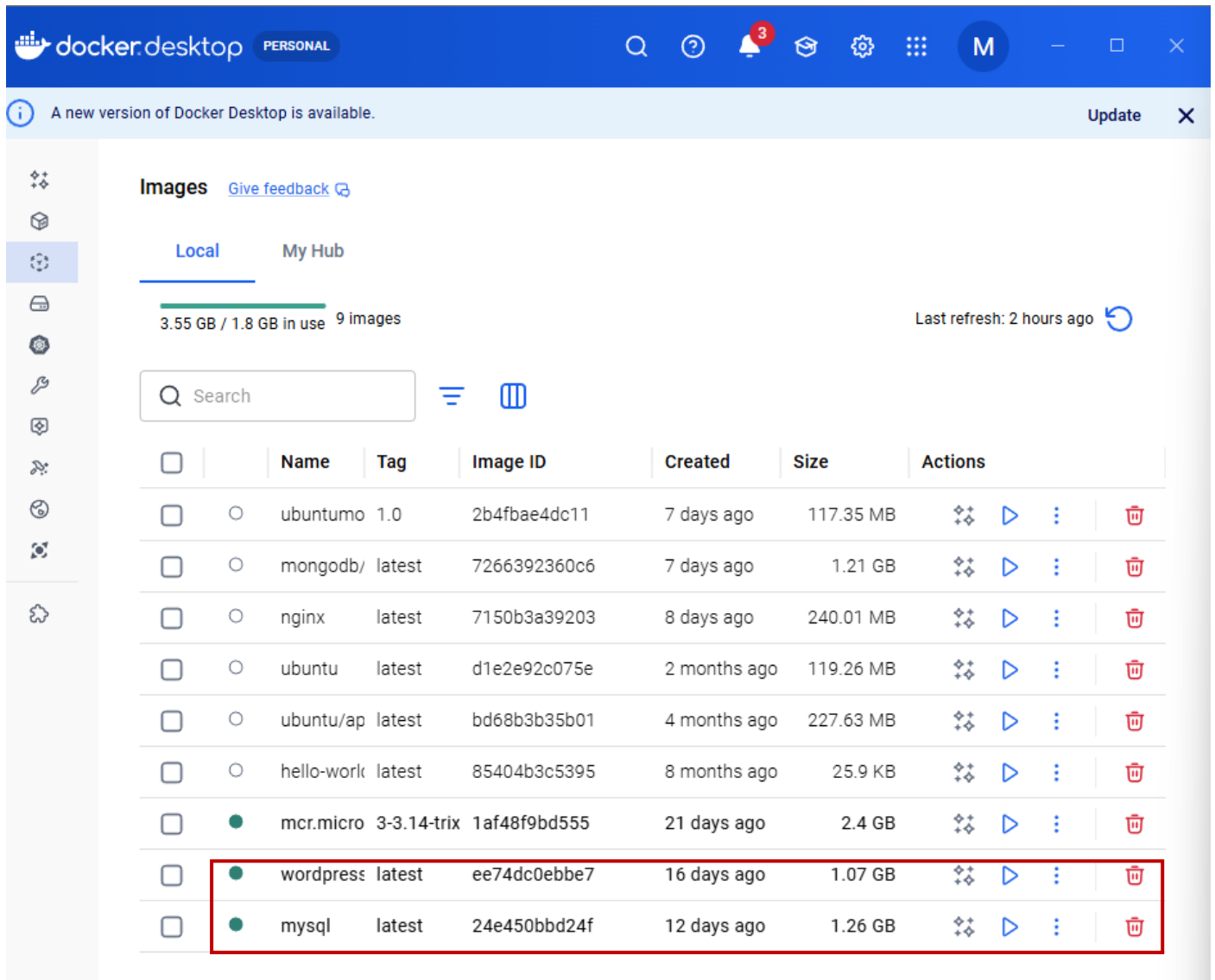
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS powershell + v [ ] [ ] ... | [ ] [ ] X  
PS C:\Users\Administrateur\Documents\Projets_B1\Compose> docker compose up  
[+] up 25/39  
- Image wordpress:latest [███████ ████ ████████████████████] Pulling 4.0s  
- Image mysql:latest [███████ ████████████████████] 58.6MB / 281.9MB Pulling 4.0s  
[ ]
```

```
PS C:\Users\Administrateur\Documents\Projets_B1\Compose> docker compose up
[+] up 2/2
  ✓ Container compose-db-1      Recreated          0.1s
  ✓ Container compose-wordpress-1 Recreated          1.5s
Attaching to db-1, wordpress-1
db-1 | 2026-04-01 11:50:03+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 9
.6.0-1.el9 started.
db-1 | 2026-04-01 11:50:04+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
db-1 | 2026-04-01 11:50:04+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 9
.6.0-1.el9 started.
wordpress-1 | AH00558: apache2: Could not reliably determine the server's fully qualified
domain name, using 172.18.0.3. Set the 'ServerName' directive globally to suppress this mes
sage
wordpress-1 | AH00558: apache2: Could not reliably determine the server's fully qualified
domain name, using 172.18.0.3. Set the 'ServerName' directive globally to suppress this mes
sage
wordpress-1 | [Wed Apr 01 11:50:04.229276 2026] [mpm_prefork:notice] [pid 1:tid 1] AH0016:
: Apache/2.4.66 (Debian) PHP/8.3.30 configured -- resuming normal operations
wordpress-1 | [Wed Apr 01 11:50:04.229325 2026] [core:notice] [pid 1:tid 1] AH00094: Comma
nd line: 'apache2 -D FOREGROUND'
db-1 | 2026-04-01 11:50:04+00:00 [Note] [Entrypoint]: Initializing database files
db-1 | 2026-04-01T11:50:04.400112Z 0 [System] [MY-015017] [Server] MySQL Server Ini
tialization - start.
db-1 | 2026-04-01T11:50:04.402053Z 0 [System] [MY-013169] [Server] /usr/sbin/mysqld
(mysql 9.6.0) initializing of server in progress as process 80
db-1 | 2026-04-01T11:50:04.418257Z 1 [System] [MY-013576] [InnoDB] InnoDB initializ
ation has started.
db-1 | 2026-04-01T11:50:04.784226Z 1 [System] [MY-013577] [InnoDB] InnoDB initializ
ation has ended.
db-1 | 2026-04-01T11:50:05.794955Z 6 [Warning] [MY-010453] [Server] root@localhost
is created with an empty password ! Please consider switching off the --initialize-insecure
option.
db-1 | 2026-04-01T11:50:07.674019Z 0 [System] [MY-015018] [Server] MySQL Server Ini
tialization - end.
db-1 | 2026-04-01 11:50:07+00:00 [Note] [Entrypoint]: Database files initialized
db-1 | 2026-04-01 11:50:07+00:00 [Note] [Entrypoint]: Starting temporary server
db-1 | 2026-04-01T11:50:07.763147Z 0 [System] [MY-015015] [Server] MySQL Server - s
tart.
db-1 | 2026-04-01T11:50:08.074747Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld
(mysql 9.6.0) starting as process 121
db-1 | 2026-04-01T11:50:08.074773Z 0 [System] [MY-015590] [Server] MySQL Server has
access to 16 logical CPUs.
db-1 | 2026-04-01T11:50:08.074789Z 0 [System] [MY-015590] [Server] MySQL Server has
access to 16437669888 bytes of physical memory.
db-1 | 2026-04-01T11:50:08.096817Z 1 [System] [MY-013576] [InnoDB] InnoDB initializ
ation has started.
db-1 | 2026-04-01T11:50:08.440207Z 1 [System] [MY-013577] [InnoDB] InnoDB initializ
ation has ended.
db-1 | 2026-04-01T11:50:08.806718Z 0 [Warning] [MY-010068] [Server] CA certificate
ca.pem is self signed.
db-1 | 2026-04-01T11:50:08.806782Z 0 [System] [MY-013602] [Server] Channel mysql_ma
in configured to support TLS. Encrypted connections are now supported for this channel.
db-1 | 2026-04-01T11:50:08.809987Z 0 [Warning] [MY-011810] [Server] Insecure config
uration for --pid-file: Location '/var/run/mysqld' in the path is accessible to all OS user
s. Consider choosing a different directory.
```

- J'accède au blog en saisissant l'URL `http://localhost:8000` :

## SCREEN A METTTTTTTTTTRGEGERGEGERGEG

- Dans Docker je me rends dans l'onglet Images et je constate l'image wordpress et mysql :



The screenshot shows the Docker Desktop interface. At the top, there's a blue header with the 'docker.desktop' logo and 'PERSONAL' label. Below the header, a notification bar says 'A new version of Docker Desktop is available.' with an 'Update' button. The main area is titled 'Images' and has tabs for 'Local' and 'My Hub'. A progress bar indicates '3.55 GB / 1.8 GB in use' and '9 images'. A search bar and refresh button are present. The image list table is as follows:

	Name	Tag	Image ID	Created	Size	Actions
<input type="checkbox"/>	ubuntu	1.0	2b4fbae4dc11	7 days ago	117.35 MB	
<input type="checkbox"/>	mongodb/	latest	7266392360c6	7 days ago	1.21 GB	
<input type="checkbox"/>	nginx	latest	7150b3a39203	8 days ago	240.01 MB	
<input type="checkbox"/>	ubuntu	latest	d1e2e92c075e	2 months ago	119.26 MB	
<input type="checkbox"/>	ubuntu/ap	latest	bd68b3b35b01	4 months ago	227.63 MB	
<input type="checkbox"/>	hello-work	latest	85404b3c5395	8 months ago	25.9 KB	
<input type="checkbox"/>	mcr.micro	3-3.14-trix	1af48f9bd555	21 days ago	2.4 GB	
<input type="checkbox"/>	wordpress	latest	ee74dc0ebbe7	16 days ago	1.07 GB	
<input type="checkbox"/>	mysql	latest	24e450bbd24f	12 days ago	1.26 GB	

- Je me rends dans l'onglet Containers, je constate la présence du conteneur compose et que les deux images vues précédemment sont sollicitées :

