

Chapitre 4 – Docker sous Windows (Docker Desktop et WSL 2)

Table des Matières :

1. Installation / Activation de WSL.....	2
2. Installation Ubuntu.....	5
3. Installation Docker.....	8
4. Premier test.....	14
5. Deuxième test.....	17
6. Troisième test.....	20
7. Quatrième test.....	25
8. Cinquième test : mode interactif d'utilisation d'un conteneur.....	28
9. Création d'une image à la main avec docker commit.....	31
10. Première approche des volumes.....	33
11. Gestion des volumes en écriture.....	35
11.1. Premier type de volume : volume nommé.....	35
11.2. Deuxième type de volume : montage lié (bind mounting).....	44
12. Utiliser un conteneur Docker et Visual Studio Code.....	47
13. Monter une stack applicative avec Docker Compose.....	58

1. Installation / Activation de WSL.

- Nous procédons à l'installation de WSL

```

Administrateur : Windows PowerShell
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:\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 : 10.0.26200.7840

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.
L'opération demandée est réussie. Les modifications ne seront pas effectives avant que le système ne soit réamorcé.
PS C:\WINDOWS\system32>

```

- Cette commande permet de consulter le catalogue officiel, ainsi de vérifier les noms exacts.

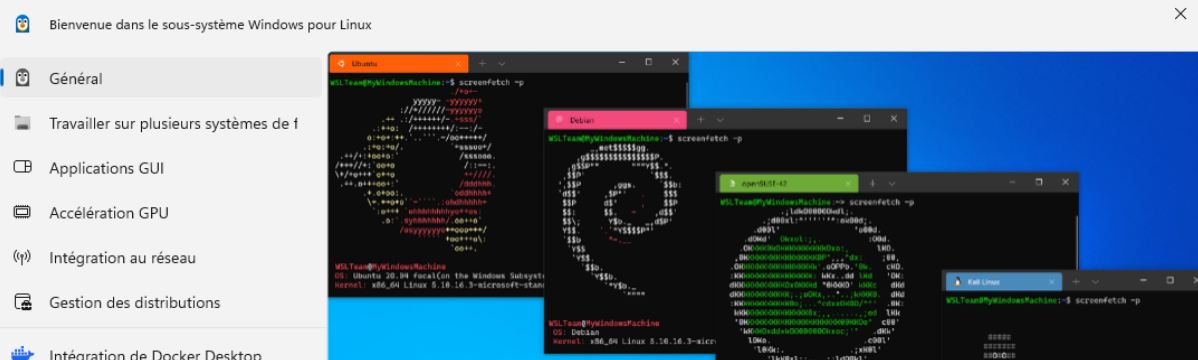
```

Administrateur : Windows PowerShell
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>

```

• Nous procédons par une vérification :



Bienvenue dans WSL

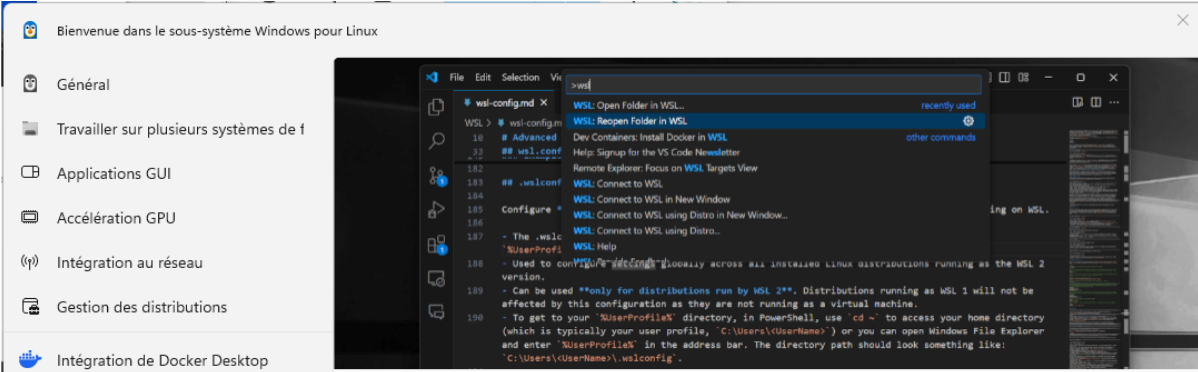
Le sous-système Windows pour Linux (WSL) vous permet d'exécuter vos outils, utilitaires, applications et flux de travail Linux préférés directement sur Windows.

Prenez un moment pour prévisualiser certaines des fonctionnalités préférées de la communauté ou consultez notre documentation complète.

[Documentation du sous-système Windows pour Linux \(WSL\)](#)

[Bonnes pratiques pour la configuration](#)

[Prise en main avec Linux](#)



Intégration de VS Code

Vous pouvez utiliser WSL comme environnement de développement à temps plein directement depuis VS Code.

Comment installer

Après avoir installé VS Code, vous pouvez installer l'extension Remote WSL à partir du terminal Windows :

```
'code --install-extension ms-vscode-remote.remote-wsl'
```

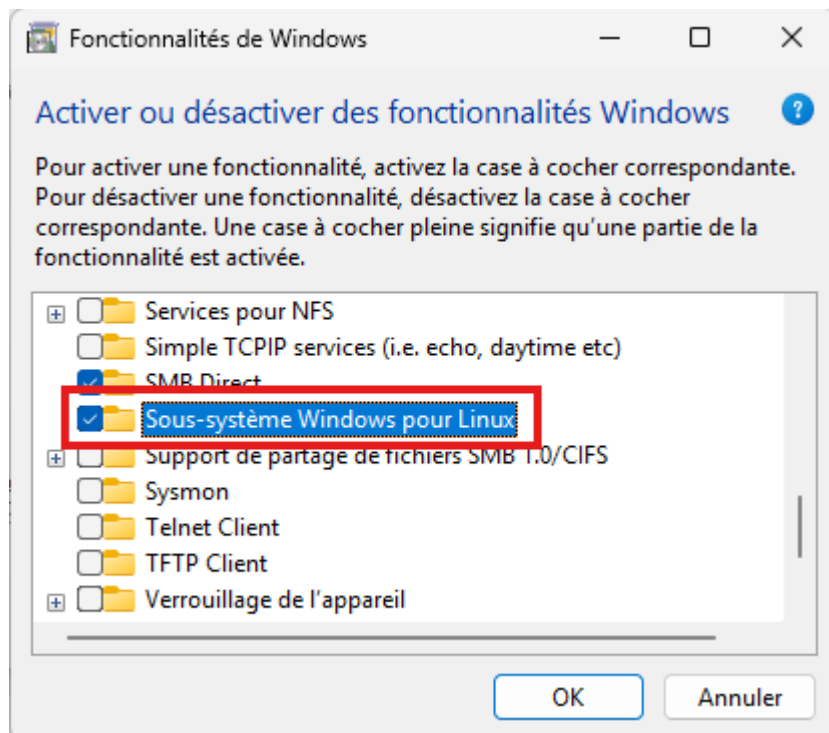
Ouvrir un Projet WSL dans Visual Studio Code

Pour ouvrir un projet dans VS Code à partir de votre distribution WSL, ouvrez la ligne de commande de la distribution

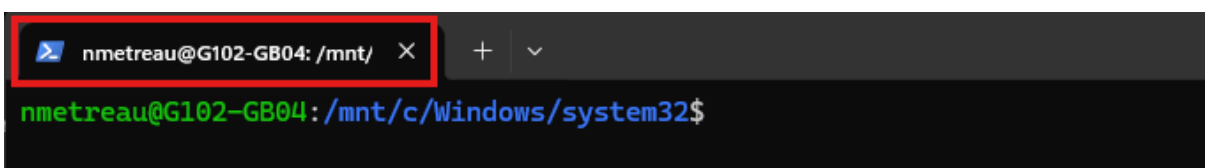
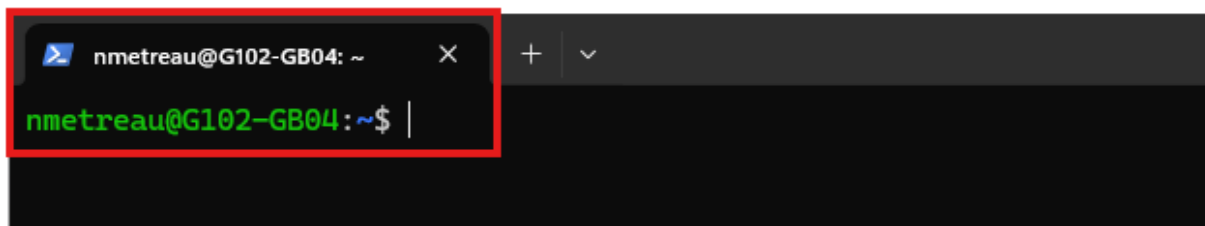
Vous pouvez également accéder à davantage d'options VS Code Remote via la palette de commandes dans VS Code

[En savoir plus sur l'utilisation de WSL avec VS Code](#)

- Nous activons “Sous-système Windows pour Linux” afin de créer une machine virtuelle sur windows

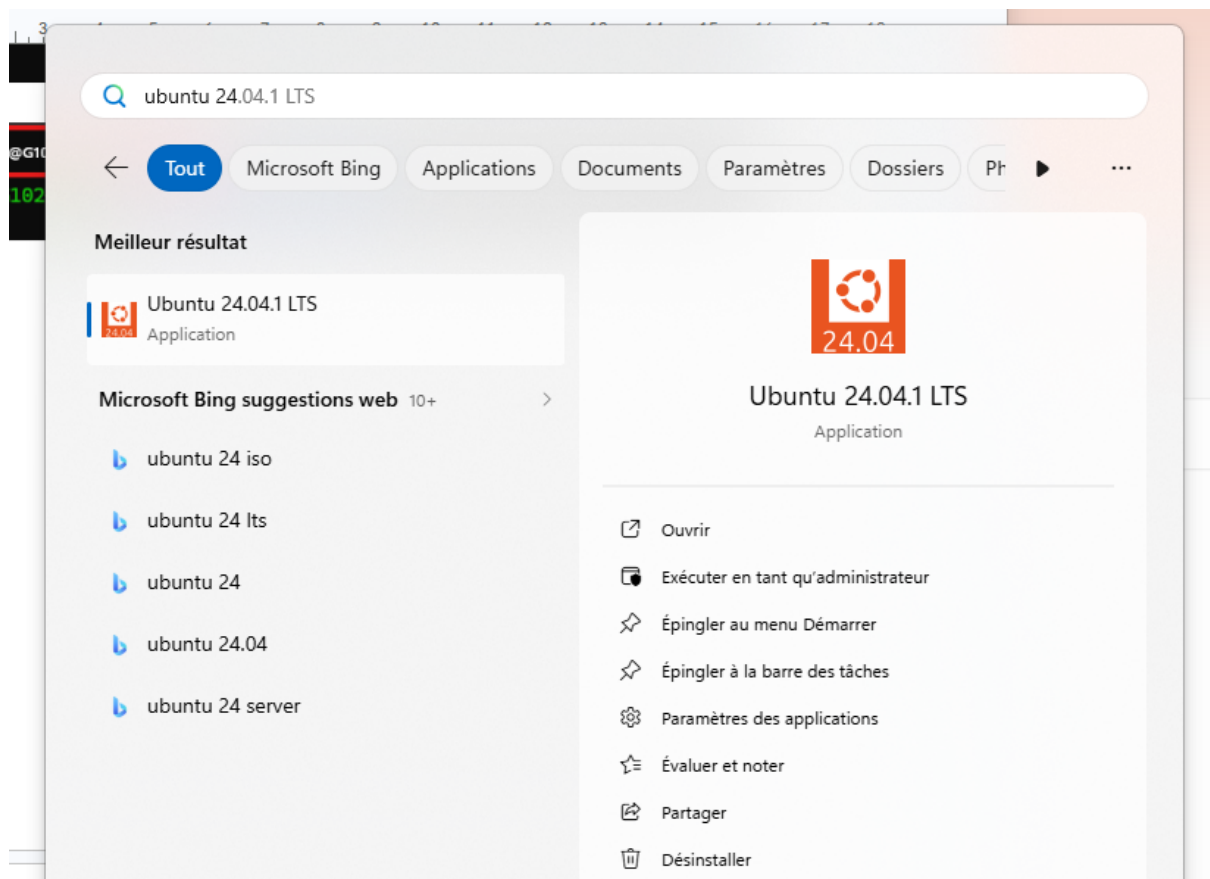
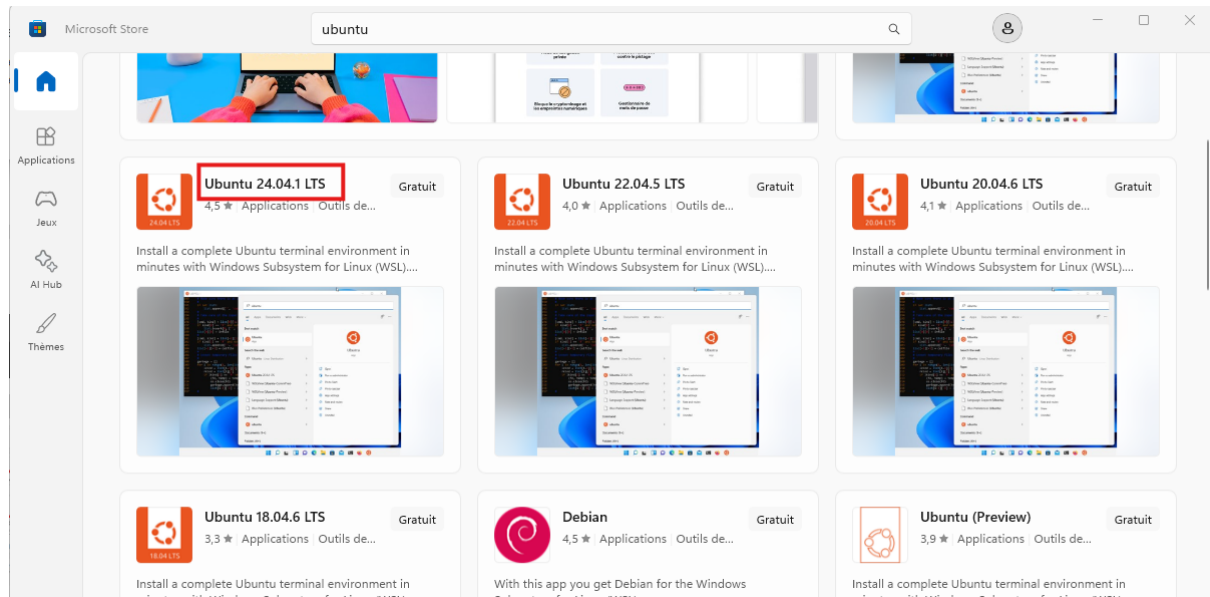


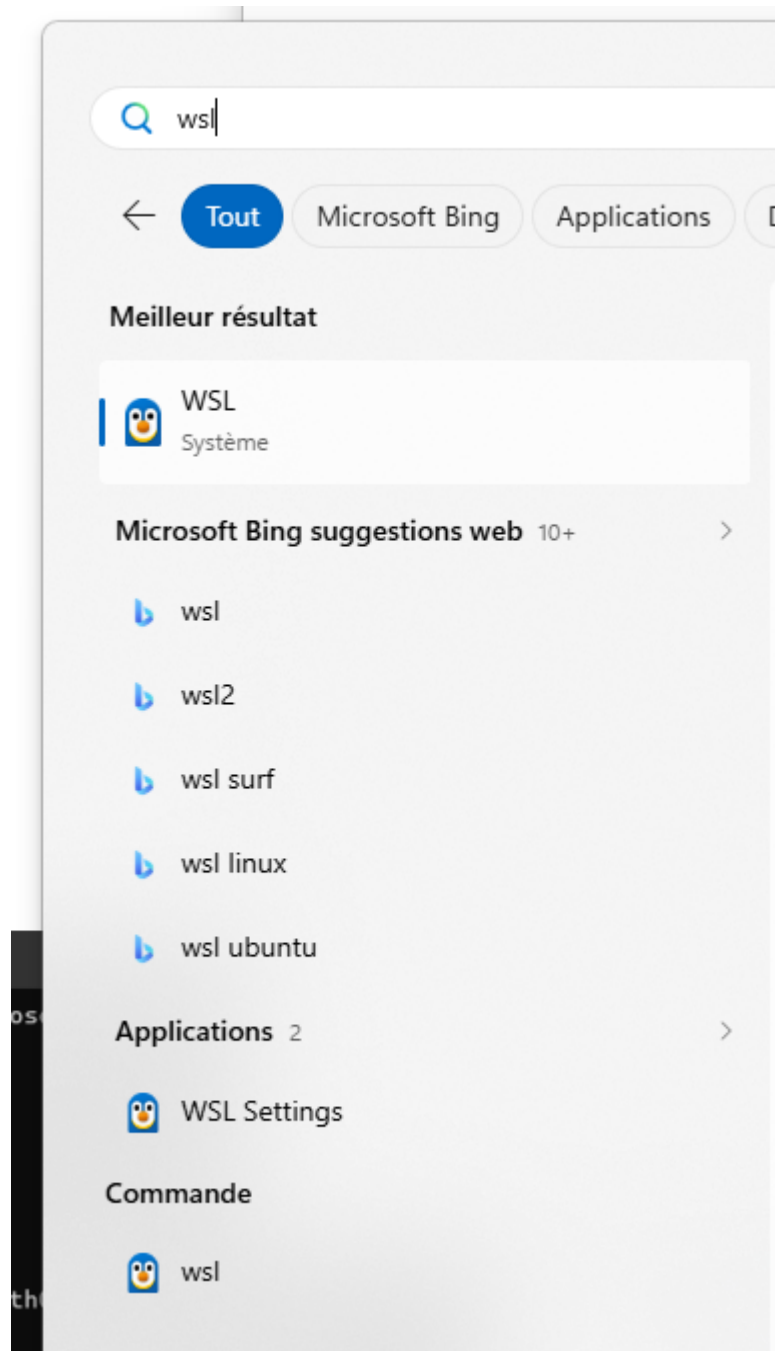
- Nous essayons de nous connecter sur le compte créé sur ubuntu au préalable



2. Installation Ubuntu.

- Nous installons ubuntu 24.04.1 LTS depuis le microsoft store :





```
nmetreau@G102-08: /
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: nmetreau
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.

nmetreau@G102-08:~$ ls
nmetreau@G102-08:~$ pwd
/home/nmetreau
nmetreau@G102-08:~$ cd /
nmetreau@G102-08:/$ ls
bin          boot      etc       init      lib.usr-is-merged  lost+found  mnt  proc  run  sbin.usr-is-merged  srv  tmp  var
bin.usr-is-merged  dev      home     lib       lib64          media      opt  root  sbin  snap              sys  usr
```

- Nous vérifions depuis le terminal windows :

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

Install the latest version for improved performance, bugs, and
features. Learn more at https://aka.ms/WindowsPowerShell

PS C:\Users\nmetreau>
```

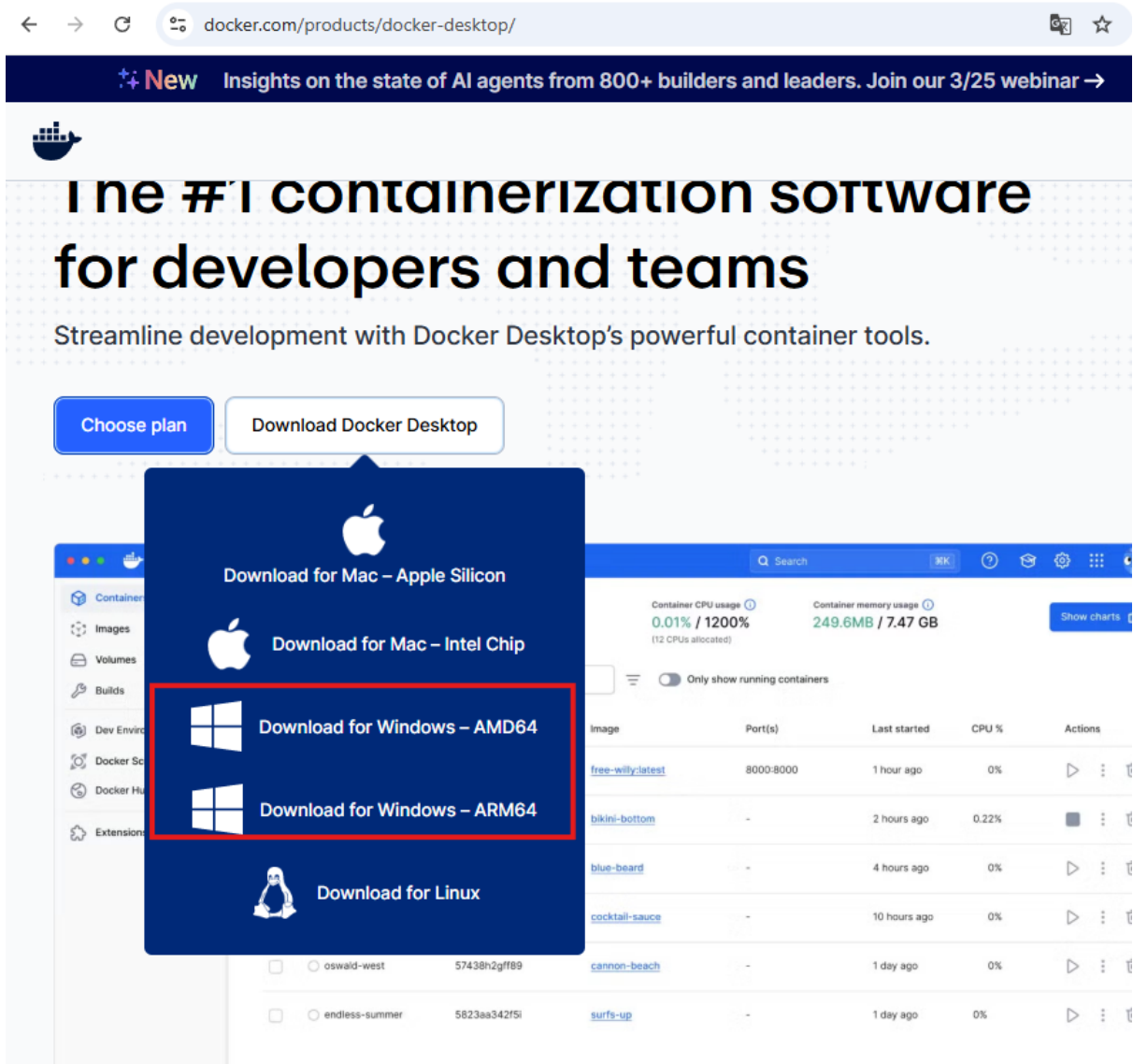


```
Windows PowerShell
PS C:\Users\nmetreau> wsl -d Ubuntu-24.04
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

nmetreau@G102-08:/mnt/c/Users/nmetreau$ exit
logout
PS C:\Users\nmetreau>
```

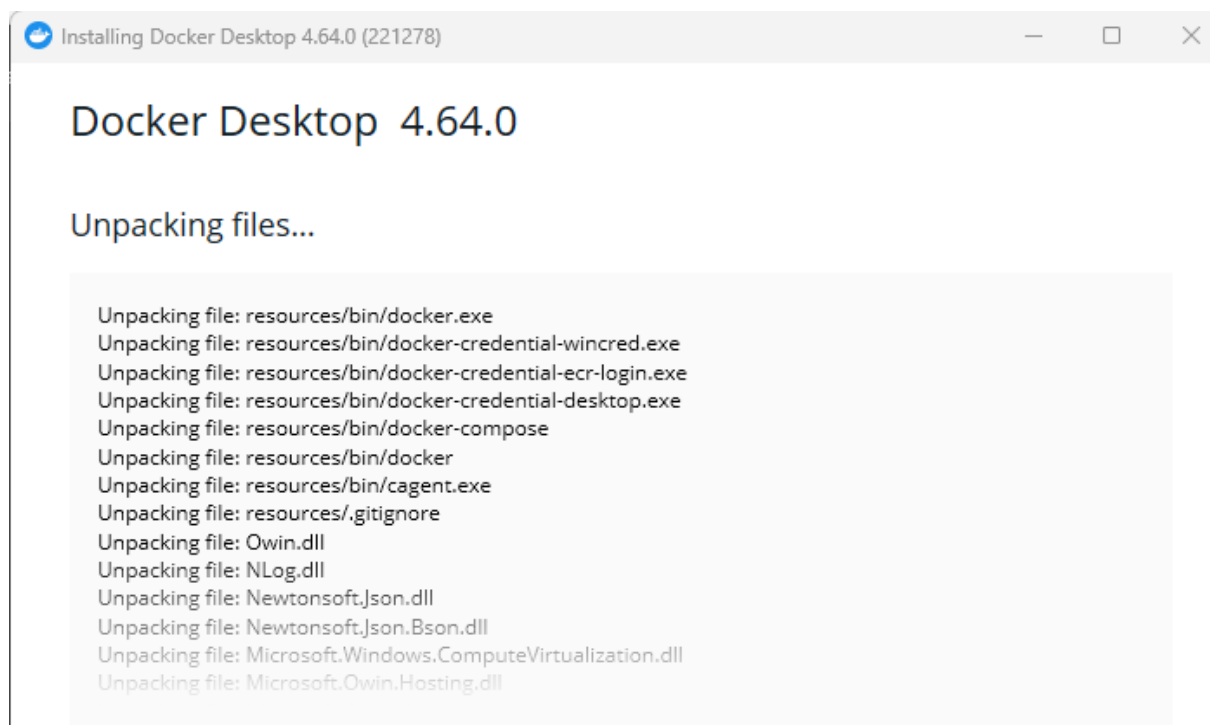
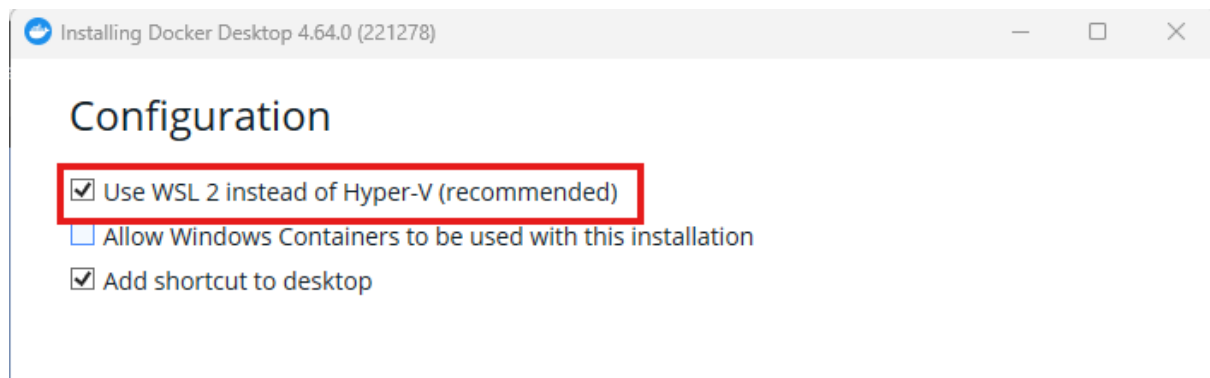
3. Installation Docker.

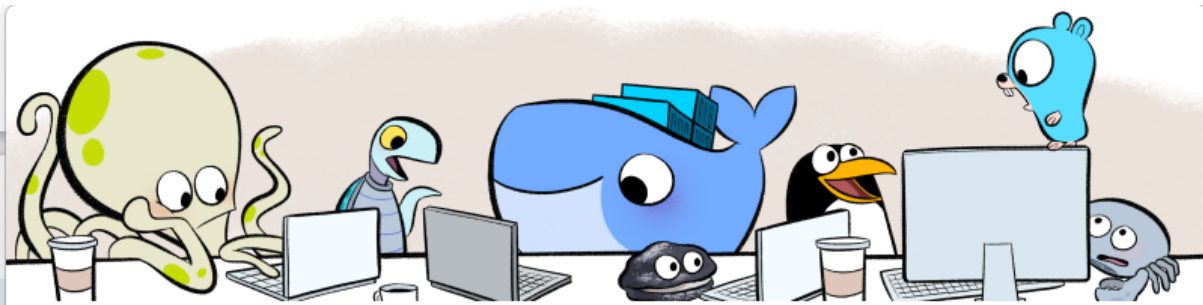
- Nous sélectionnons AMD64



Nom	Modifié le	Type	Taille
Aujourd'hui			
 Docker Desktop Installer	11/03/2026 14:52	Application	609 236 Ko

- Nous cochons Use WSL2... :





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



Create your account

Work

Personal

Work email

metreau.nicolas@gmail.com

Username

aeaeaezz

Password



Send me occasional product updates and announcements.

Sign up

OR

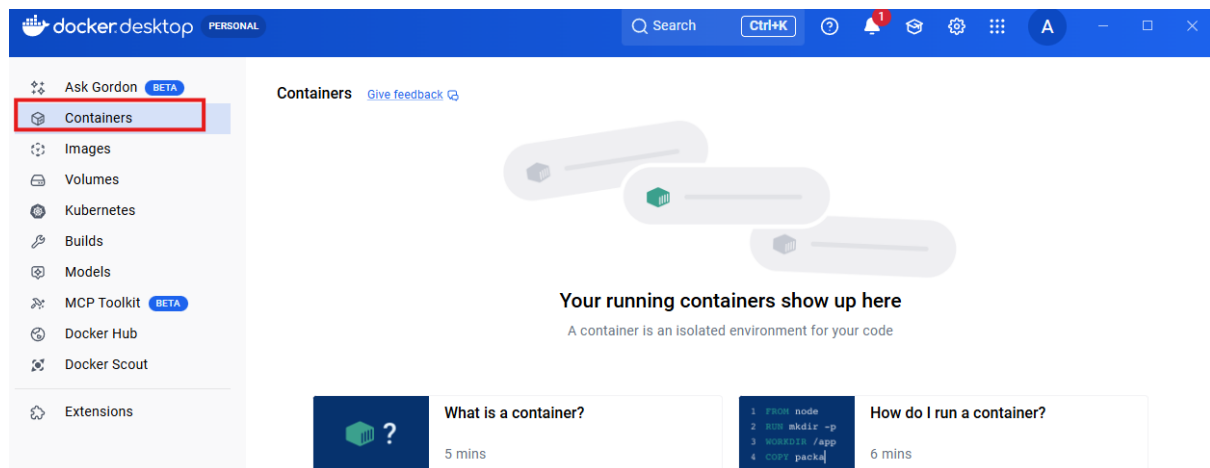


Continue with Google

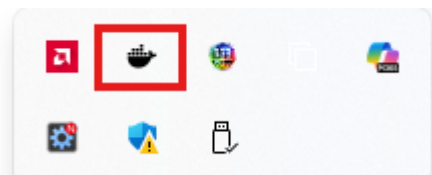
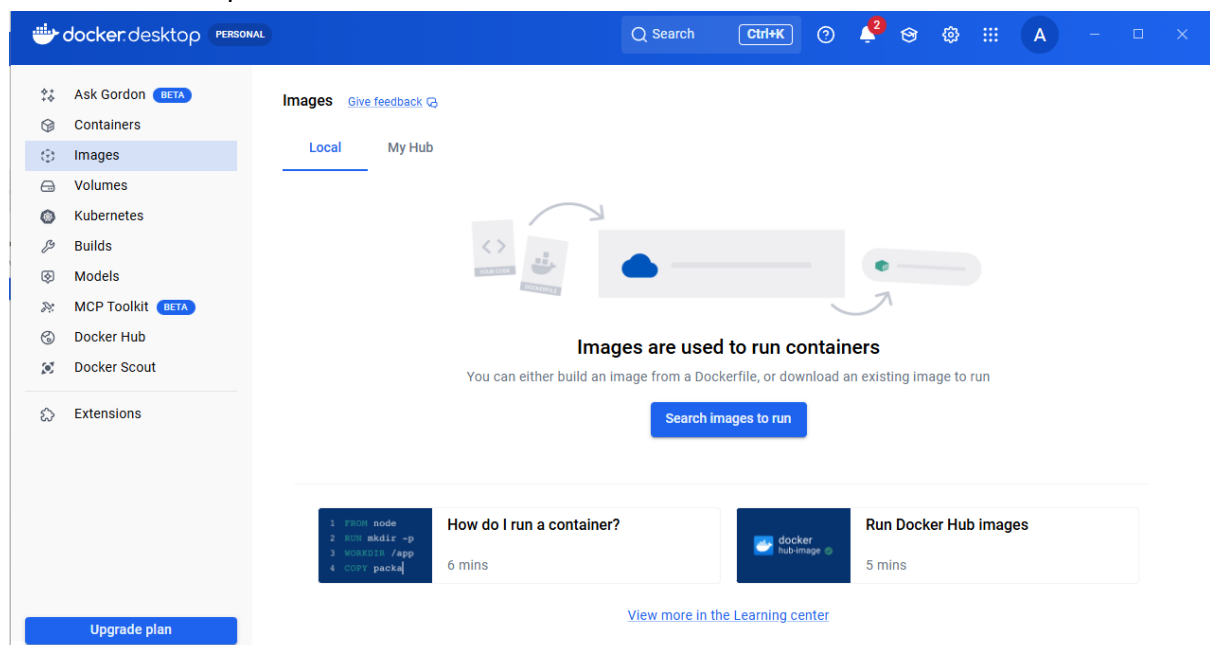


Continue with GitHub

[Already have an account? Sign in](#)



- Nous vérifions que Docker soit bien lancé :



 Docker Desktop is running

 Go to the Dashboard

 aeaeaezz >

Change settings Ctrl+Comma

Troubleshoot

Give feedback

About Docker Desktop

Docker Hub

Documentation

Extensions >

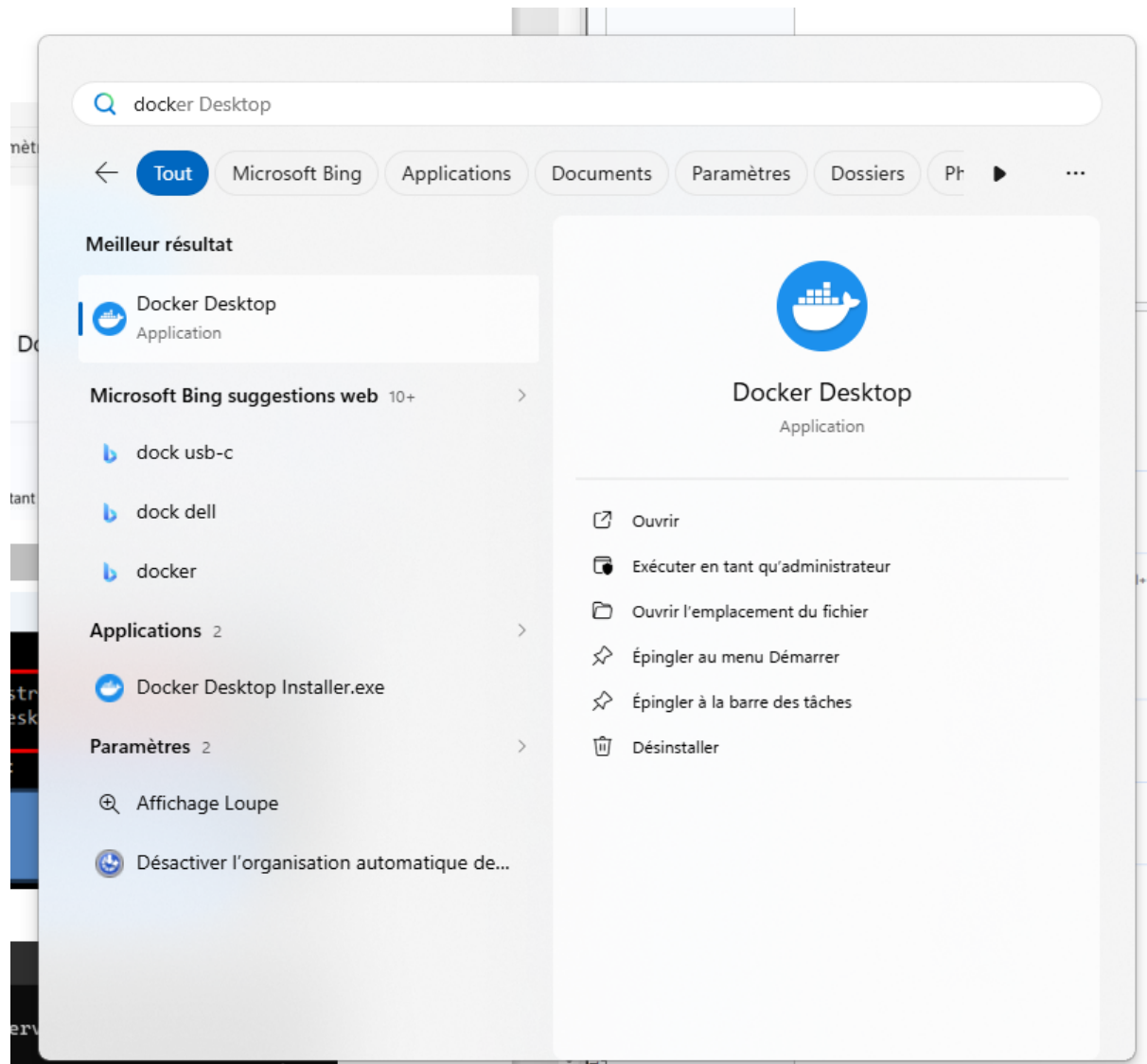
Kubernetes Context >

Check for updates

 Restart

 Pause

 Quit Docker Desktop



4. Premier test.

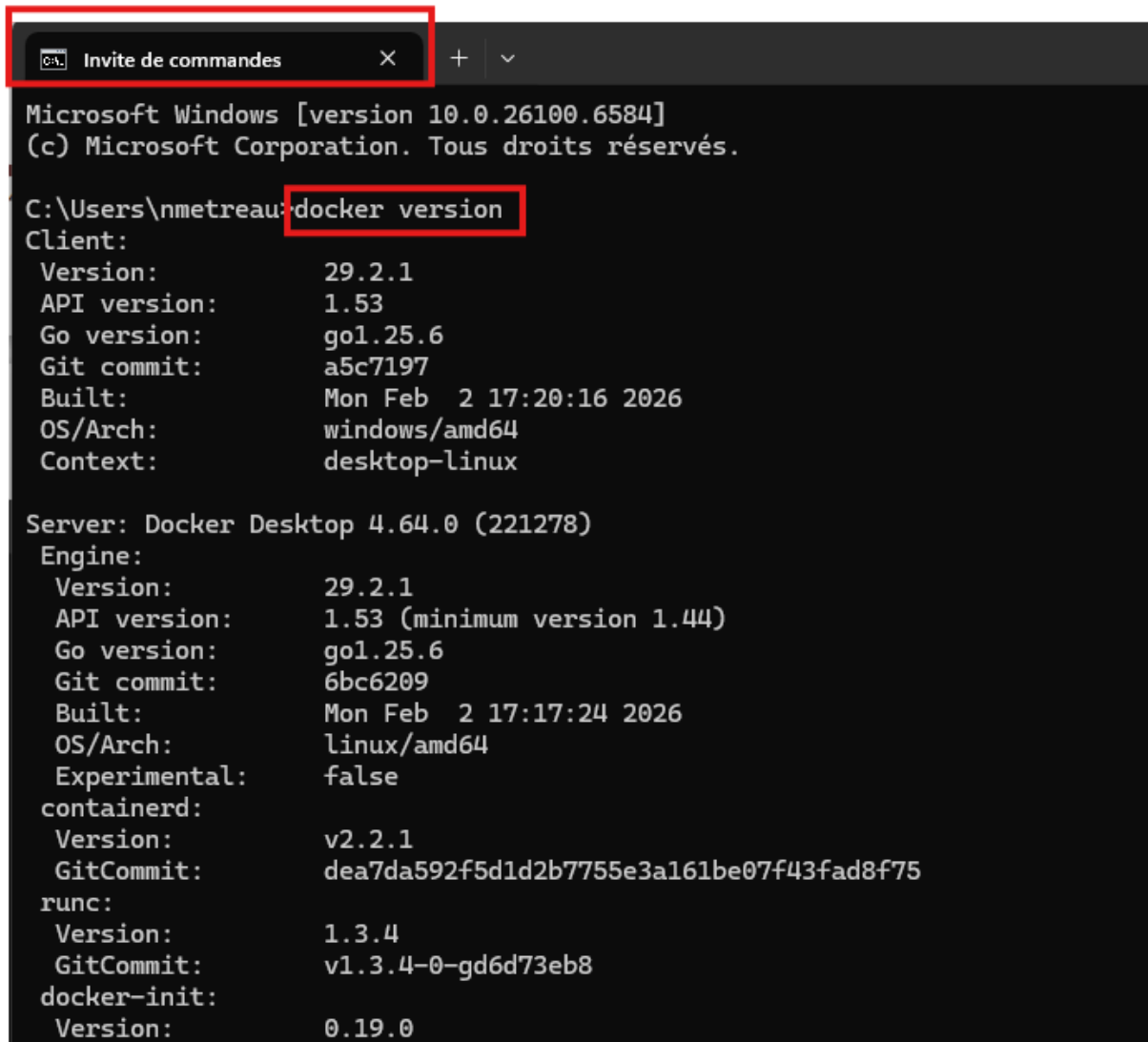
- Nous essayons de vérifier la version de docker sur l'application ubuntu, sauf que l'on doit activer l'intégration dans les paramètres dans docker :

```
nmetreau@G102-08: ~  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
nmetreau@G102-08:~$ 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/  
  
nmetreau@G102-08:~$
```

- Depuis le terminal Windows, nous vérifions la version de docker :

```
Windows PowerShell  
Copyright (C) Microsoft Corporation. Tous droits réservés.  
  
Installez la dernière version de PowerShell pour de nouvelles fonctionnalités  
s  
  
PS C:\Users\nmetreau> docker --version  
Docker version 29.2.1, build a5c7197  
PS C:\Users\nmetreau>
```

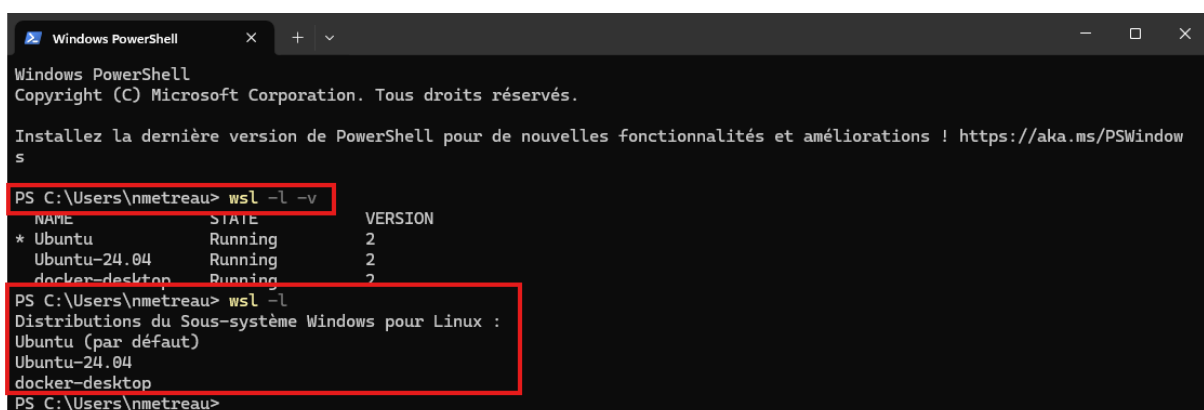
- Nous vérifions, depuis le terminal, la version de docker :



```
Microsoft Windows [version 10.0.26100.6584]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\nmetreau> 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
```



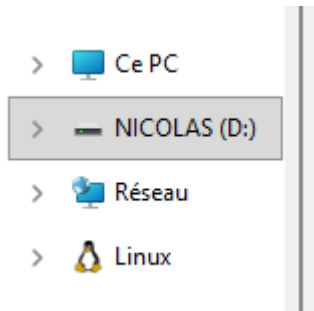
```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

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

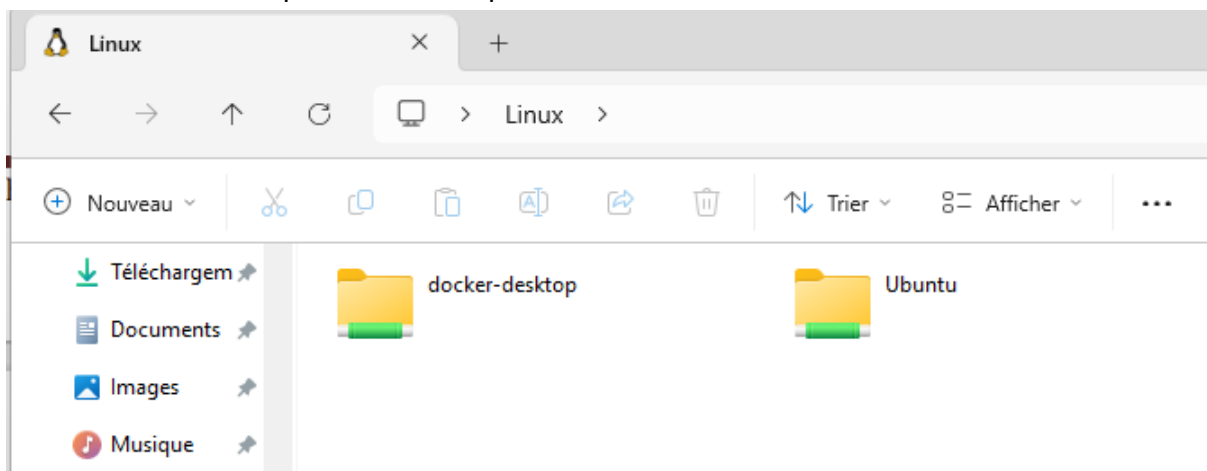
PS C:\Users\nmetreau> wsl -l -v
  NAME                STATE      VERSION
* Ubuntu              Running    2
  Ubuntu-24.04        Running    2
  docker-desktop       Running    2

PS C:\Users\nmetreau> wsl -l
Distributions du Sous-système Windows pour Linux :
Ubuntu (par défaut)
Ubuntu-24.04
docker-desktop
PS C:\Users\nmetreau>
```

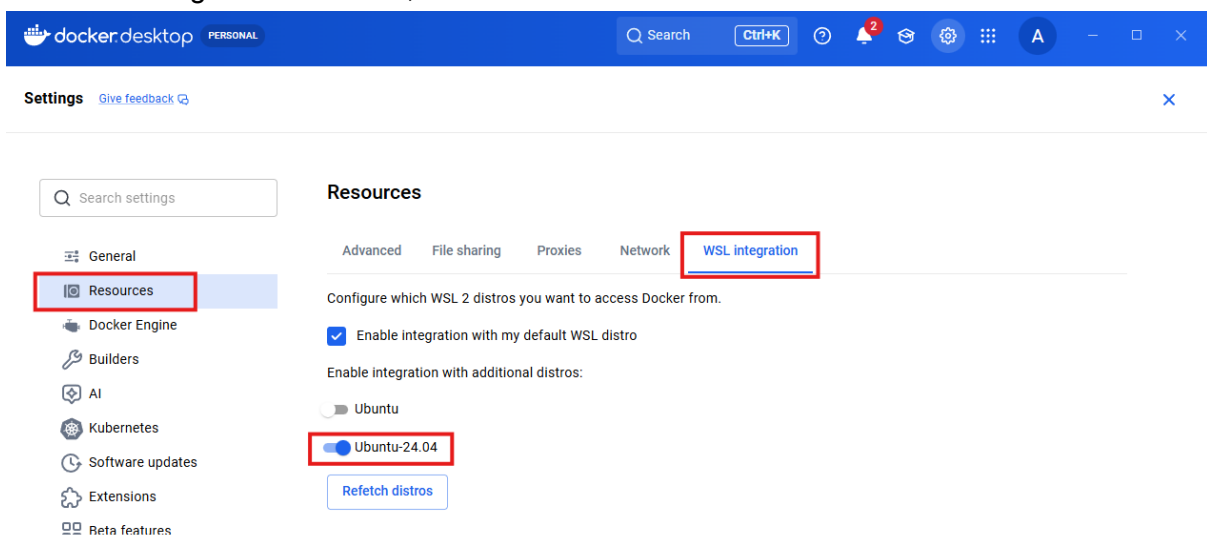
- Nous vérifions la disponibilité de l'environnement virtuel Linux



- Nous vérifions la disponibilité des répertoires :



- Dans les intégrations de WSL, nous cochons la case Ubuntu 24.04 :

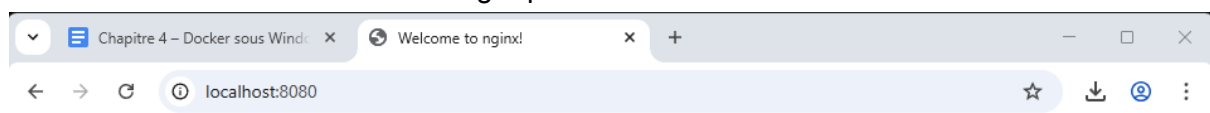


5. Deuxième test.

- Création d'un conteneur ayant pour image nginx.

```
nmetreau@G102-08: ~  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
nmetreau@G102-08:~$ docker run --name nginx -p 8080:80 nginx  
Unable to find image 'nginx:latest' locally  
latest: Pulling from library/nginx  
9eef040df109: Pull complete  
a9d395129dce: Pull complete  
18a071c04bd1: Pull complete  
75a1d70aee50: Pull complete  
df9da45c1db2: Pull complete  
79697674b897: Pull complete  
206356c42440: Pull complete  
d99947bc9177: Download complete  
23abb0f9ce55: 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 14:16:55 [notice] 1#1: using the "epoll" event method  
2026/03/11 14:16:55 [notice] 1#1: nginx/1.29.6  
2026/03/11 14:16:55 [notice] 1#1: built by gcc 14.2.0 (Debian 14.2.0-19)  
2026/03/11 14:16:55 [notice] 1#1: OS: Linux 6.6.87.2-microsoft-standard-WSL2  
2026/03/11 14:16:55 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576  
2026/03/11 14:16:55 [notice] 1#1: start worker processes  
2026/03/11 14:16:55 [notice] 1#1: start worker process 29  
2026/03/11 14:16:55 [notice] 1#1: start worker process 30  
2026/03/11 14:16:55 [notice] 1#1: start worker process 31  
2026/03/11 14:16:55 [notice] 1#1: start worker process 32  
2026/03/11 14:16:55 [notice] 1#1: start worker process 33  
2026/03/11 14:16:55 [notice] 1#1: start worker process 34  
2026/03/11 14:16:55 [notice] 1#1: start worker process 35  
2026/03/11 14:16:55 [notice] 1#1: start worker process 36  
2026/03/11 14:16:55 [notice] 1#1: start worker process 37  
2026/03/11 14:16:55 [notice] 1#1: start worker process 38  
2026/03/11 14:16:55 [notice] 1#1: start worker process 39
```

- L'instance a été créée de cette image qui est le conteneur



Welcome to nginx!

If you see this page, nginx is successfully installed and working. Further configuration is required for the web server, reverse proxy, API gateway, load balancer, content cache, or other features.

For online documentation and support please refer to nginx.org.

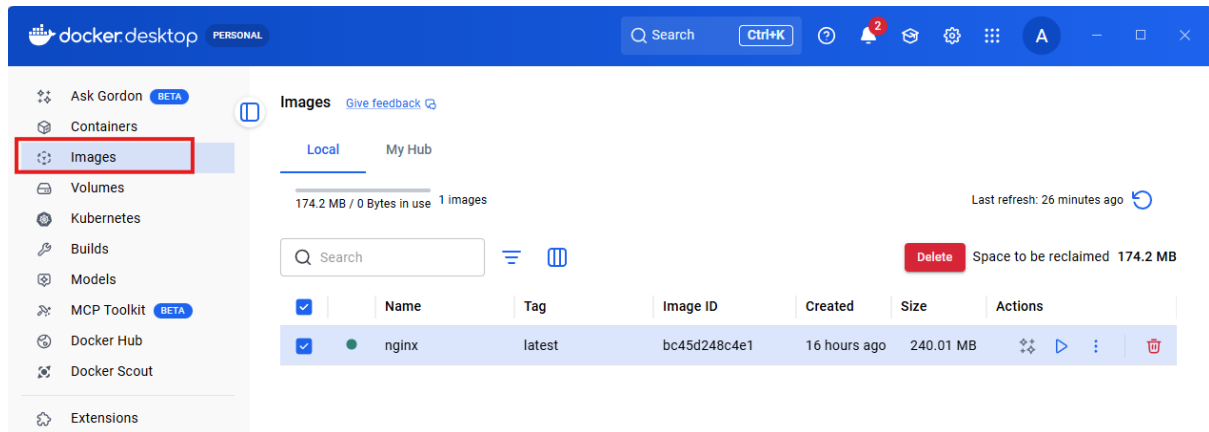
To engage with the community please visit community.nginx.org.

For enterprise grade support, professional services, additional security features and capabilities please refer to f5.com/nginx.

Thank you for using nginx.

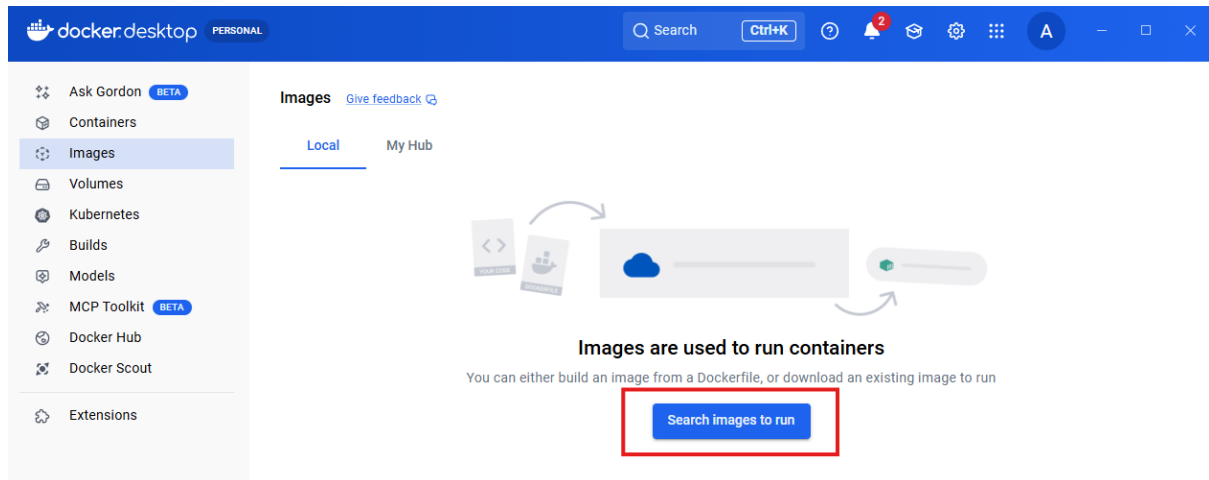
- 100%

- Nous pouvons delete l'image en cliquant sur l'icône de la poubelle :

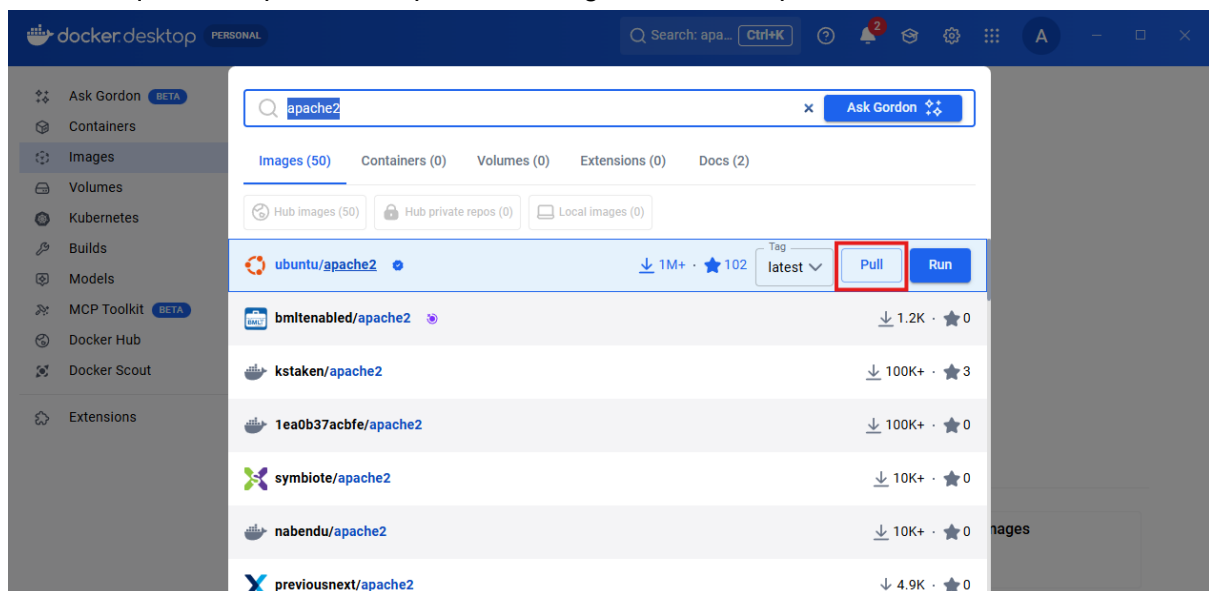


6. Troisième test.

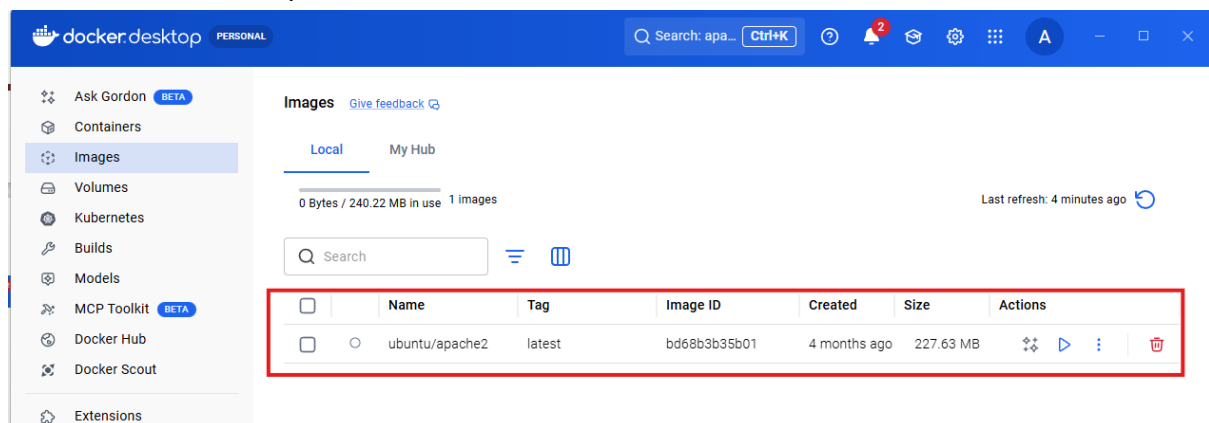
- Nous cliquons sur search images to run :



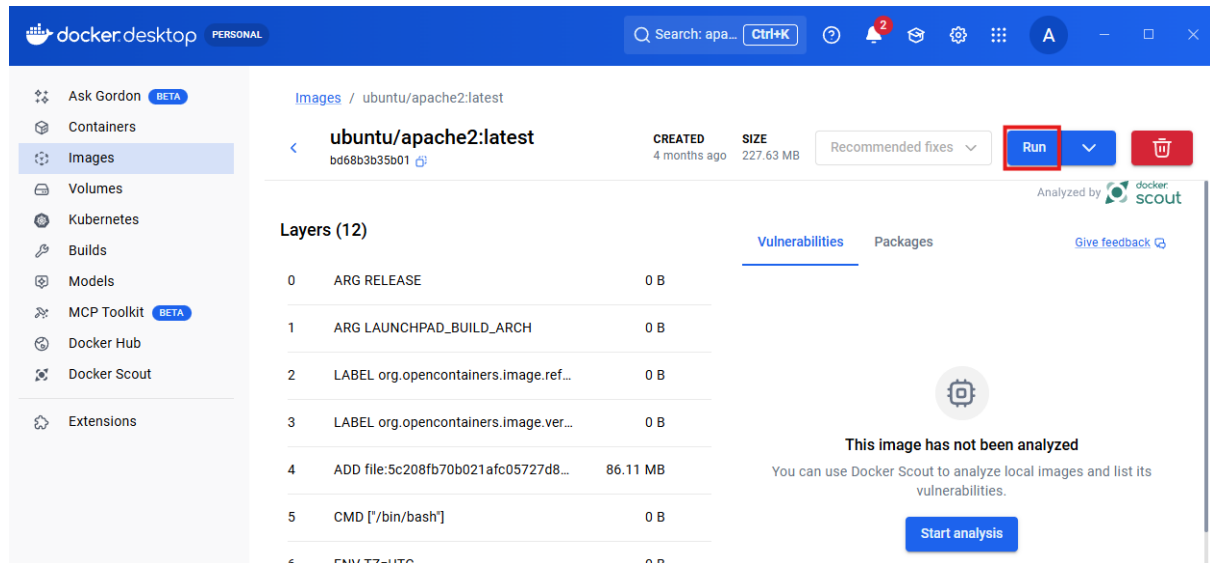
- Nous cliquons sur pull afin de prendre l'image de ubuntu/apache2 :



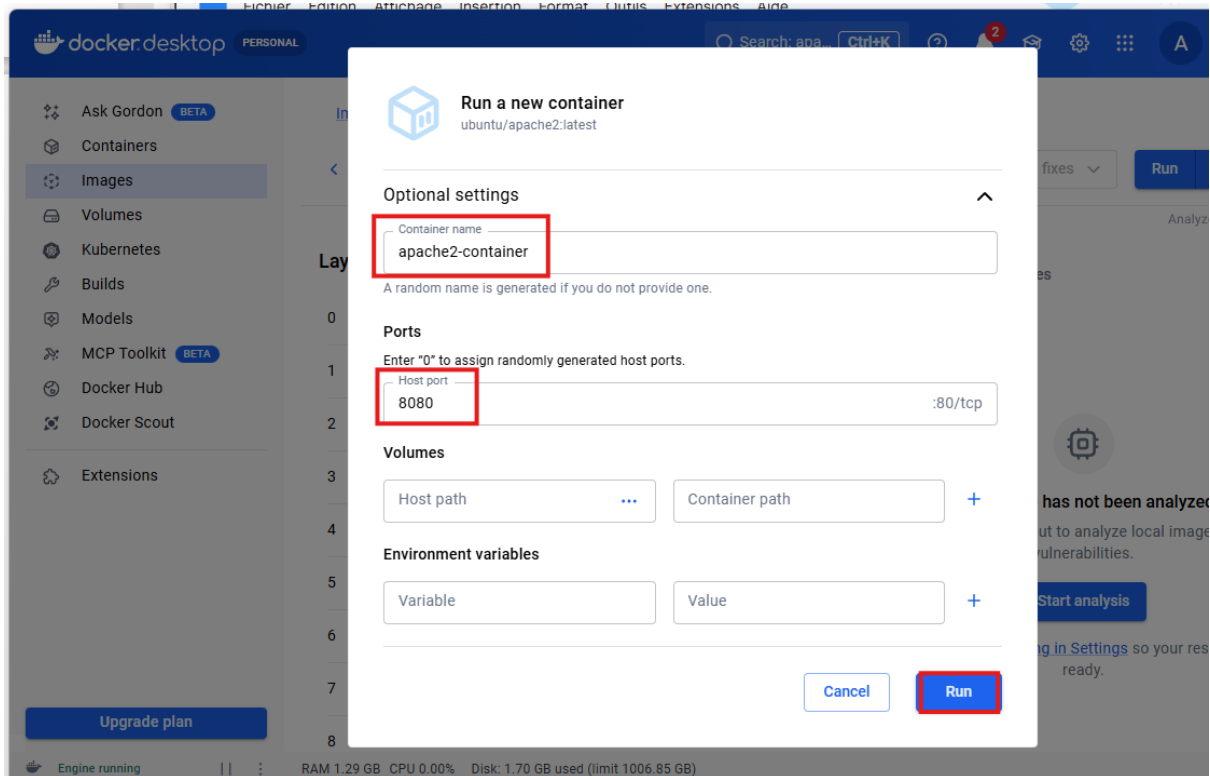
- Nous constatons sa présence :



▪ Nous lançons l'image



▪ Nous constatons son nom ainsi que son port d'écoute :



- Nous vérifions les logs dans l'onglet conteneur :

The screenshot shows the Docker Desktop interface. On the left sidebar, the 'Containers' tab is selected. The main panel displays the 'apache2-container' details. The 'Logs' tab is selected, showing the following log output:

```
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 14:33:20.076670 2026] [mpm_event:notice] [pid 23:tid 23] AH00489: Apache/2.4.63 (Ubuntu) configured -- resuming normal operations
[Wed Mar 11 14:33:20.077254 2026] [core:notice] [pid 23:tid 23] AH00094: Command line: '/usr/sbin/apache2 -D FOREGROUND'
```

The screenshot shows the Docker Desktop interface with the 'Exec' tab selected for the 'apache2-container'. The terminal output shows the following commands and results:

```
# 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
#
```

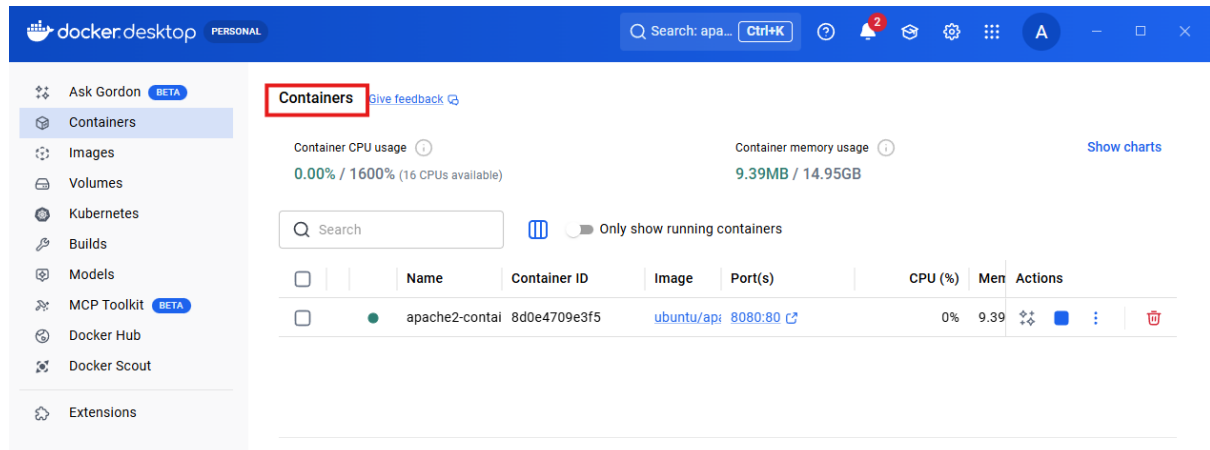
The screenshot shows the Docker Desktop interface with the 'Stats' tab selected for the 'apache2-container'. The status is 'Running (2 minutes ago)'. The CPU usage is 0.01%. The memory usage is 9.39MB / 15.31GB.

CPU usage: 0.01%

Time	CPU
15:33	0.008%
15:33	0.004%
15:33	0.004%
15:34	0.008%
15:34	0.004%
15:34	0.004%
15:34	0.008%
15:34	0.008%
15:35	0.008%
15:35	0.008%
15:35	0.008%
15:35	0.008%
15:35	0.008%

Memory usage: 9.39MB / 15.31GB

Time	Memory
15:33	0B
15:33	0B
15:33	0B
15:33	0B
15:33	0B
15:33	0B
15:33	0B
15:33	0B
15:33	0B
15:33	0B
15:33	0B



- Nous arrêtons et supprimons le container puis l'image.

No containers are running.

No containers are running.

Q Search

Only show running containers

Delete

☒		Name	Container ID	Image	Port(s)	CPU (%)	Memory usag	Actions
☒	☐	apache2-contai	8d0e4709e3f5	ubuntu/apache2:latest	8080:80	N/A	N/A	

Images

Volumes

Kubernetes

Builds

Models

MCP Toolkit BETA

Local

My Hub

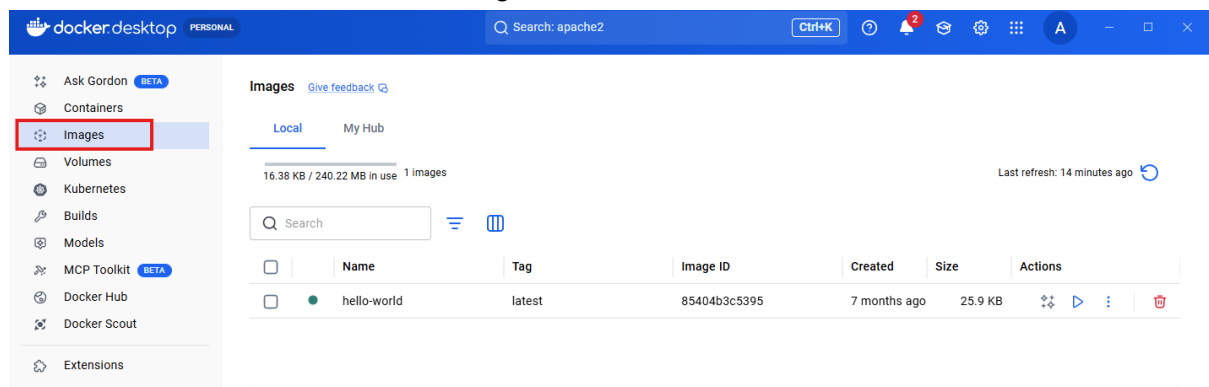
Docker Win - Page 24

7. Quatrième test.

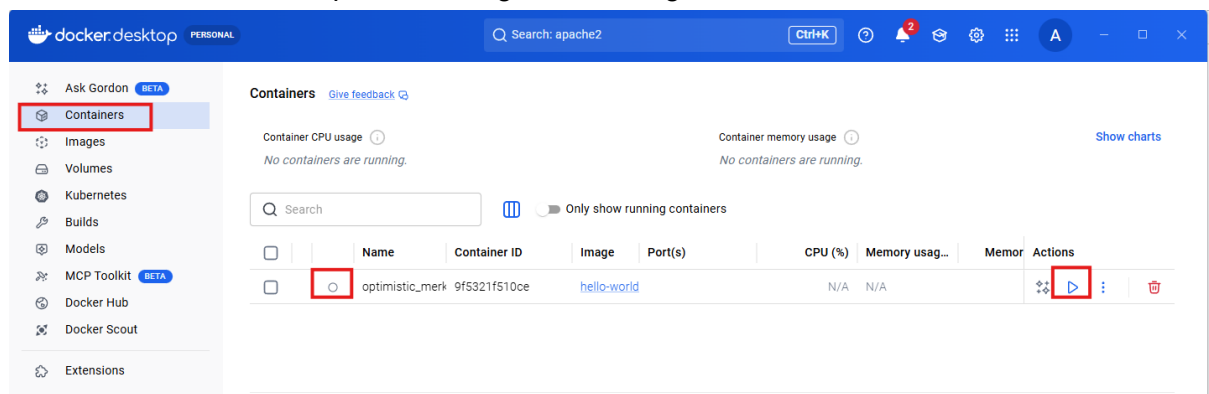
- Nous effectuons un docker run hello-world afin de chercher l'image hello-world en local :

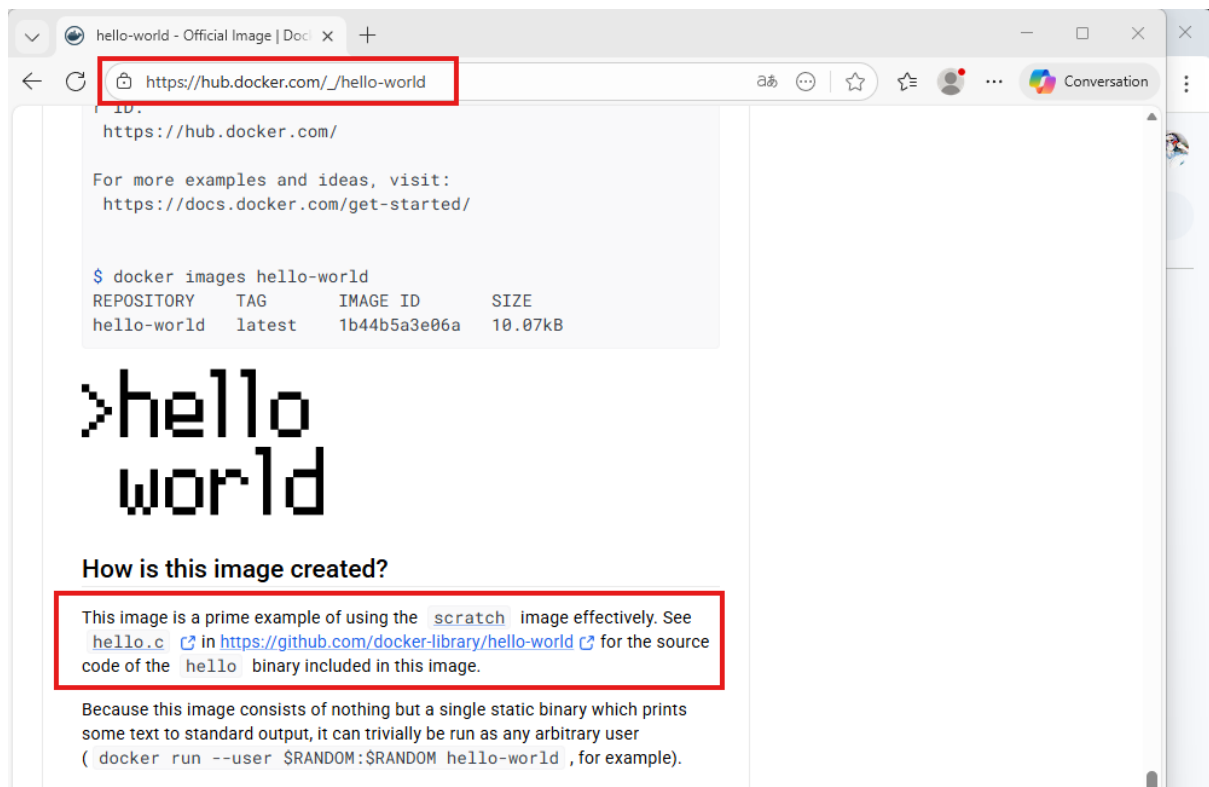
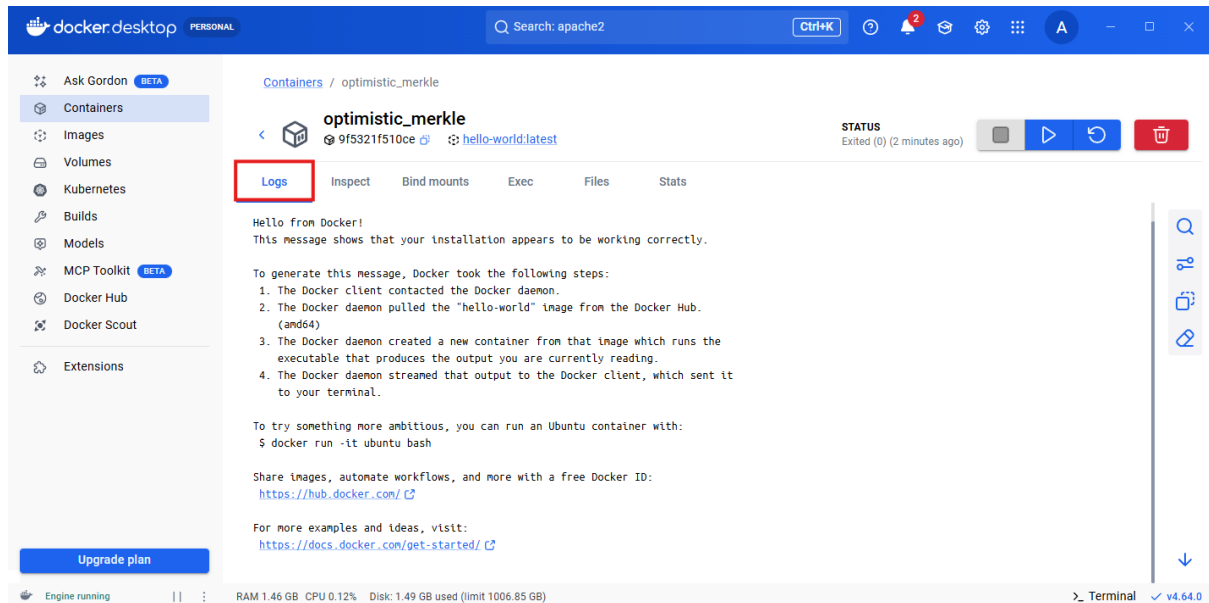
```
nmetreau@G102-08: ~  
nmetreau@G102-08:~$ docker run hello-world  
Unable to find image 'hello-world:latest' locally  
latest: Pulling from library/hello-world  
17eec7bbc9d7: Pull complete  
ea52d200f90: Download complete  
Digest: sha256:85404b3c53951c3ff5d40de0972b1bb21fafa2e8daa23535baf44f33db9dbdd  
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/  
nmetreau@G102-08:~$
```

- Nous constatons la création de l'image

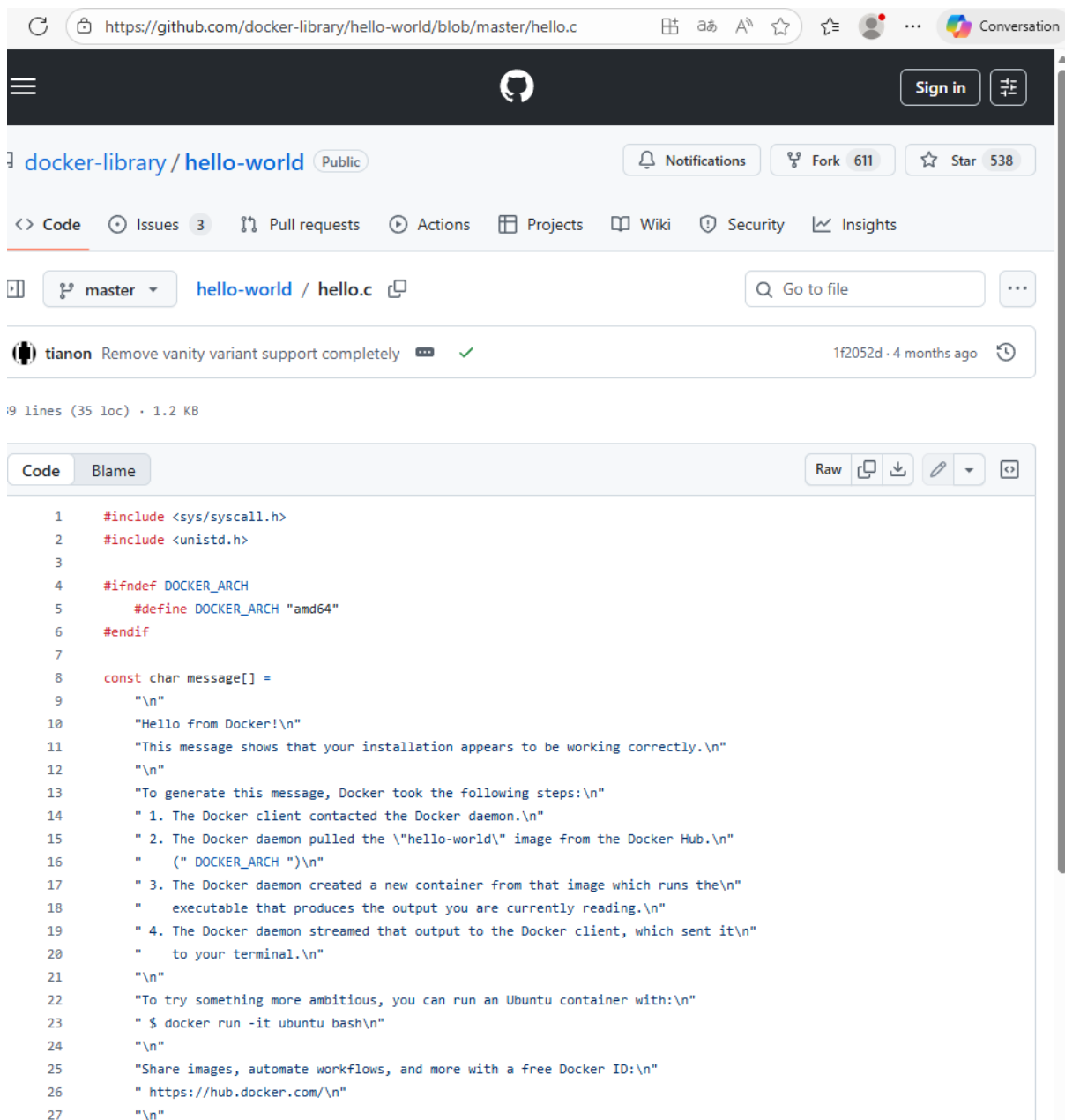
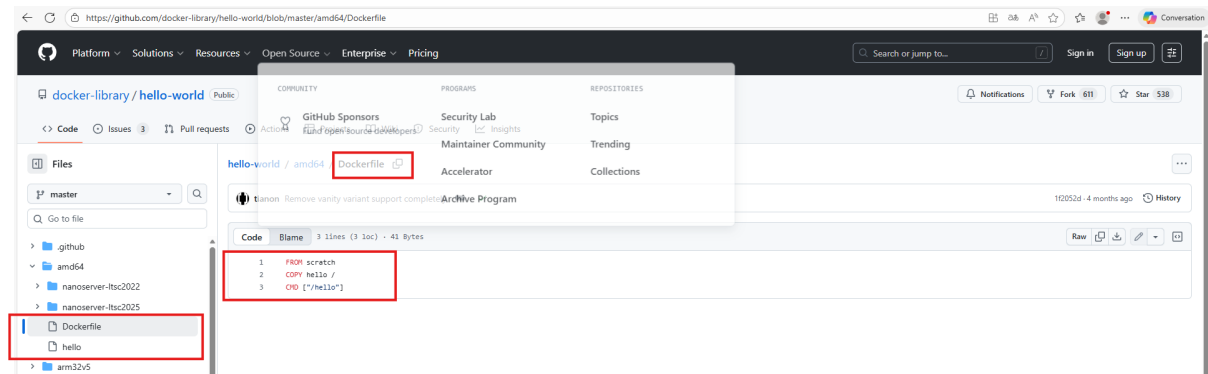


Le conteneur est arrêté après l'affichage du message :



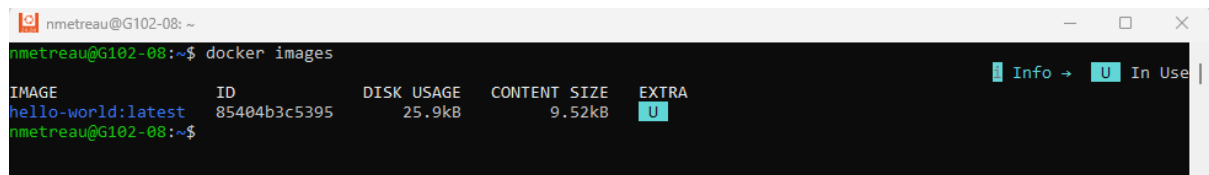


Les fichiers qui servent à créer l'image hello-world :



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

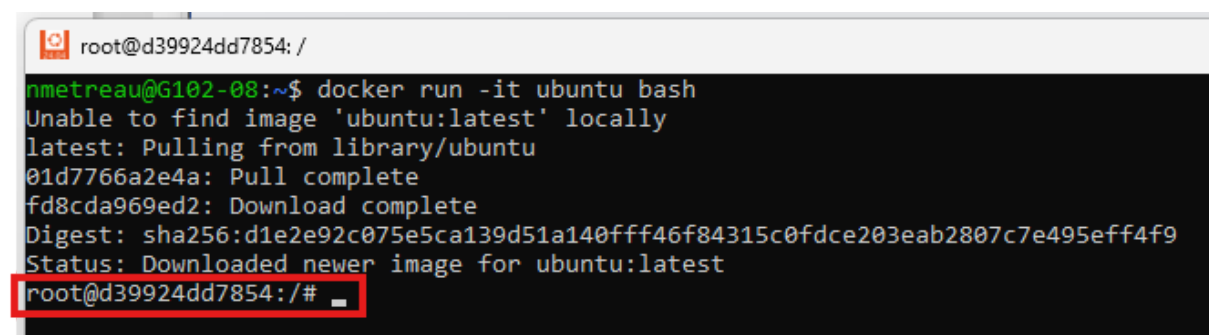
- Commande docker images : liste toutes les images sur le cache local de la machine (identifiant = code hexadécimal)



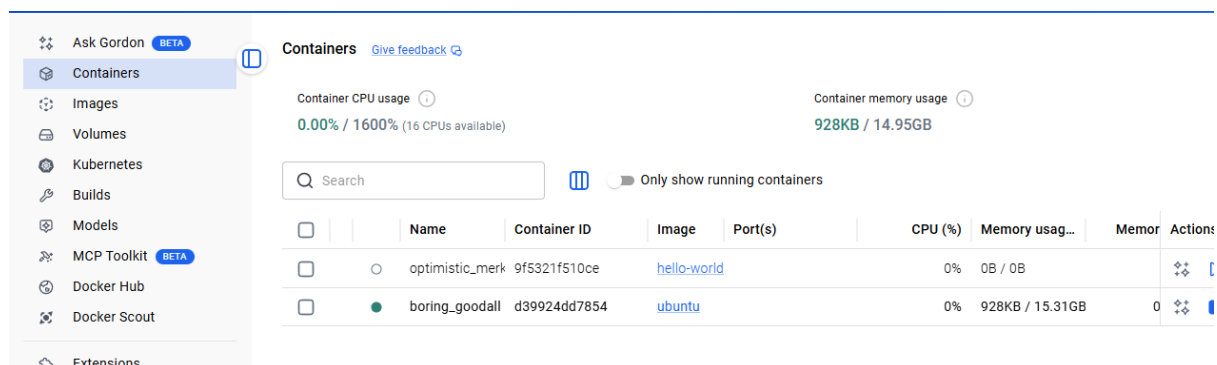
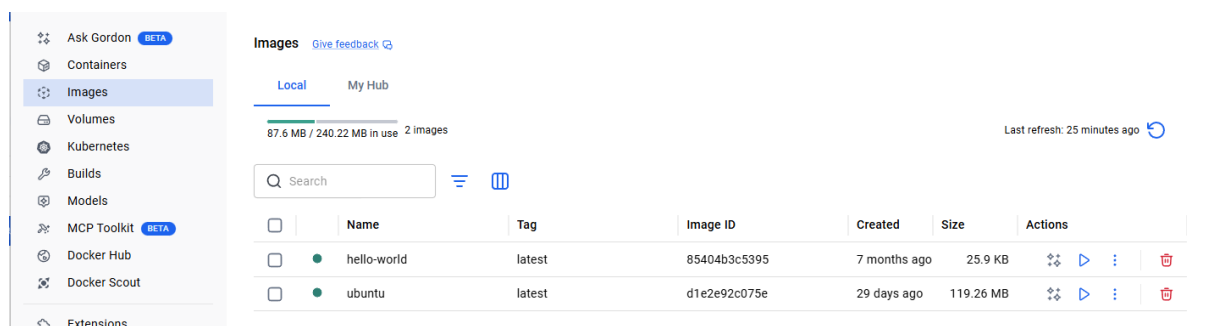
```
nmetreau@G102-08: ~$ docker images
```

IMAGE	ID	DISK USAGE	CONTENT SIZE	EXTRA
hello-world:latest	85404b3c5395	25.9kB	9.52kB	U

- Lancement d'un conteneur en mode interactif qui lancera un shell bash qui interprètera les commandes :



```
nmetreau@G102-08:~$ 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@d39924dd7854: /#
```



- La commande `docker ps` permet de lister les conteneurs en cours d'exécution :

```
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

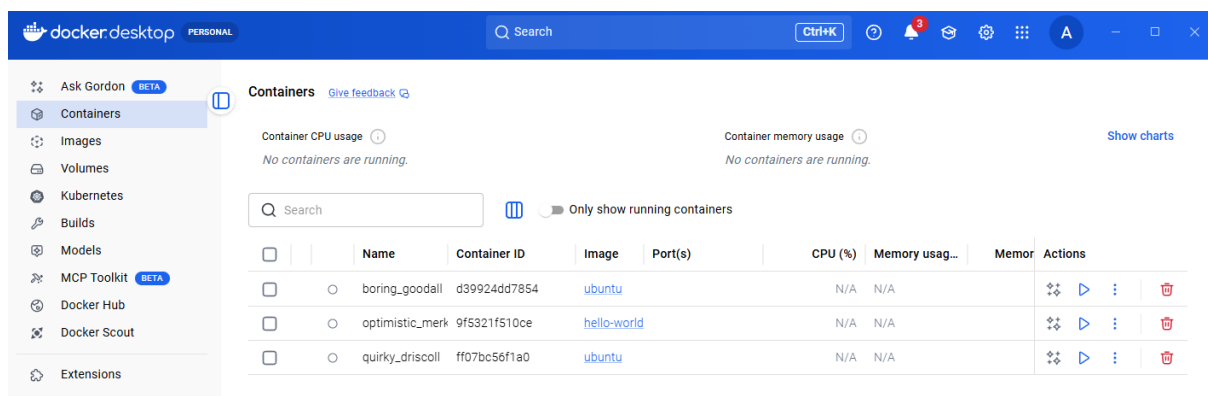
PS C:\Users\nmetreau> docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
d39924dd7854   ubuntu   "bash"    About a minute ago   Up About a minute   boring_goodall
PS C:\Users\nmetreau>
```

- Nous tapons les commandes `pwd` ainsi que `ls` et `top`

```
root@d39924dd7854:/# pwd
/
root@d39924dd7854:/# ls
bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
root@d39924dd7854:/# top
top - 14:53:32 up 54 min, 0 user, load average: 0.04, 0.03, 0.01
Tasks: 2 total, 1 running, 1 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 15674.2 total, 13877.3 free, 1197.3 used, 813.6 buff/cache
MiB Swap: 4096.0 total, 4096.0 free, 0.0 used, 14476.9 avail Mem

Unknown command - try 'h' for help
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
1 root 20 0 4588 3840 3456 S 0.0 0.0 0:00.02 bash
11 root 20 0 8864 5120 3072 R 0.0 0.0 0:00.00 top

root@ff07bc56f1a0:/# rm -fr /home
root@ff07bc56f1a0:/# ls
bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
root@ff07bc56f1a0:/# exit
exit
nmetreau@G102-08:~$
```



```
Administrateur : Windows PowerShell

PS C:\WINDOWS\system32>
PS C:\WINDOWS\system32> docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
PS C:\WINDOWS\system32>
```

- Nous affichons les conteneurs

```
PS C:\WINDOWS\system32> docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
ff07bc56f1a0   ubuntu        "bash"                  6 minutes ago   Exited (0)    4 minutes ago   quirky_driscoll
d39924dd7854   ubuntu        "bash"                  12 days ago    Exited (255)  7 minutes ago   boring_goodall
9f5321f510ce   hello-world   "/hello"                12 days ago    Exited (0)    12 days ago    optimistic_merkle
PS C:\WINDOWS\system32>
```

- Nous effectuons “start” du conteneur commençant par “ff07”

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ docker start ff07
ff07
nmetreau@G102-08:~$ docker stop ff07
ff07
nmetreau@G102-08:~$ dcoker start -ai ff07
dcoker: command not found
nmetreau@G102-08:~$ docker start -ai ff07
root@ff07bc56f1a0:/# ls
bin boot dev etc lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
root@ff07bc56f1a0:/# exit
exit
nmetreau@G102-08:~$
```

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ docker start ff07
ff07
nmetreau@G102-08:~$ docker exec -it ff07 bash
root@ff07bc56f1a0:/# exit
exit
nmetreau@G102-08:~$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
ff07bc56f1a0   ubuntu        "bash"                  13 minutes ago   Up 2 minutes                quirky_driscoll
nmetreau@G102-08:~$ docker stop ff07
ff07
nmetreau@G102-08:~$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
nmetreau@G102-08:~$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
ff07bc56f1a0   ubuntu        "bash"                  14 minutes ago   Exited (137)  15 seconds ago   quirky_driscoll
d39924dd7854   ubuntu        "bash"                  12 days ago     Exited (137)  5 minutes ago    boring_goodall
9f5321f510ce   hello-world   "/hello"                12 days ago     Exited (0)    6 minutes ago    optimistic_merkle
nmetreau@G102-08:~$
```

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ docker start ff07
ff07
nmetreau@G102-08:~$ docker exec -it ff07 bash
root@ff07bc56f1a0:/# exit
exit
nmetreau@G102-08:~$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
ff07bc56f1a0   ubuntu        "bash"                  13 minutes ago   Up 2 minutes                quirky_driscoll
nmetreau@G102-08:~$ docker stop ff07
ff07
nmetreau@G102-08:~$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
nmetreau@G102-08:~$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
ff07bc56f1a0   ubuntu        "bash"                  14 minutes ago   Exited (137)  15 seconds ago   quirky_driscoll
d39924dd7854   ubuntu        "bash"                  12 days ago     Exited (137)  5 minutes ago    boring_goodall
9f5321f510ce   hello-world   "/hello"                12 days ago     Exited (0)    6 minutes ago    optimistic_merkle
nmetreau@G102-08:~$ docker rm boring_goodall
boring_goodall
nmetreau@G102-08:~$ docker rm quirky_driscoll
quirky_driscoll
nmetreau@G102-08:~$ docker rm quirky_driscoll rm 9f
9f
Error response from daemon: No such container: quirky_driscoll
Error response from daemon: No such container: rm
nmetreau@G102-08:~$ docker rm 9f
Error response from daemon: No such container: 9f
nmetreau@G102-08:~$
nmetreau@G102-08:~$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
nmetreau@G102-08:~$
```

9. Création d'une image à la main avec docker commit.

- Nous lançons un conteneur ayant pour image ubuntu qui sera détruit quand nous le quittons

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ docker run -it --rm ubuntu bash
root@623b78a1f190:/# ls
bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
root@623b78a1f190:/# exit
exit
nmetreau@G102-08:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
nmetreau@G102-08:~$
```

- Création d'un nouveau conteneur ayant pour nom "test_modification".

```
root@cfc523bab38:/
nmetreau@G102-08:~$
nmetreau@G102-08:~$ docker run -it --name test_modification ubuntu
root@cfc523bab38:/#
```

- Nous créons un fichier vide de test puis nous supprimons le fichier os et quittons ce conteneur.

```
nmetreau@G102-08: ~
root@cfc523bab38:/#
root@cfc523bab38:/# cd home
root@cfc523bab38:/home# touch fichier_test
root@cfc523bab38:/home# cd ..
root@cfc523bab38:/# ls
bin boot dev etc home lib lib64 media mnt opt proc root run sbin srv sys tmp usr var
root@cfc523bab38:/# cd lib
root@cfc523bab38:/lib# ls
apt dpkg init locale lsb mime os-release sysctl.d systemd tmpfiles.d udev x86_64-linux-gnu
root@cfc523bab38:/lib# rm os-release
root@cfc523bab38:/lib# exit
exit
nmetreau@G102-08:~$
```

- Nous effectuons docker diff test_modification afin de lister tous les changements effectués dans le système de fichiers du conteneur par rapport à son image d'origine .

```
nmetreau@G102-08: ~
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

nmetreau@G102-08:~$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
nmetreau@G102-08:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
cfc523bab38   ubuntu   "/bin/bash"   25 hours ago   Exited (0) 25 hours ago
nmetreau@G102-08:~$ docker diff test_modification
C /home
A /home/fichier_test
C /root
A /root/.bash_history
C /usr
C /usr/lib
D /usr/lib/os-release
nmetreau@G102-08:~$
```

- Nous effectuons la commande docker commit afin de sauvegarder les modifications effectuées dans un conteneur

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ docker commit --help
Usage: docker commit [OPTIONS] CONTAINER [REPOSITORY[:TAG]]

Create a new image from a container's changes

Aliases:
  docker container commit, docker commit

Options:
  -a, --author string      Author (e.g., "John Hannibal Smith <hannibal@a-team.com>")
  -c, --change list        Apply Dockerfile instruction to the created image
  -m, --message string     Commit message
      --no-pause           Disable pausing container during commit
nmetreau@G102-08:~$
```

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ docker commit test_modification ubuntu:1.0
sha256:3b6083dfff1138206b4f4315536ece2f7d44e3ccf2cb83282f632e4c97a5c2d34
nmetreau@G102-08:~$
```

- Nous regardons que l'image ce soit bien créée

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ docker images
nmetreau@G102-08:~$ docker images
```

IMAGE	ID	DISK USAGE	CONTENT SIZE	EXTRA
hello-world:latest	85404b3c5395	25.9kB	9.52kB	
ubuntu:latest	d1e2e92c075e	119MB	31.7MB	U
ubuntu:1.0	3b6083dfff113	117MB	29.7MB	

```
nmetreau@G102-08:~$
```

- Nous listons les conteneurs

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ docker ps -a
nmetreau@G102-08:~$ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
cfc523bab38	ubuntu	"/bin/bash"	25 hours ago	Exited (0) 25 hours ago		test_modification

```
nmetreau@G102-08:~$ docker rm test_modification
test_modification
nmetreau@G102-08:~$
```

- Nous lançons un nouveau conteneur et nous vérifions s'il bien du fichier test

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ docker run -it --rm --name test ubuntu:1.0
root@681c5a8fbff9:/# ls /home
fichier_test  ubuntu
root@681c5a8fbff9:/# exit
exit
nmetreau@G102-08:~$ docker ps -a
nmetreau@G102-08:~$ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

```
nmetreau@G102-08:~$
```

10. Première approche des volumes.

- Nous créons un répertoire web dans /home/"nmetreau" de la machine ubuntu ainsi qu'une page html :

```
nmetreau@G102-08: ~/web
nmetreau@G102-08:~$ pwd
/home/nmetreau
nmetreau@G102-08:~$ mkdir web
nmetreau@G102-08:~$ cd web
nmetreau@G102-08:~/web$ nano index.html
nmetreau@G102-08:~/web$
```

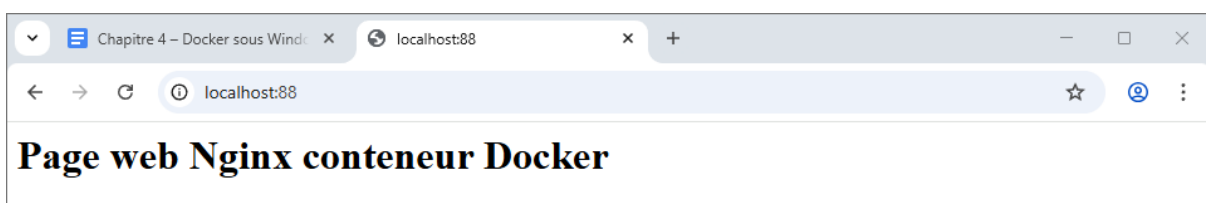
```
GNU nano 7.2 index.html
<html>
<body>
<h1>Page web Nginx conteneur Docker</h1>
</body>
</html>
```

- Nous créons un conteneur qui va pointer sur le répertoire web du FS local accessible en lecture :

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ docker run -d --name web -v /home/nmetreau/web:/usr/share/nginx/html:ro -p 88:80 nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
5e815e07e569: Pull complete
3189680c601f: Pull complete
cde7a05ae428: Pull complete
bb3d0aa29654: Pull complete
ec781dee3f47: Pull complete
510ddf6557d6: Pull complete
587e3d84dbb5: Pull complete
96a6cfe061e0: Download complete
669e0ab8e7fa: Download complete
Digest: sha256:7150b3a39203cb5bee612ff4a9d18774f8c7caf6399d6e8985e97e28eb751c18
Status: Downloaded newer image for nginx:latest
8caeeaeef2e1cd154854c1f172e208314730a05fee67700914db89a295464e75
nmetreau@G102-08:~$
```

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
8caeeaeef2e1   nginx    "/docker-entrypoint..." 44 seconds ago Up 43 seconds 0.0.0.0:88->80/tcp, [::]:88->80/tcp web
nmetreau@G102-08:~$
```

- Nous testons l'accès aux données :



- Affichage des logs du conteneur web :

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ 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 14:18:21 [notice] 1#1: using the "epoll" event method
2026/03/25 14:18:21 [notice] 1#1: nginx/1.29.7
2026/03/25 14:18:21 [notice] 1#1: built by gcc 14.2.0 (Debian 14.2.0-19)
2026/03/25 14:18:21 [notice] 1#1: OS: Linux 6.6.87.2-microsoft-standard-WSL2
2026/03/25 14:18:21 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2026/03/25 14:18:21 [notice] 1#1: start worker processes
2026/03/25 14:18:21 [notice] 1#1: start worker process 29
2026/03/25 14:18:21 [notice] 1#1: start worker process 30
2026/03/25 14:18:21 [notice] 1#1: start worker process 31
2026/03/25 14:18:21 [notice] 1#1: start worker process 32
2026/03/25 14:18:21 [notice] 1#1: start worker process 33
2026/03/25 14:18:21 [notice] 1#1: start worker process 34
2026/03/25 14:18:21 [notice] 1#1: start worker process 35
2026/03/25 14:18:21 [notice] 1#1: start worker process 36
2026/03/25 14:18:21 [notice] 1#1: start worker process 37
2026/03/25 14:18:21 [notice] 1#1: start worker process 38
2026/03/25 14:18:21 [notice] 1#1: start worker process 39
2026/03/25 14:18:21 [notice] 1#1: start worker process 40
2026/03/25 14:18:21 [notice] 1#1: start worker process 41
2026/03/25 14:18:21 [notice] 1#1: start worker process 42
2026/03/25 14:18:21 [notice] 1#1: start worker process 43
2026/03/25 14:18:21 [notice] 1#1: start worker process 44
172.17.0.1 - - [25/Mar/2026:14:19:34 +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" "-"
172.17.0.1 - - [25/Mar/2026:14:19:34 +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" "-"
2026/03/25 14:19:34 [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", referer: "http://localhost:88/"
nmetreau@G102-08:~$
```

- Nous arrêtons et supprimons le conteneur web ainsi que le mappage de volume :

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ docker rm -fv web
web
nmetreau@G102-08:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
nmetreau@G102-08:~$

nmetreau@G102-08: ~/web
nmetreau@G102-08:~$ cd web
nmetreau@G102-08:~/web$ ls
index.html
nmetreau@G102-08:~/web$
```

11. Gestion des volumes en écriture.

11.1. Premier type de volume : volume nommé

- Nous récupérons l'image mongodb-community-server avec la commande docker pull :

```
nmetreau@G102-08: ~/data
nmetreau@G102-08:~/data$ docker pull mongodb/mongodb-community-server:latest
latest: Pulling from mongodb/mongodb-community-server
96c832531c38: Pull complete
2e4f7f4523be: Pull complete
568a1e397067: Pull complete
6ad2e513b847: Pull complete
4f4fb700ef54: Pull complete
d68094177302: Pull complete
ae6c644e0c20: Pull complete
f3e0282bbb14: Pull complete
fe2fbbbc7ec0: Pull complete
6989f79dc9b5: Pull complete
a9af3f48621c: Pull complete
Digest: sha256:7266392360c651bf99ad725eb78ee17d723e51a8f4cb7c3382aa89e7b2c3eee3
Status: Downloaded newer image for mongodb/mongodb-community-server:latest
docker.io/mongodb/mongodb-community-server:latest
nmetreau@G102-08:~/data$
```

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

```
nmetreau@G102-08: ~/data
nmetreau@G102-08:~/data$ docker run -d -p 27017:27017 -e MONGODB_INITDB_ROOT_USERNAME=sio -e MONGO_INITDB_ROOT_PASSWORD=password1234 --name mongodb mongodb/mongodb-community-server:latest
71bad3ae83fdd4d6f78ddad691deb7baa32af79a861e9264a0550c429dcd89c
nmetreau@G102-08:~/data$
```

- Nous installons MongoDB Shell sur la machine physique pour pouvoir vous connecter au serveur MongoDB :

MongoDB Atlas MongoDB Enterprise Advanced MongoDB Community Edition Tools MongoDB Atlas Terraform Provider MongoDB Shell MongoDB Compass (GUI) Atlas CLI

Learn more

Version 2.8.2

Platform Windows x64 (10+)

Package msi

Download Copy link More Options

- Connexion au serveur MongoDB :
 - Nous créons la base de données ma_bdd.
 - Suite à la création de la bdd, nous créons une collection "etudiants" en ajoutant comme document "smet" pour le nom et "jp" pour son prénom.

```
Administrateur : Windows PowerShell
PS C:\WINDOWS\system32> mongosh mongodb://localhost:27017
Current Mongosh Log ID: 69c3f5735356b4d523114029
Connecting to:      mongodb://localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.8.1
Using MongoDB:      7.0.8
Using Mongosh:       2.8.1

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

-----
  The server generated these startup warnings when booting
  2026-03-25T13:40:56.283+01:00: 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.createCollection("etudiants");
{ ok: 1 }
ma_bdd> db.etudiants.insertOne({"nom":"smet", "prenom":"jp"});
{
  acknowledged: true,
  insertedId: ObjectId('69c3f5d05356b4d52311402a')
}
ma_bdd> db.etudiants.find();
[
  {
    _id: ObjectId('69c3f5d05356b4d52311402a'),
    nom: 'smet',
    prenom: 'jp'
  }
]
ma_bdd> exit
PS C:\WINDOWS\system32>
```

- Nous renvoyons toutes les informations techniques d'un conteneur (ou d'une image) au format JSON à l'aide de la commande docker inspect mongodb.

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ 
nmetreau@G102-08:~$ docker inspect mongodb
[
  {
    "Id": "f71bad3ae83f0d4d6f70ddad691deb7baa32af79a861e9264a0550c429dcd89c",
    "Created": "2026-03-25T14:25:45.125412799Z",
    "Path": "python3",
    "Args": [
      "/usr/local/bin/docker-entrypoint.py",
      "mongod"
    ],
    "State": {
      "Status": "running",
      "Running": true,

```

```

nmetreau@G102-08: ~
},
"Image": "sha256:7266392360c651bf99ad725eb78ee17d723e51a8f4cb7c3382aa89e7b2c3eee3",
"ResolvConfPath": "/var/lib/docker/containers/f71bad3ae83f0d4d6f70ddad691deb7baa32af79a861e9264a0550c429dcd89c/resolv.conf",
"HostnamePath": "/var/lib/docker/containers/f71bad3ae83f0d4d6f70ddad691deb7baa32af79a861e9264a0550c429dcd89c/hostname",
"HostsPath": "/var/lib/docker/containers/f71bad3ae83f0d4d6f70ddad691deb7baa32af79a861e9264a0550c429dcd89c/hosts",
"LogPath": "/var/lib/docker/containers/f71bad3ae83f0d4d6f70ddad691deb7baa32af79a861e9264a0550c429dcd89c/f71bad3ae83f0d4d6f70ddad691deb7baa32af79a861e9264a0550c429dcd89c-json.log",
"Name": "/mongodb",
"RestartCount": 0,
"Driver": "overlayfs",
"Platform": "linux",
"MountLabel": "",
"ProcessLabel": "",
"AppArmorProfile": "",
"ExecIDs": null,
"HostConfig": {
  "Binds": null,
  "ContainerIDFile": "",
  "LogConfig": {
    "Type": "json-file",
    "Config": {}
  },
  "NetworkMode": "bridge",
  "PortBindings": {
    "27017/tcp": [
      {
        "HostIp": "",
        "HostPort": "27017"
      }
    ]
  },
  "RestartPolicy": {
    "Name": "no",
    "MaximumRetryCount": 0
  },
  "AutoRemove": false,
  "VolumeDriver": "",
  "VolumesFrom": null,
  "ConsoleSize": [
    47,
    150
  ],
  "CapAdd": null,
  "CapDrop": null,
  "CgroupnsMode": "private",
  "Dns": null,
  "DnsOptions": [],
  "DnsSearch": [],
  "ExtraHosts": null,
  "GroupAdd": null,
  "IpcMode": "private",
  "Cgroup": "",
  "Links": null,
  "OomScoreAdj": 0,
  "PidMode": "",
  "Privileged": false,
  "PublishAllPorts": false,
  "ReadonlyRootfs": false,
  "SecurityOpt": null,
  "UTSMode": "",
  "UsersMode": "",
  "ShmSize": 67108864,
  "Runtime": "runc",
  "Isolation": "",
  "CpuShares": 0,
  "Memory": 0,

```

- Nous accédons au conteneur puis aux données et nous arrêtons le conteneur :

```
nmetreau@G102-08: ~
nmetreau@G102-08:~$ docker exec -it mongodb bash
mongodb@f71bad3ae83f:/data$ cd /data
mongodb@f71bad3ae83f:/data$ ls
configdb.db
mongodb@f71bad3ae83f:/data$ cd db
mongodb@f71bad3ae83f:/data$ ls
WiredTiger            _tmp
WiredTiger.lock       collection-1806f827-85be-48be-a860-31a8690a4e71.wt  index-6902b5bf-21a5-4f9e-81e1-01c13452e6a1.wt  journal
WiredTiger.turtle     collection-41baef0e-7175-43a0-918a-e7a1878b3778.wt  index-94b4bc75-4802-40be-9c5b-57e15812fe73.wt  mongod.lock
WiredTiger.wt         collection-8f6b38cd-c6c6-462b-be37-c5e29f46bfc8.wt  index-a9fd0973-7225-4caa-a1ec-e8e2701ea4d2.wt  sizeStorer.wt
WiredTigerHS.wt       collection-fa9cb867-66b4-4d66-9b70-9dde7cad21b8.wt  index-aea4130a-5b4c-432c-af62-0c2ba7343198.wt  storage.bson
_mdb_catalog.wt       diagnostic.data
mongodb@f71bad3ae83f:/data$ exit
nmetreau@G102-08:~$
```

- Suppression du conteneur :

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

- Création d'un volume nommé à l'aide de la commande docker volume create :

```
nmetreau@PCDeNitrooX: ~
nmetreau@PCDeNitrooX:~$ docker volume create mon_volume_mongodb
mon_volume_mongodb
nmetreau@PCDeNitrooX:~$
```

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

nmetreau@PCDeNitrooX:~$ docker volume ls
DRIVER      VOLUME NAME
local       4bfc1efd858c783a5bdc88a86495938b1745e12cf7b0d8d958e0539fee2be024
local       b60b39695868a8b58411314e62c6d867e6971ef95335fd55903a34d7c66b899a
local       mon_volume_mongodb
nmetreau@PCDeNitrooX:~$
```

```
nmetreau@PCDeNitrooX: ~
nmetreau@PCDeNitrooX:~$ docker volume inspect mon_volume_mongodb
[
  {
    "CreatedAt": "2026-03-27T21:04:28Z",
    "Driver": "local",
    "Labels": null,
    "Mountpoint": "/var/lib/docker/volumes/mon_volume_mongodb/_data",
    "Name": "mon_volume_mongodb",
    "Options": null,
    "Scope": "local"
  }
]
nmetreau@PCDeNitrooX:~$
```

- Nous lançons un conteneur avec un mappage de volume (volume nommé:/volume du conteneur) :

```
nmetreau@PCDeNitrooX: ~
nmetreau@PCDeNitrooX:~$ 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
d92dccc258f7b11c87145ecb51b26da31adaaa28cfc789e67e6f1cad05514380d
nmetreau@PCDeNitrooX:~$
```

- Nous nous connectons au serveur MongoDB :

```

mongosh mongodb://<credentials>@localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000
PS C:\WINDOWS\system32> mongosh mongodb://sio:password1234@localhost:27017
Current Mongosh Log ID: 69c6f24e5e8c36e6123682d0
Connecting to:      mongodb://<credentials>@localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000&app
Name=mongosh+2.8.2
Using MongoDB:      8.2.6
Using Mongosh:      2.8.2

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

-----
The server generated these startup warnings when booting
2026-03-27T21:09:21.846+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See
http://dochub.mongodb.org/core/prodnotes-filesystem
2026-03-27T21:09:22.391+00:00: For customers running the current memory allocator, we suggest changing the contents o
f the following sysfsFile
2026-03-27T21:09:22.391+00:00: For customers running the current memory allocator, we suggest changing the contents o
f the following sysfsFile
2026-03-27T21:09:22.391+00:00: We suggest setting the contents of sysfsFile to 0.
2026-03-27T21:09:22.391+00:00: We suggest setting swappiness to 0 or 1, as swapping can cause performance problems.
-----
test>

```

```

Administrateur : Windows PowerShell
The server generated these startup warnings when booting
2026-03-27T21:09:21.846+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. S
ee http://dochub.mongodb.org/core/prodnotes-filesystem
2026-03-27T21:09:22.391+00:00: For customers running the current memory allocator, we suggest changing the contents o
f the following sysfsFile
2026-03-27T21:09:22.391+00:00: For customers running the current memory allocator, we suggest changing the contents o
f the following sysfsFile
2026-03-27T21:09:22.391+00:00: We suggest setting the contents of sysfsFile to 0.
2026-03-27T21:09:22.391+00:00: We suggest setting swappiness to 0 or 1, as swapping can cause performance problems.
-----
test> use("ma_bdd");
switched to db ma_bdd
ma_bdd> db.createCollection("etudiants");
{ ok: 1 }
ma_bdd> db.etudiants.insertOne({"nom":"smet", "prenom":"jp"});
{
  acknowledged: true,
  insertedId: ObjectId('69c6f2c35e8c36e6123682d1')
}
ma_bdd> db.etudiants.find();
[
  {
    _id: ObjectId('69c6f2c35e8c36e6123682d1'),
    nom: 'smet',
    prenom: 'jp'
  }
]
ma_bdd> exit
PS C:\WINDOWS\system32>

```

```

nmetreau@PCDeNitrooX: ~
nmetreau@PCDeNitrooX:~$ docker ps
CONTAINER ID   IMAGE                                     NAMES                COMMAND                CREATED    STATUS    PORTS
d92dcc258f7b   mongodb/mongodb-community-server:latest  "python3 /usr/local/..."  3 minutes ago  Up 3 minutes  0.0.0.0
:27017->27017/tcp, [::]:27017->27017/tcp  mongodb
nmetreau@PCDeNitrooX:~$

```

- Nous arrêtons le conteneur et nous le supprimons :

```
nmetreau@PCDeNitrooX: ~
nmetreau@PCDeNitrooX:~$ docker stop mongodb
nmetreau@PCDeNitrooX:~$ docker rm -v mongodb
nmetreau@PCDeNitrooX:~$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS        NAMES
nmetreau@PCDeNitrooX:~$
```

- Nous démarrons un conteneur mongo :

```
nmetreau@PCDeNitrooX: ~
nmetreau@PCDeNitrooX:~$ 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
e73aac4b6e20f85f1db65e9257788940329565281c70125a03adc57db7bae0cb
nmetreau@PCDeNitrooX:~$
```

- Nous regardons si la collection "etudiants" est présente :

```
Administrateur : Windows PowerShell
Name=mongosh+2.8.2
Using MongoDB:      8.2.6
Using Mongosh:      2.8.2

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

-----
The server generated these startup warnings when booting
2026-03-27T21:14:40.003+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
2026-03-27T21:14:40.356+00:00: For customers running the current memory allocator, we suggest changing the contents of the following sysfsFile
2026-03-27T21:14:40.356+00:00: For customers running the current memory allocator, we suggest changing the contents of the following sysfsFile
2026-03-27T21:14:40.356+00:00: We suggest setting the contents of sysfsFile to 0.
2026-03-27T21:14:40.356+00:00: We suggest setting swappiness to 0 or 1, as swapping can cause performance problems.
-----

test> use("ma_bdd");
switched to db ma_bdd
ma_bdd> db.etudiants.find();
[
  {
    _id: ObjectId('69c6f2c35e8c36e6123682d1'),
    nom: 'smet',
    prenom: 'jp'
  }
]
ma_bdd> exit
PS C:\WINDOWS\system32>
```

- Suppression du conteneur :

```
nmetreau@PCDeNitrooX: ~
nmetreau@PCDeNitrooX:~$ docker rm -fv mongodb
nmetreau@PCDeNitrooX:~$
```

- Nous lançons un conteneur et nous ajoutons une deuxième ligne pour vérifier la persistance :

```
nmetreau@PCDeNitrooX: ~  
nmetreau@PCDeNitrooX:~$ 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  
e092cf208a993da495e248358fd8cc2462c72eb0ca54bcf00e9f38f765a8be4  
nmetreau@PCDeNitrooX:~$  
  
test> use("ma_bdd");  
switched to db ma_bdd  
ma_bdd> db.etudiants.insertOne({"nom":"hallyday", "prenom":"johnny"});  
{  
  acknowledged: true,  
  insertedId: ObjectId('69c6f4530b9c2762843682d1')  
}  
ma_bdd> db.etudiants.find();  
[  
  {  
    _id: ObjectId('69c6f2c35e8c36e6123682d1'),  
    nom: 'smet',  
    prenom: 'jip'  
  },  
  {  
    _id: ObjectId('69c6f4530b9c2762843682d1'),  
    nom: 'hallyday',  
    prenom: 'johnny'  
  }  
]  
ma_bdd> exit  
PS C:\WINDOWS\system32>
```

- Nous supprimons le conteneur :

```
nmetreau@PCDeNitrooX: ~  
nmetreau@PCDeNitrooX:~$ docker rm -fv mongodb  
mongodb  
nmetreau@PCDeNitrooX:~$
```

- Nous relançons un conteneur et nous ajoutons une deuxième ligne pour vérifier la persistance des données :

```
nmetreau@PCDeNitrooX: ~  
nmetreau@PCDeNitrooX:~$ 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  
521998ba61321a680e1551a854a25e5c9f9f0683a815f6584aeba5f38d8b4678  
nmetreau@PCDeNitrooX:~$
```

```

Administrateur : Windows PowerShell
PS C:\WINDOWS\system32> mongosh mongodb://sio:password1234@localhost:27017
Current Mongosh Log ID: 69c6f4f7c3f3461d373682d0
Connecting to:      mongodb://<credentials>@localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000&app
Name=mongosh+2.8.2
Using MongoDB:      8.2.6
Using Mongosh:       2.8.2

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

-----
The server generated these startup warnings when booting
2026-03-27T21:20:45.673+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See
http://dochub.mongodb.org/core/prodnotes-filesystem
2026-03-27T21:20:45.976+00:00: For customers running the current memory allocator, we suggest changing the contents o
f the following sysfsFile
2026-03-27T21:20:45.976+00:00: For customers running the current memory allocator, we suggest changing the contents o
f the following sysfsFile
2026-03-27T21:20:45.976+00:00: We suggest setting the contents of sysfsFile to 0.
2026-03-27T21:20:45.976+00:00: We suggest setting swappiness to 0 or 1, as swapping can cause performance problems.
-----

test> use("ma_bdd");
switched to db ma_bdd
ma_bdd> db.etudiants.find();
[
  {
    _id: ObjectId('69c6f2c35e8c36e6123682d1'),
    nom: 'smet',
    prenom: 'jp'
  },
  {
    _id: ObjectId('69c6f4530b9c2762843682d1'),
    nom: 'hallyday',
    prenom: 'johnny'
  }
]
ma_bdd> exit
PS C:\WINDOWS\system32>

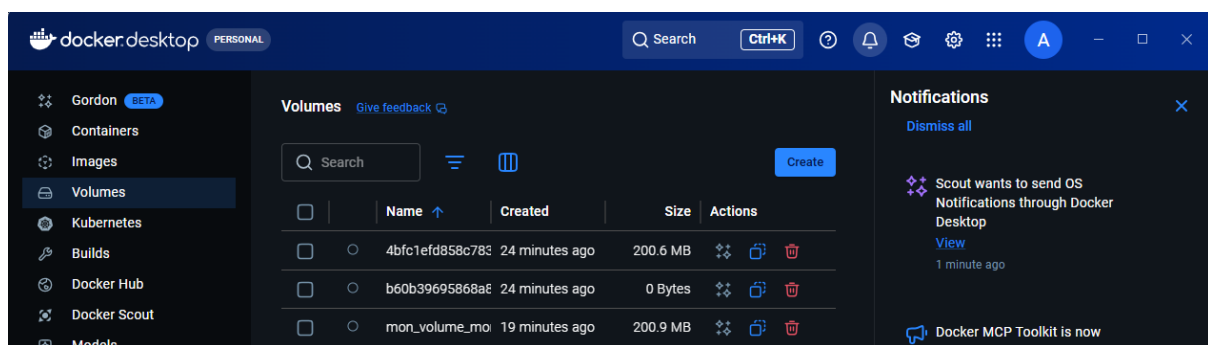
```

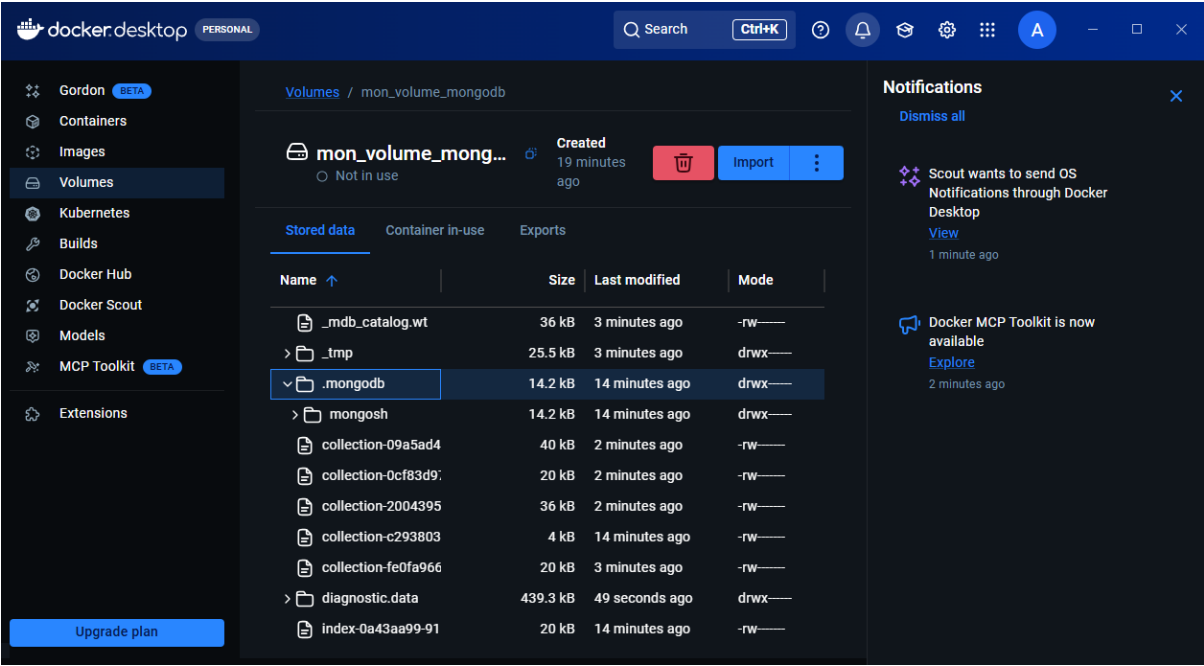
- Nous supprimons le conteneur :

```

nmetreau@PCDeNitrooX: ~
nmetreau@PCDeNitrooX:~$ docker rm -fv mongodb
mongodb
nmetreau@PCDeNitrooX:~$

```





11.2. Deuxième type de volume : montage lié (bind mounting)

- Nous créons le répertoire data :

```
nmetreau@PCDeNitrooX: ~
nmetreau@PCDeNitrooX:~$ mkdir data
mkdir: cannot create directory 'data': File exists
nmetreau@PCDeNitrooX:~$ ls
data
nmetreau@PCDeNitrooX:~$

nmetreau@PCDeNitrooX: ~
nmetreau@PCDeNitrooX:~$ docker run -d --name mongodb2 -p 27017:27017 -e MONGODB_INITDB_ROOT_USERNAME=sio -e MONGODB_INITDB_ROOT_PASSWORD=password1234 -v /home/nmetreau/data:/data/db mongodb/mongodb-community-server:latest
b0e55c695fff4eba132580d1588d15c5ebf47bd52fe824d0a21e1f6f98b2890d
nmetreau@PCDeNitrooX:~$
```

- Dans cette collection, nous ajoutons “eddy mitchemm”

```
Administrateur : Windows PowerShell
PS C:\WINDOWS\system32> mongosh mongodb://sio:password1234@localhost:27017
Current Mongosh Log ID: 69c6f71baa3460a8143682d0
Connecting to: mongodb://<credentials>@localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.8.2
Using MongoDB: 8.2.6
Using Mongosh: 2.8.2

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

-----
The server generated these startup warnings when booting
2026-03-27T21:29:47.572+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
2026-03-27T21:29:47.935+00:00: For customers running the current memory allocator, we suggest changing the contents of the following sysfsFile
2026-03-27T21:29:47.935+00:00: For customers running the current memory allocator, we suggest changing the contents of the following sysfsFile
2026-03-27T21:29:47.935+00:00: We suggest setting the contents of sysfsFile to 0.
2026-03-27T21:29:47.935+00:00: We suggest setting swappiness to 0 or 1, as swapping can cause performance problems.
-----

test> use("ma_bdd");
switched to db ma_bdd
ma_bdd> db.etudiants.insertOne({"nom":"mitchell", "prenom":"eddy"});
{
  acknowledged: true,
  insertedId: ObjectId('69c6f72faa3460a8143682d1')
}
ma_bdd> db.etudiants.find();
[
  {
    _id: ObjectId('69c6f72faa3460a8143682d1'),
    nom: 'mitchell',
    prenom: 'eddy'
  }
]
ma_bdd> exit
PS C:\WINDOWS\system32>
```

- Nous supprimons le conteneur :

```
Administrateur : Windows PowerShell
PS C:\WINDOWS\system32> docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS          NAMES
b0e55c695fff   mongodb/mongodb-community-server:latest  "python3 /usr/local/..."  2 minutes ago  Up About a minute  0.0.0.0:27017->27017/tcp, [::]:27017->27017/tcp  mongodb2
PS C:\WINDOWS\system32> docker stop mongodb2
mongodb2
PS C:\WINDOWS\system32> docker rm -v mongodb2
mongodb2
PS C:\WINDOWS\system32> docker ps -a
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS          NAMES
PS C:\WINDOWS\system32>
```

```

nmetreau@PCDeNitrooX: ~/data
nmetreau@PCDeNitrooX:~$ cd data/
nmetreau@PCDeNitrooX:~/data$ ls
WiredTiger                                diagnostic.data
WiredTiger.lock                          index-3d3663d1-3240-4c61-a413-dfd56a3fb609.wt
WiredTiger.turtle                       index-66dab97e-4ac7-484c-882a-cb751dfe0940.wt
WiredTiger.wt                           index-7d4345cc-4c8d-4f11-bee6-96e150663534.wt
WiredTigerHS.wt                         index-93707564-96c5-4294-9157-bb43598675cc.wt
_mdb_catalog.wt                        index-c45133a8-4131-48af-9943-514a7bce855f.wt
_tmp                                    index-dc3c0006-095a-41ae-93b4-d27bb46d72b4.wt
collection-0bf96b9e-7390-41c2-81d6-961b0ce3d1ab.wt index-fb2fec39-d02d-4788-8ea5-c9ba46688278.wt
collection-23c427c5-24bb-40cd-abaa-37e70f6e1e34.wt journal
collection-abd4e79b-ac4b-4b47-b93f-d068724d1911.wt mongod.lock
collection-d7797c5b-9b57-48e1-9f16-4ed153e25d0f.wt sizeStorer.wt
collection-fd6c0601-6c6f-4a82-b8ab-4cbcd03ca608.wt storage.bson
nmetreau@PCDeNitrooX:~/data$

```

- Nous relançons une nouvelle instance de l'image :

```

nmetreau@PCDeNitrooX: ~
nmetreau@PCDeNitrooX:~$ docker run -d --name mongodb2 -p 27017:27017 -e MONGODB_INITDB_ROOT_USERNAME=sio -e MONGODB_INITDB_ROOT_PASSWORD=password1234 -v /home/nmetreau/data:/data/db mongodb/mongodb-community-server:latest
12802253cc60d9a364f0d657b3e8c1ed240e1a6d78c39c9296751d0f28c6694c
nmetreau@PCDeNitrooX:~$

```

```

Administrateur : Windows PowerShell
PS C:\WINDOWS\system32> mongosh mongodb://sio:password1234@localhost:27017
Current Mongosh Log ID: 69c6f7e432f4ddcc093682d0
Connecting to:      mongodb://<credentials>@localhost:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.8.2
Using MongoDB:      8.2.6
Using Mongosh:      2.8.2

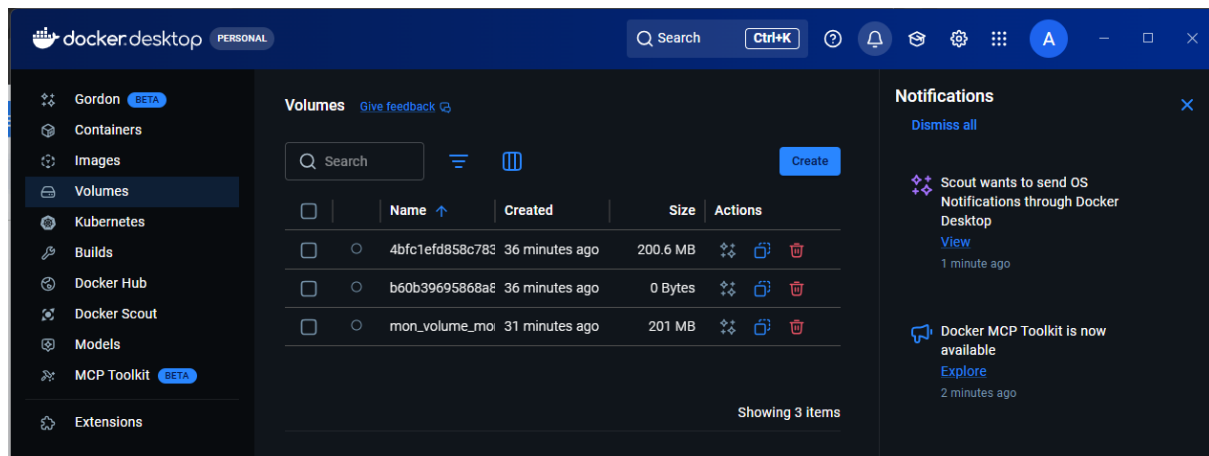
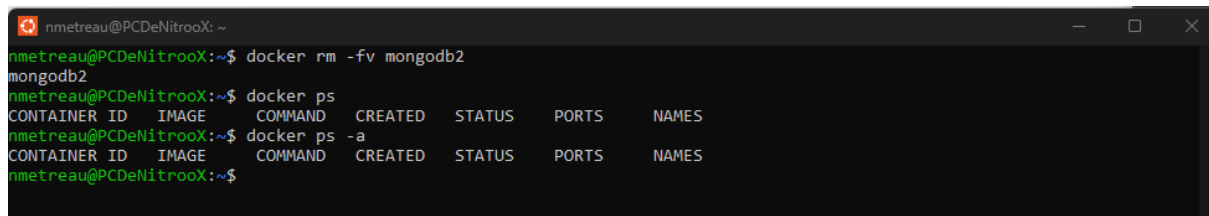
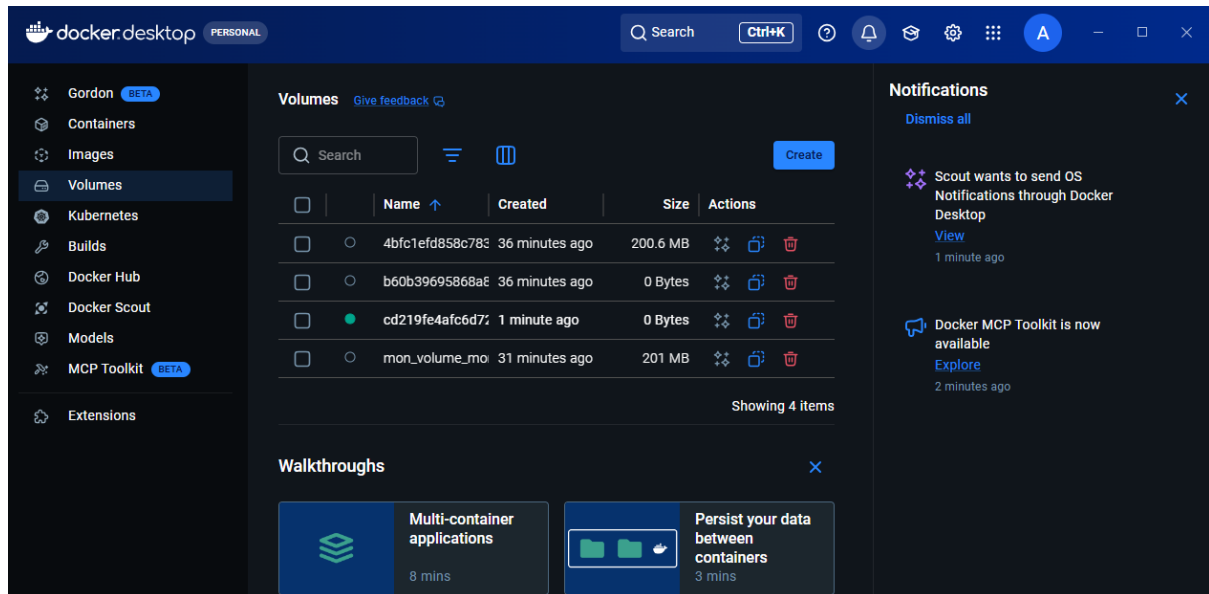
For mongosh info see: https://www.mongodb.com/docs/mongosh-shell/

-----
The server generated these startup warnings when booting
2026-03-27T21:33:37.546+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
2026-03-27T21:33:37.921+00:00: For customers running the current memory allocator, we suggest changing the contents of the following sysfsFile
2026-03-27T21:33:37.921+00:00: For customers running the current memory allocator, we suggest changing the contents of the following sysfsFile
2026-03-27T21:33:37.921+00:00: We suggest setting the contents of sysfsFile to 0.
2026-03-27T21:33:37.921+00:00: We suggest setting swappiness to 0 or 1, as swapping can cause performance problems.
-----

test> use("ma_bdd");
switched to db ma_bdd
ma_bdd> db.etudiants.find();
[
  {
    _id: ObjectId('69c6f72faa3460a8143682d1'),
    nom: 'mitchell',
    prenom: 'eddy'
  }
]
ma_bdd> exit
PS C:\WINDOWS\system32>

```

- Nous constatons la persistance

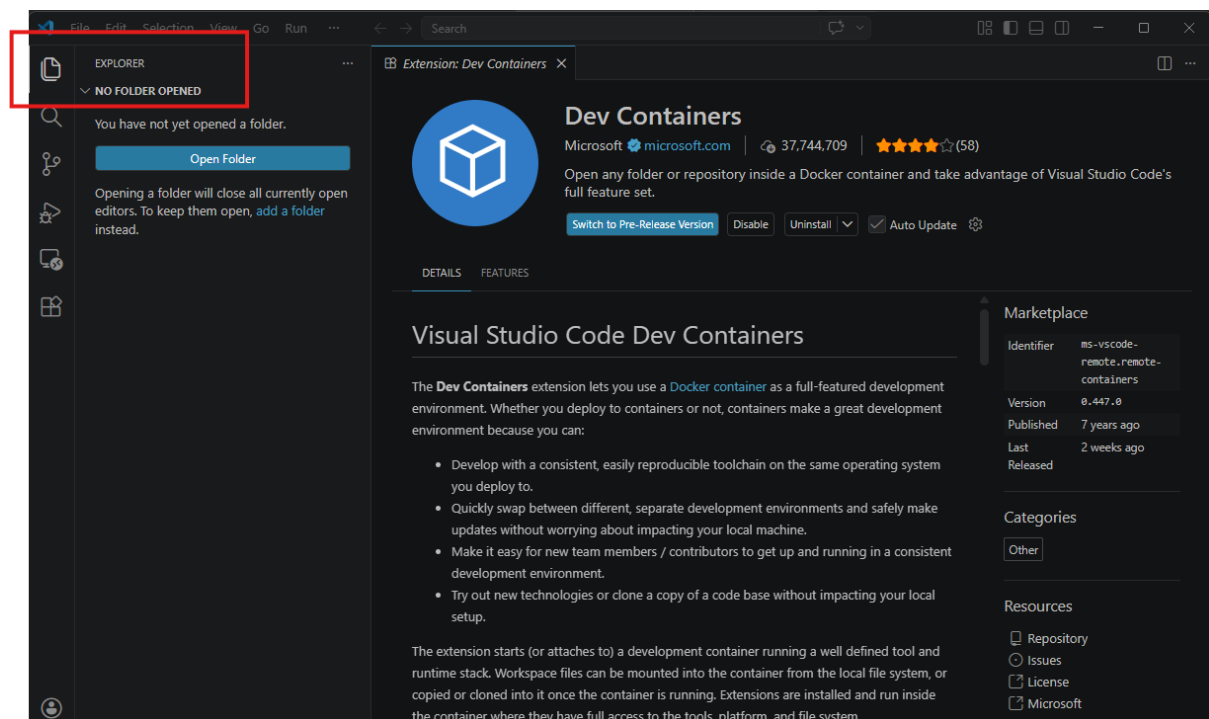


12. Utiliser un conteneur Docker et Visual Studio Code.

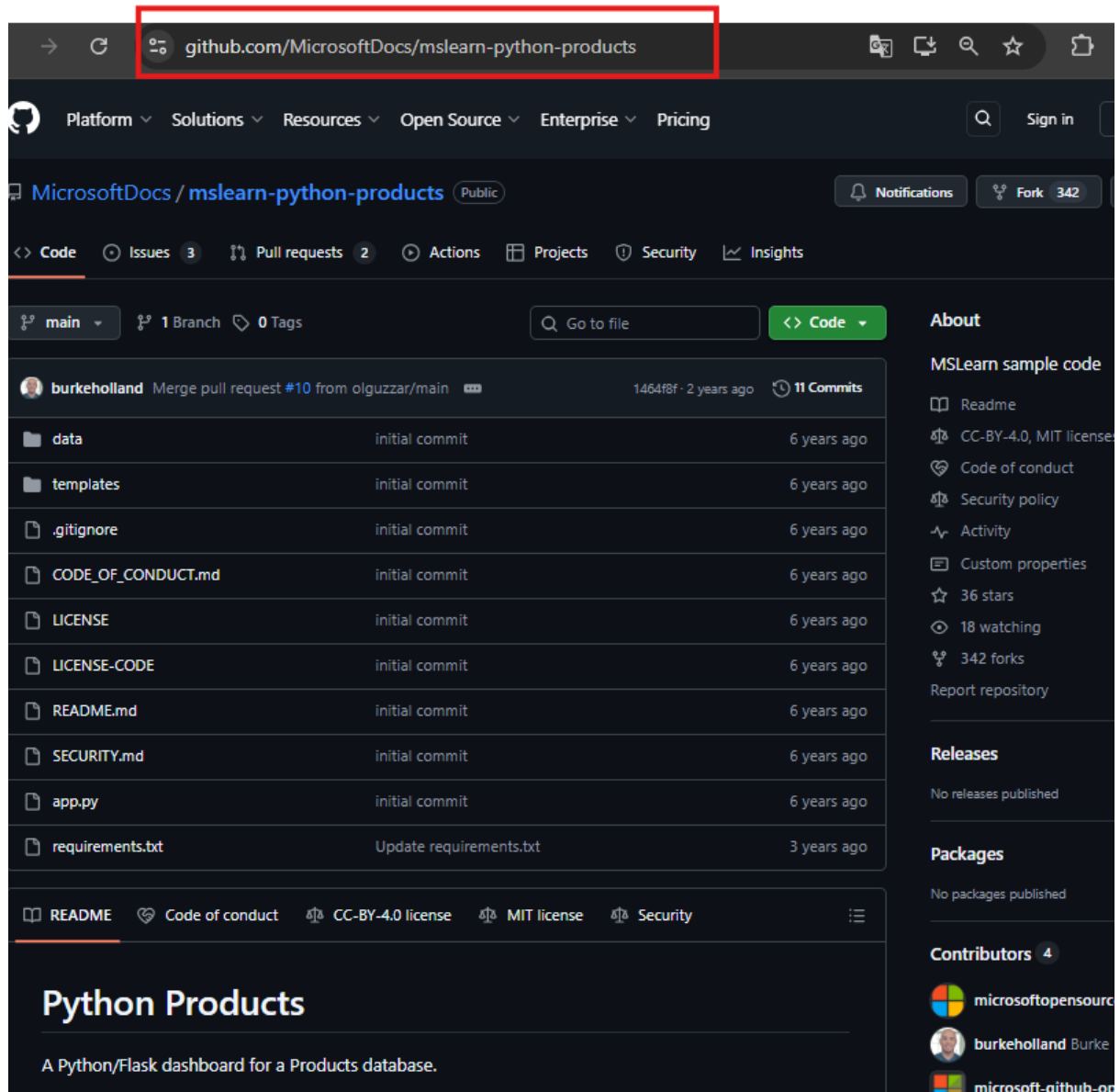
- Installation de Visual Studio Coded ainsi que Git.

Nom	Modifié le	Type	Taille
▼ Aujourd'hui			
Git-2.53.0.2-64-bit	27/03/2026 22:48	Application	63 052 Ko
VSCodeUserSetup-x64-1.113.0	27/03/2026 22:37	Application	136 339 Ko
mongosh-2.8.2-x64	27/03/2026 21:52	Package Windows...	64 452 Ko
Docker Desktop Installer	27/03/2026 21:33	Application	604 703 Ko

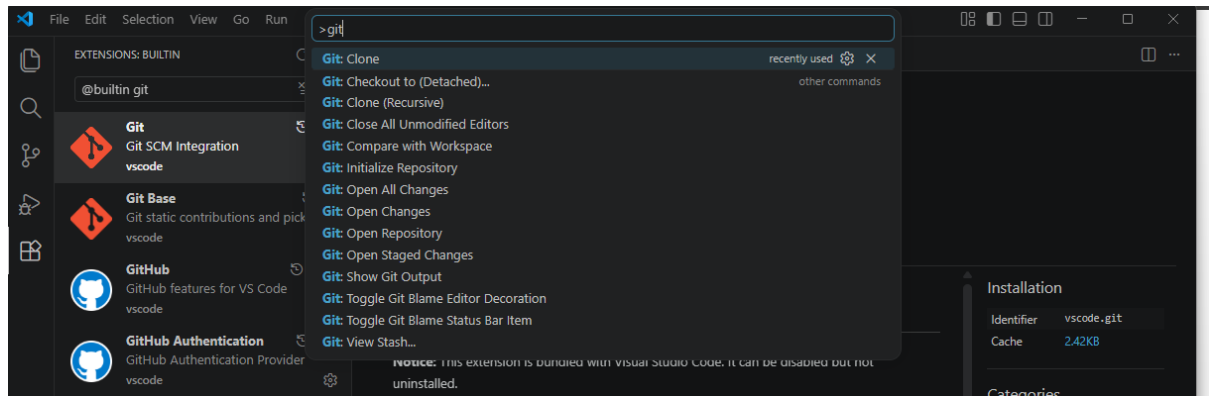
- Nous ouvrons une nouvelle instance dans VS Code :



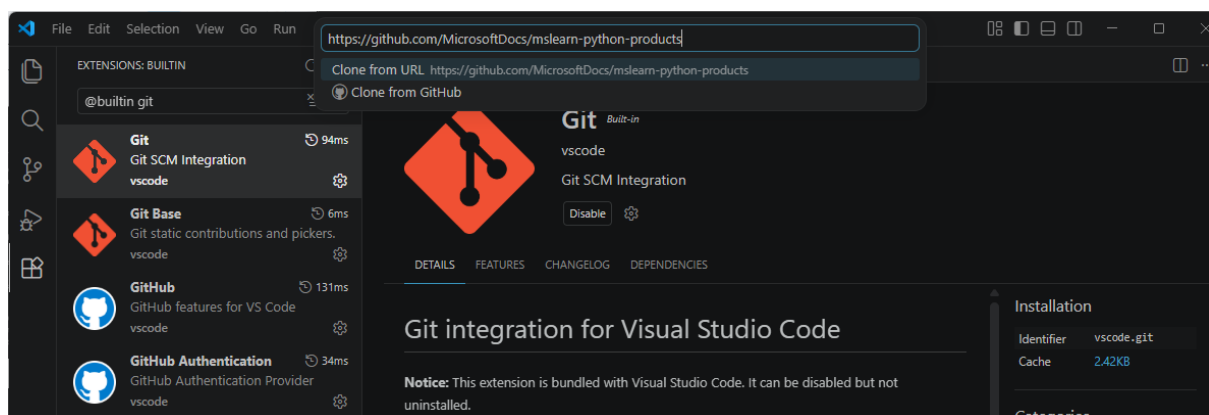
- Nous copions l'URL du projet mslearn-python-products qui nous servira d'exemple :



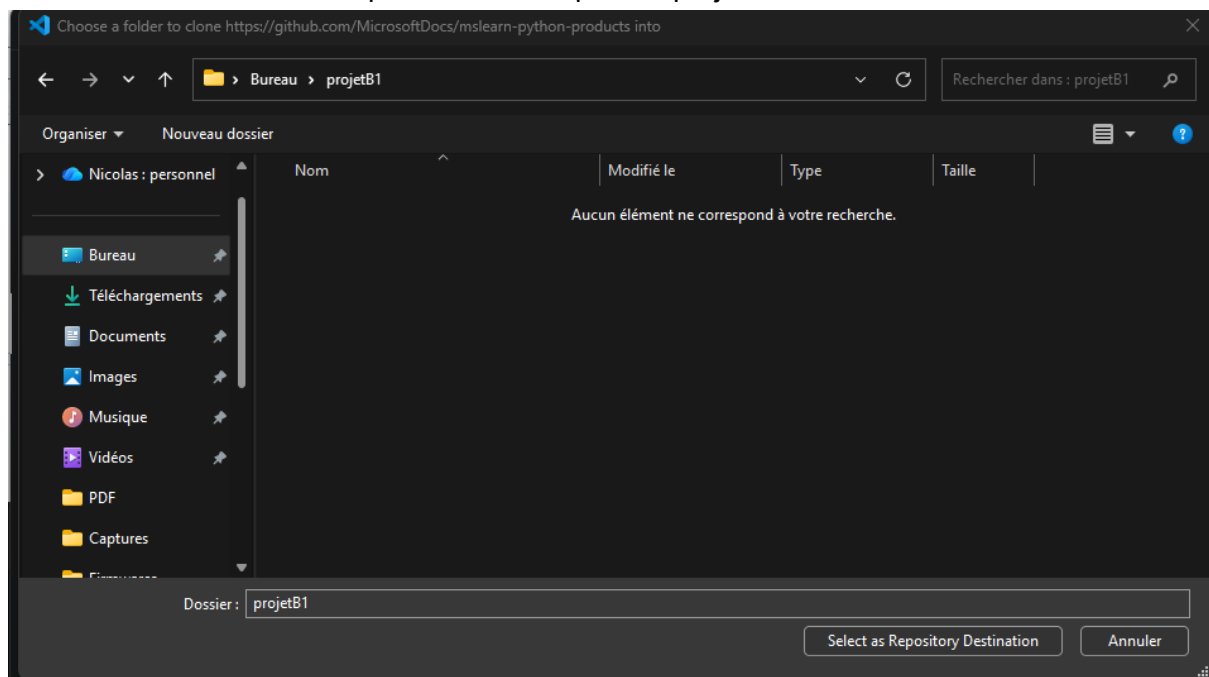
- Nous sélectionnons Clone Repository en appuyant sur F1 pour rechercher Git : Clone :

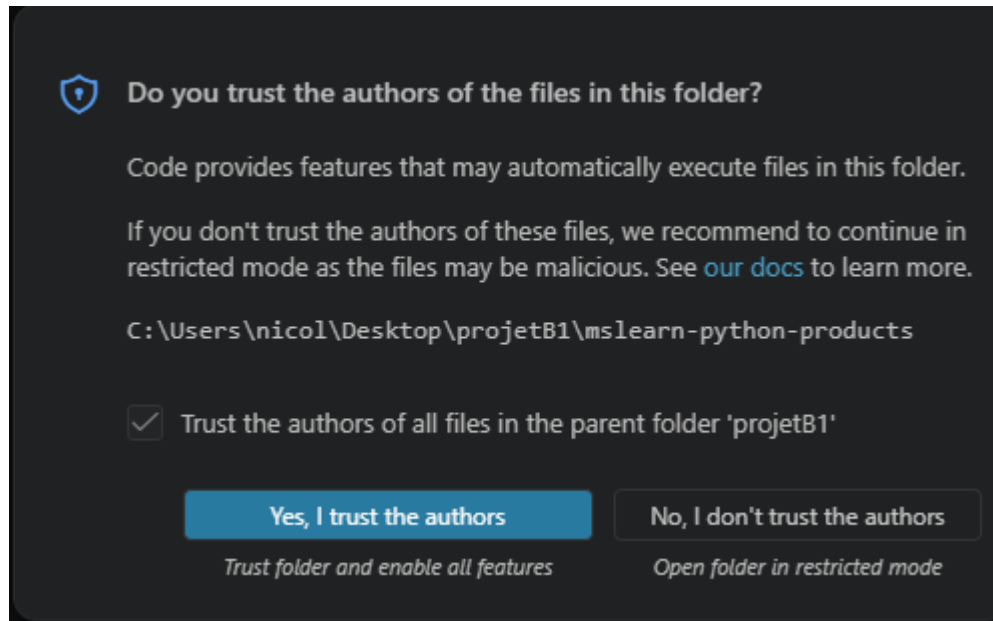
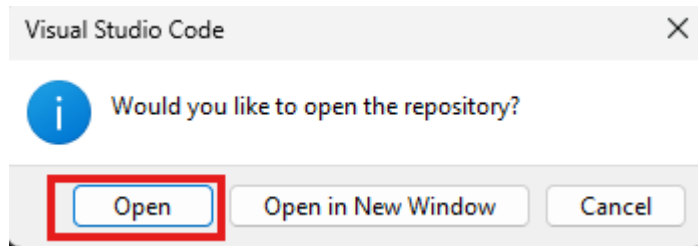


- Saisie de l'URL :

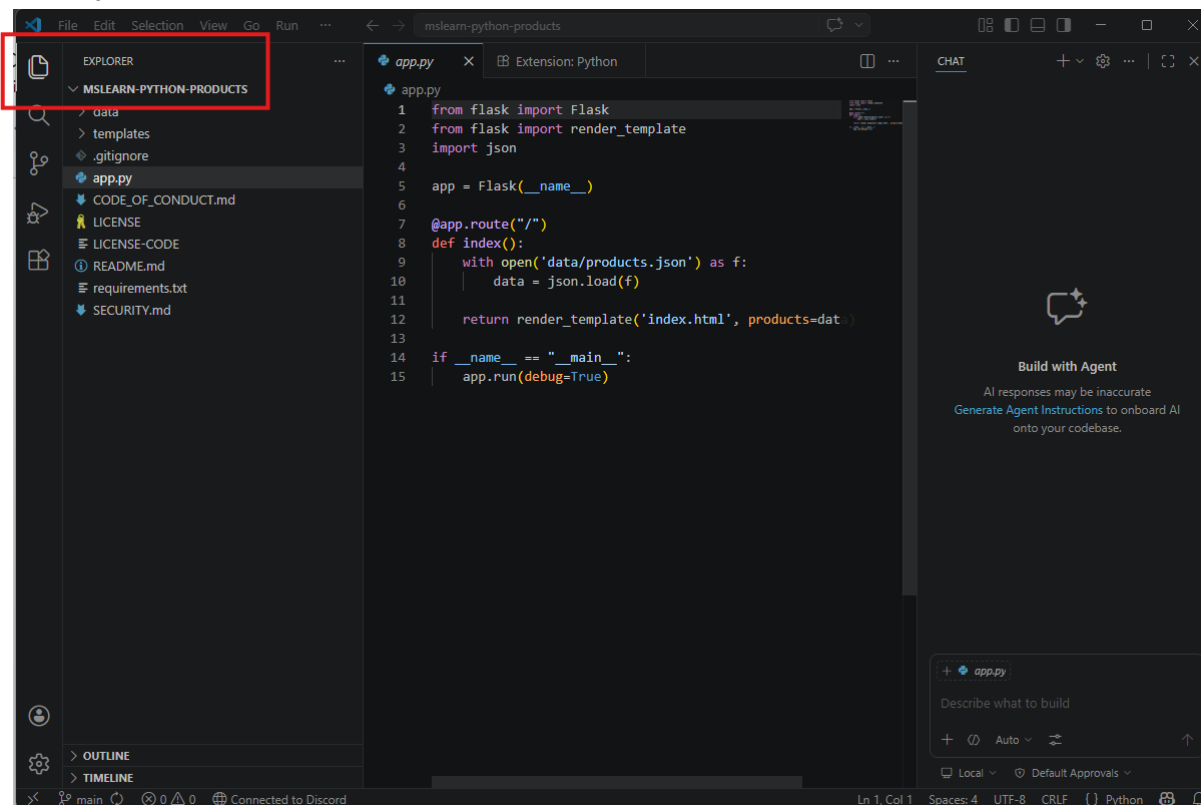


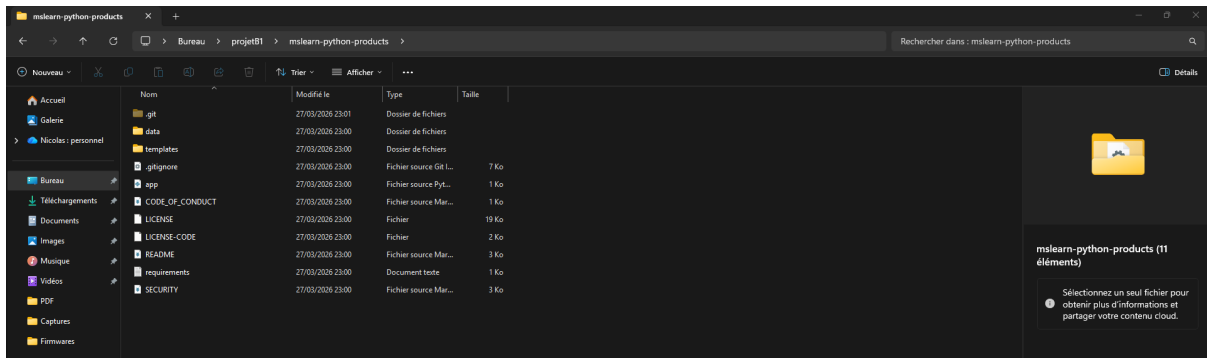
- Nous sélectionnons un emplacement libre pour le projet :



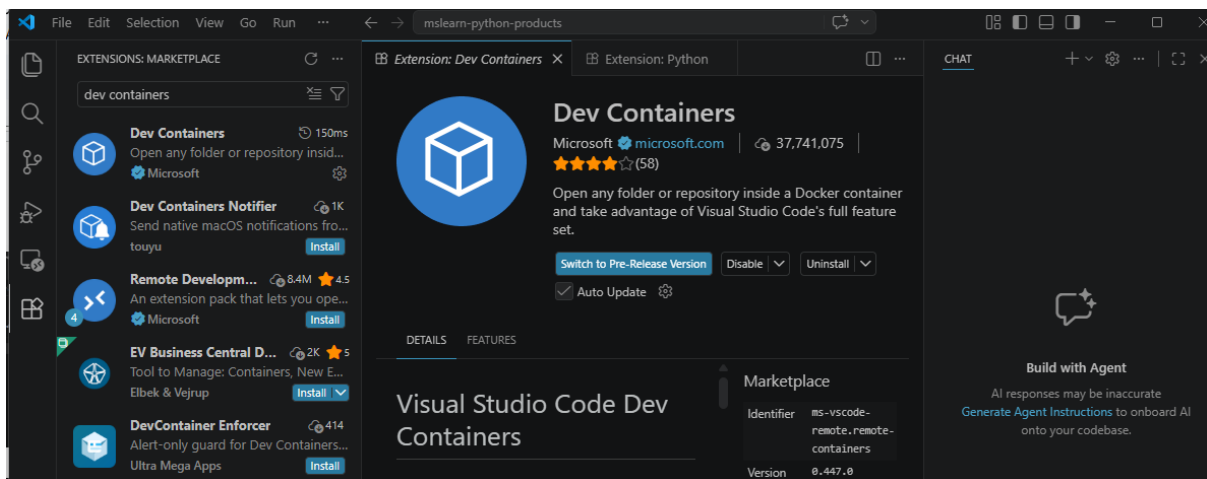
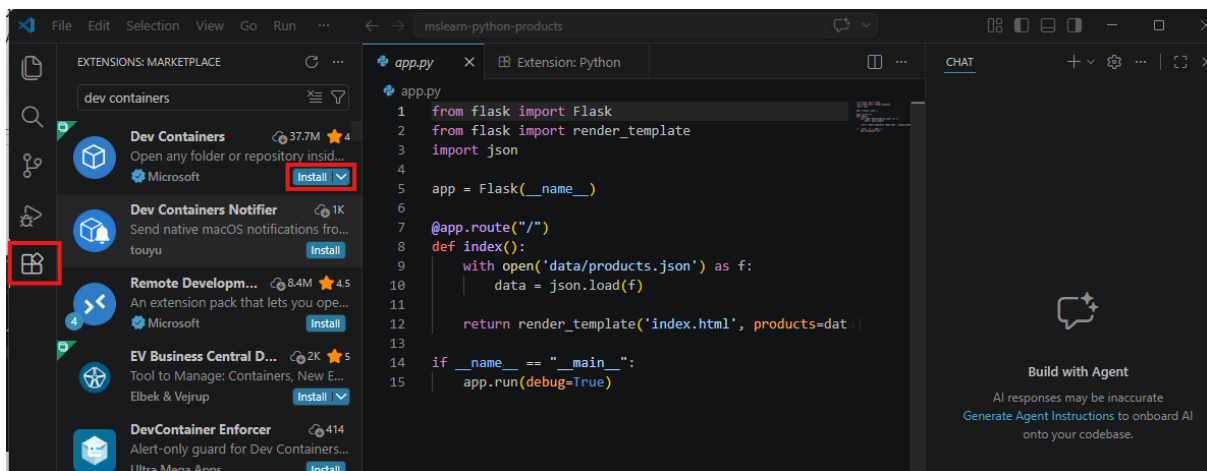


- Le projet est ouvert :

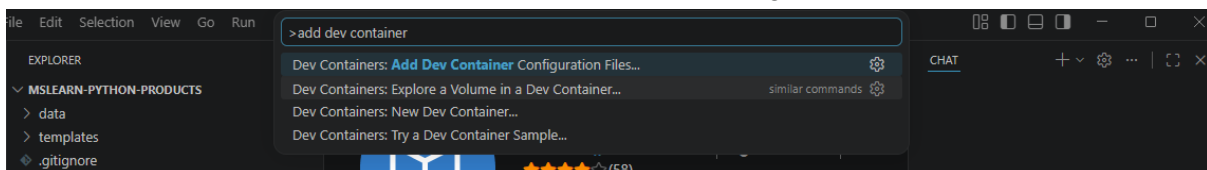




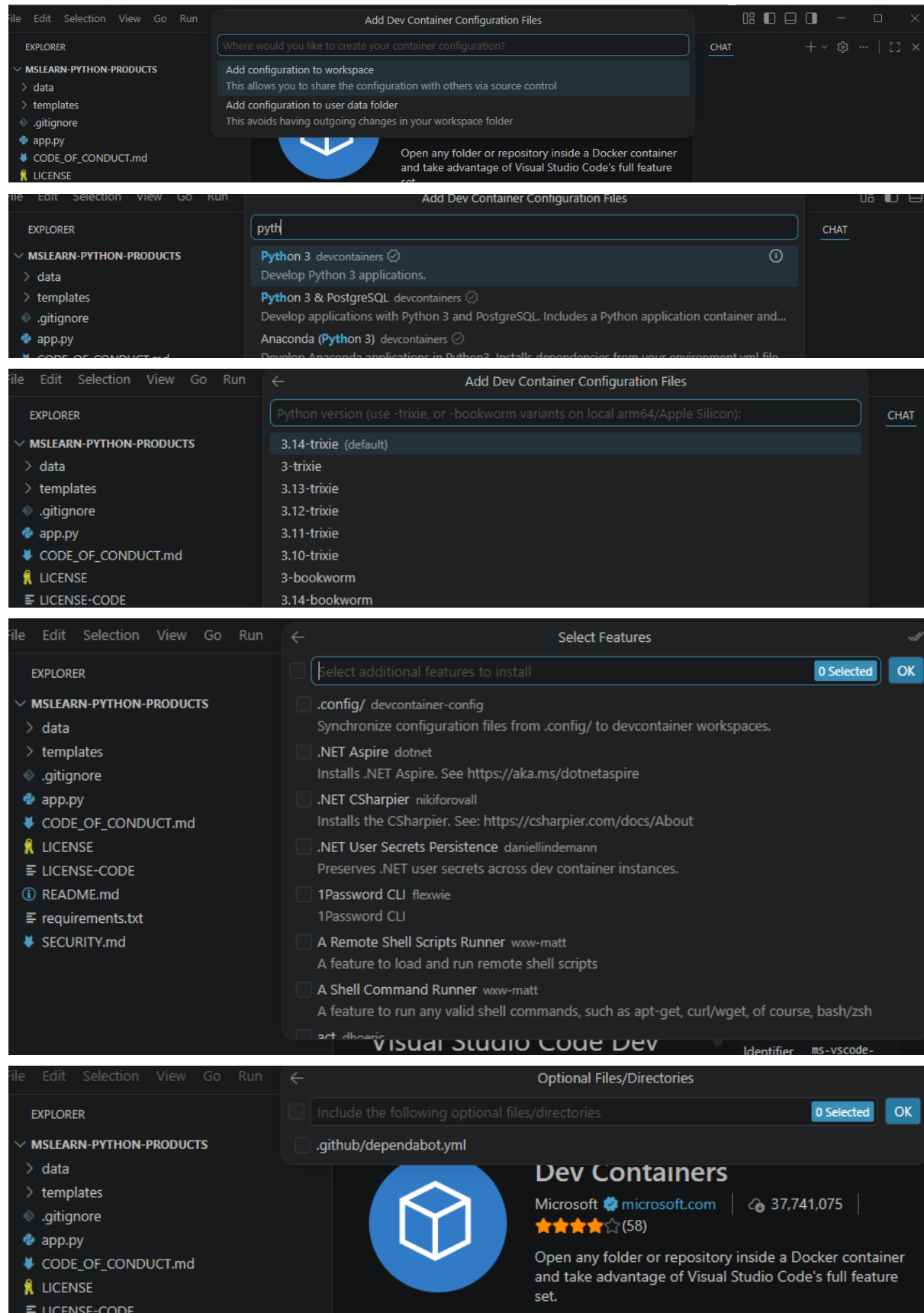
- Nous sélectionnons l'icône Extensions afin d'installer l'extension Dev Containers :



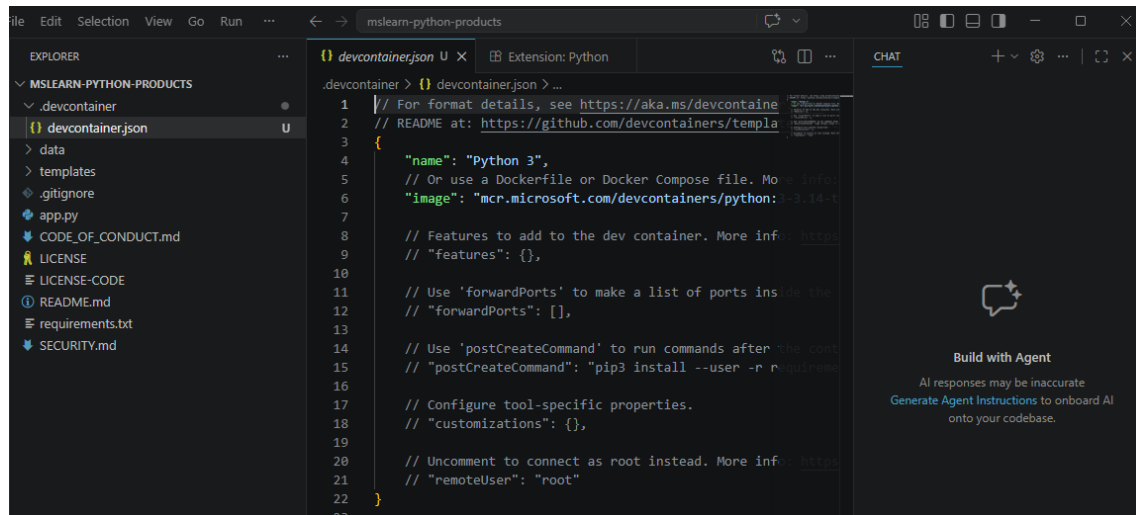
- On sélectionne Dev Containers: Add Dev Container Configuration Files.



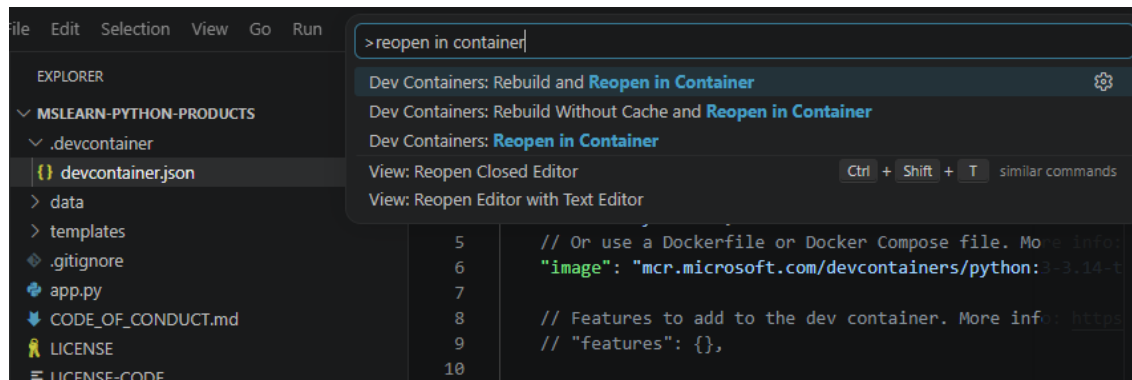
- On sélectionne ensuite Add configuration to workspace



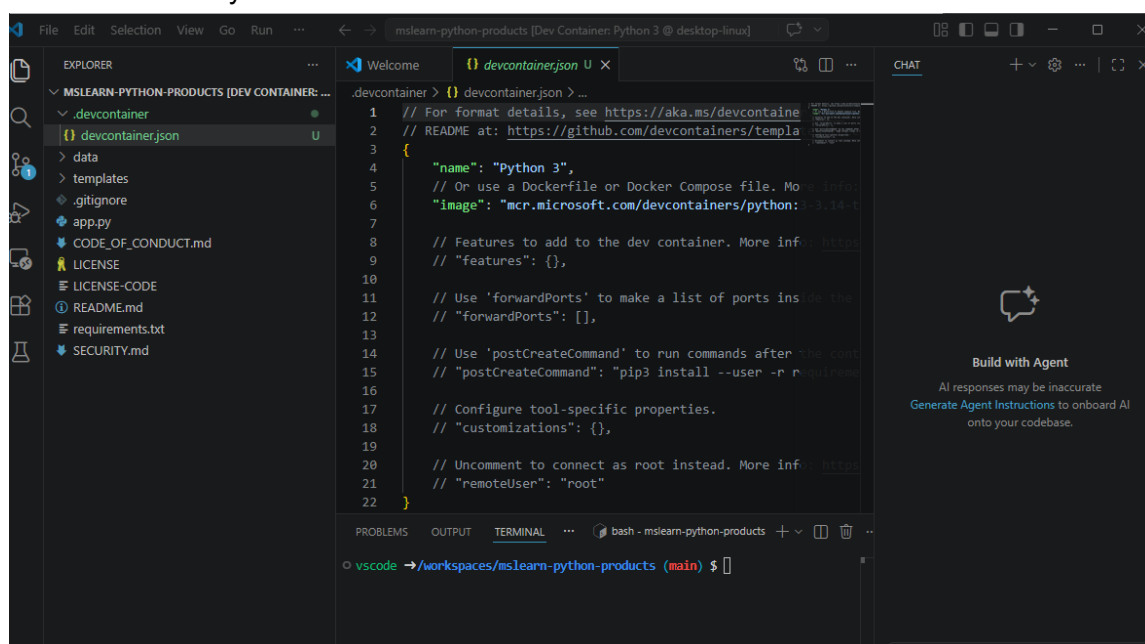
- Nous développons ce nouveau dossier contenant un fichier devcontainer.json.



- Nous tapons `reopen in container` et sélectionnons Dev Containers: Reopen in Container dans la liste des options disponibles.



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



- Nous exécutons la commande `python --version` pour vérifier que Python est installé et `pip3 install --user -r requirements.txt` :

```
TERMINAL  ...  bash - mslearn-python-products  +  -  [ ]  [X]  ...  |  [ ]  [X]  X

• vscode →/workspaces/mslearn-python-products (main) $ python --version
Python 3.14.3
• vscode →/workspaces/mslearn-python-products (main) $ pip3 install --user
-r requirements.txt
Collecting Flask==2.3.3 (from -r requirements.txt (line 2))
  Downloading flask-2.3.3-py3-none-any.whl.metadata (3.6 kB)
Collecting Werkzeug>=2.3.7 (from Flask==2.3.3->-r requirements.txt (line 2
))
  Downloading werkzeug-3.1.7-py3-none-any.whl.metadata (4.0 kB)
Collecting Jinja2>=3.1.2 (from Flask==2.3.3->-r requirements.txt (line 2))
  Downloading jinja2-3.1.6-py3-none-any.whl.metadata (2.9 kB)
Collecting itsdangerous>=2.1.2 (from Flask==2.3.3->-r requirements.txt (li
ne 2))
```

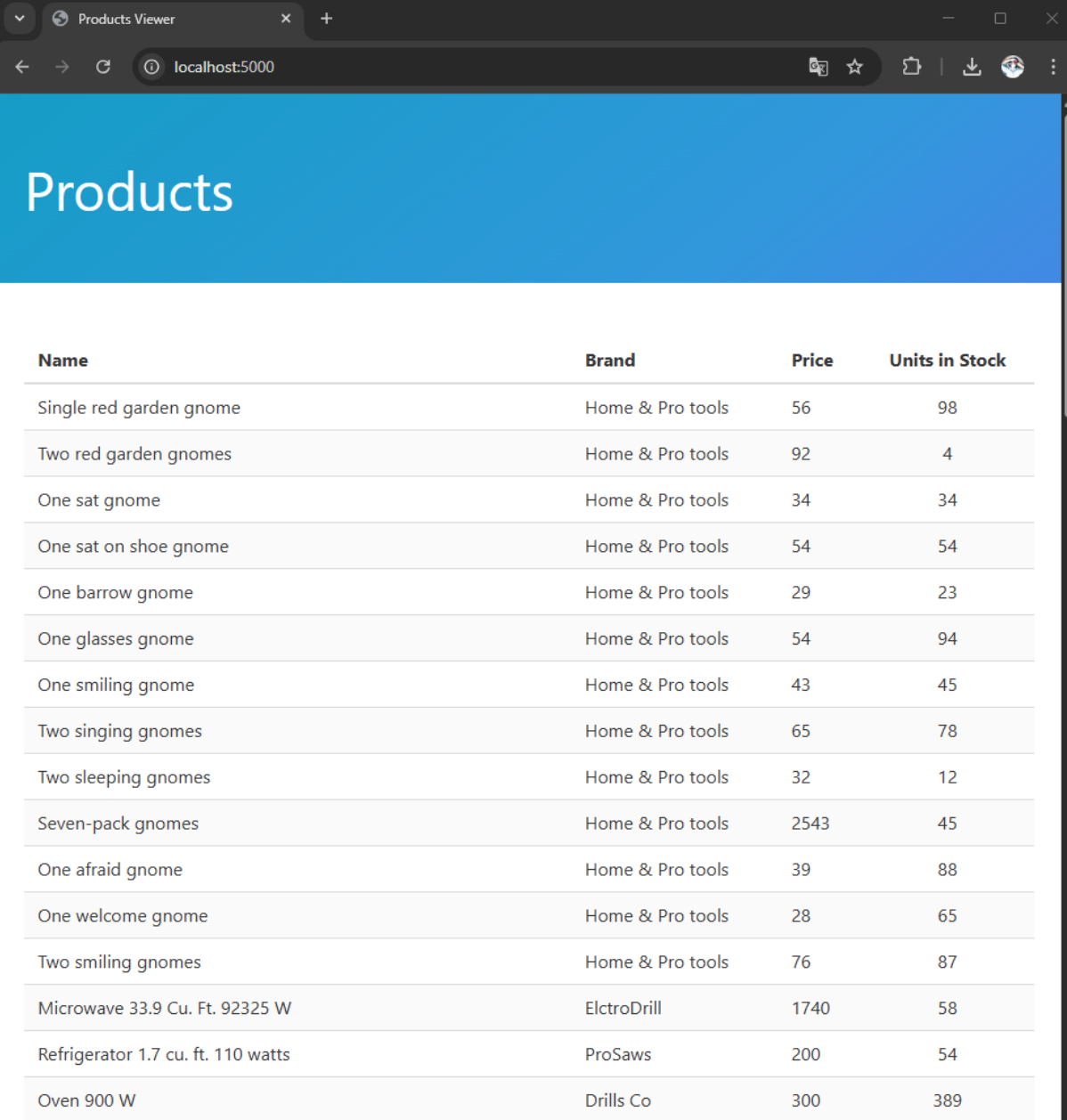
- Nous entrons la commande `python app.py` pour exécuter le projet :

```
PROBLEMS 2  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS 2  python - mslearn-python-products  +  -  [ ]  [X]  ...  |  [ ]  [X]  X

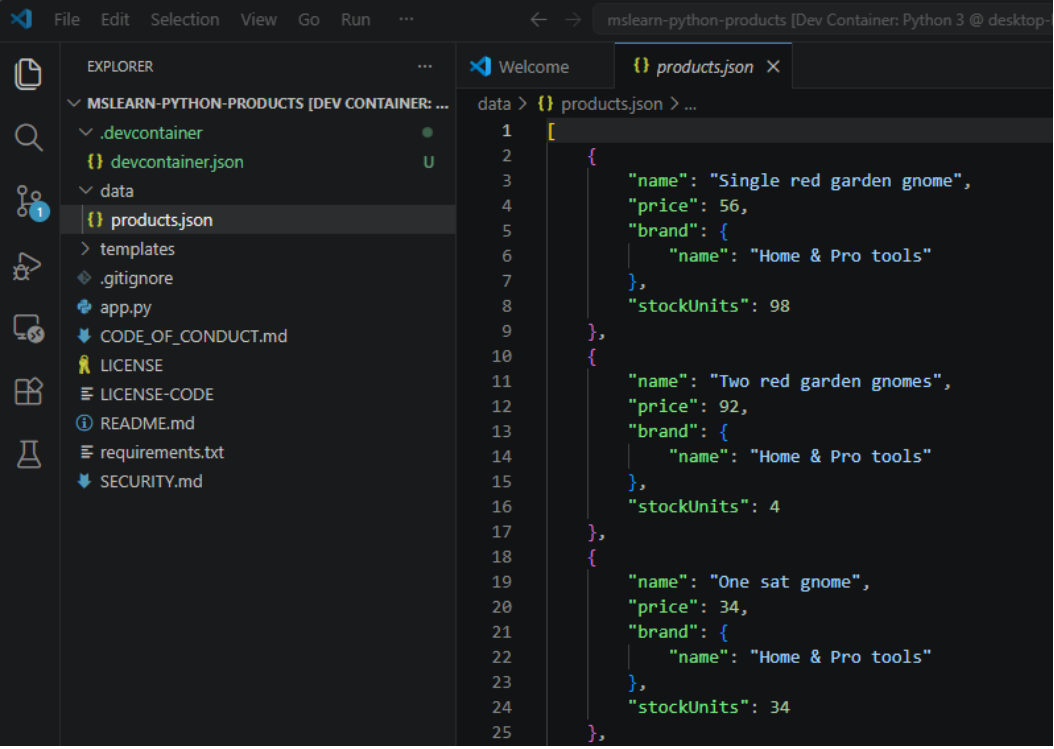
vscode →/workspaces/mslearn-python-products (main) $ pip3 install --user -r requirements.txt
○ vscode →/workspaces/mslearn-python-products (main) $ python app.py
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 120-910-722
127.0.0.1 - - [27/Mar/2026 22:18:26] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [27/Mar/2026 22:18:26] "GET /favicon.ico HTTP/1.1" 404 -
[]

0  2  2  Connected to Discord  Ln 1, Col 1  Spaces: 4  UTF-8  CRLF  [ ] Python  3.14.3  Python 3.14.3
```

- L'application web python s'est bien exécutée :

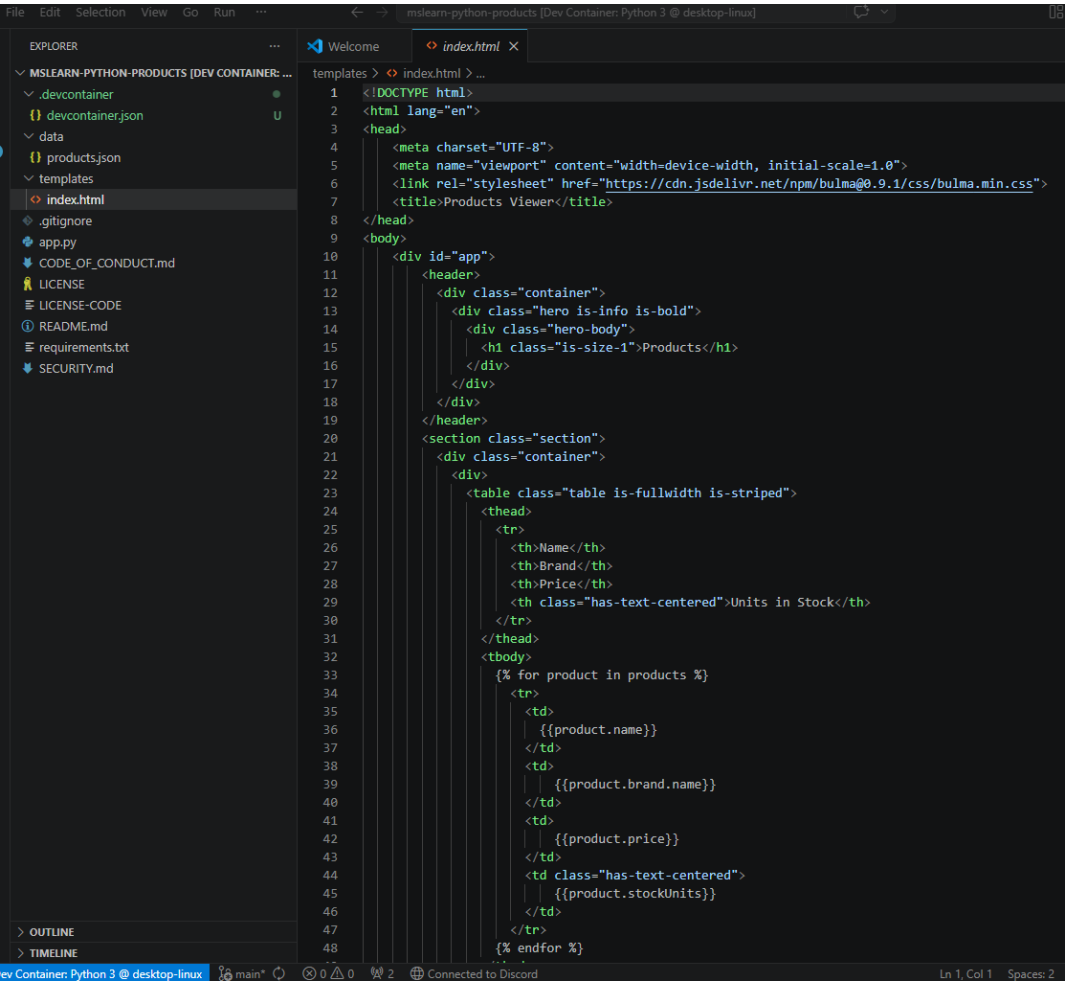


Name	Brand	Price	Units in Stock
Single red garden gnome	Home & Pro tools	56	98
Two red garden gnomes	Home & Pro tools	92	4
One sat gnome	Home & Pro tools	34	34
One sat on shoe gnome	Home & Pro tools	54	54
One barrow gnome	Home & Pro tools	29	23
One glasses gnome	Home & Pro tools	54	94
One smiling gnome	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



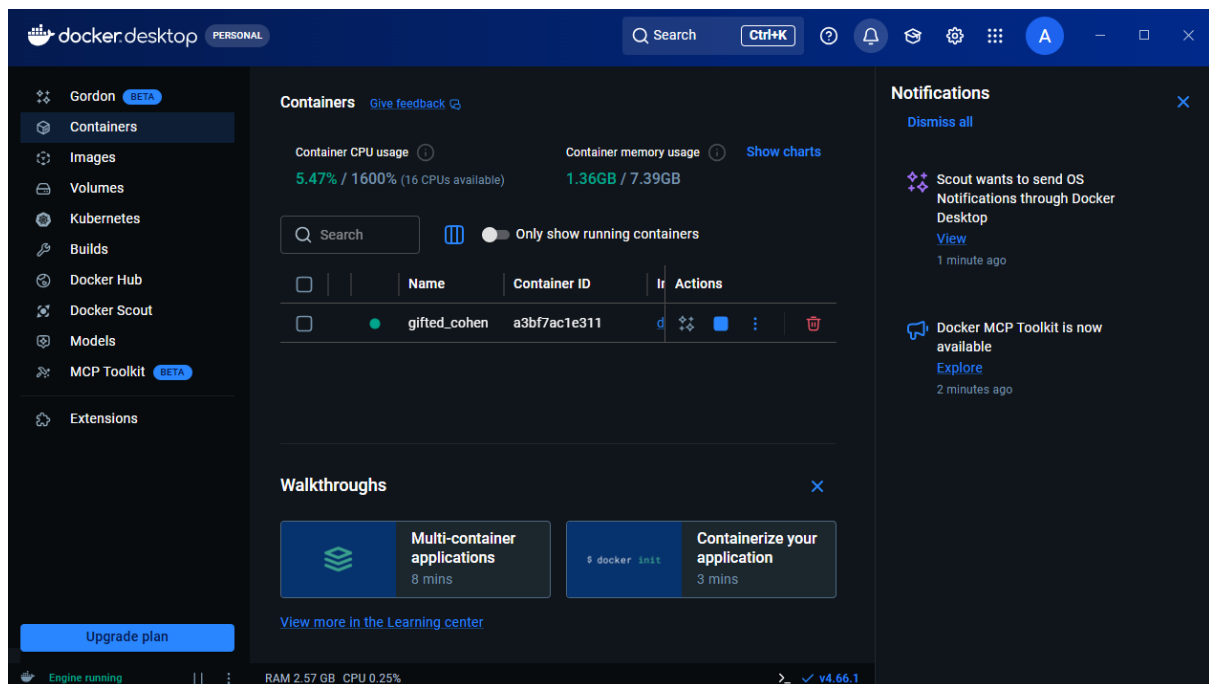
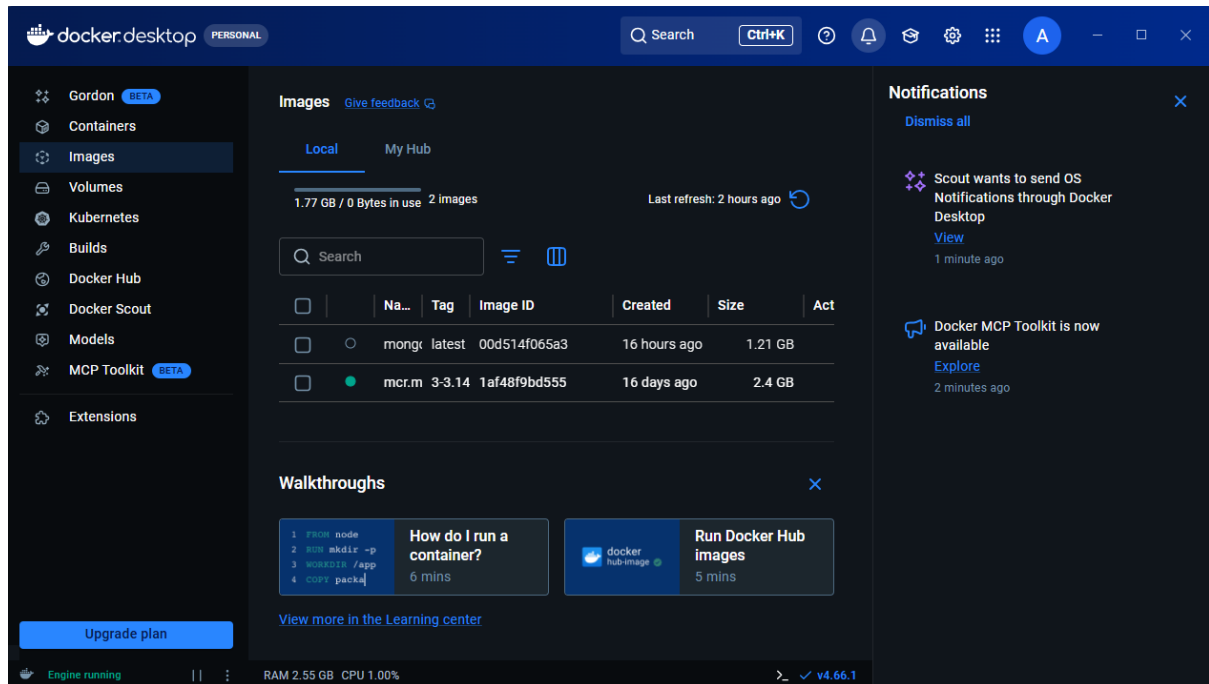
This screenshot shows the VS Code interface with a Dev Container for 'mslearn-python-products'. The Explorer sidebar on the left shows the file structure, including a 'data' folder containing 'products.json'. The main editor area displays the contents of 'products.json', which is a JSON array of three product objects. The first product is 'Single red garden gnome' with a price of 56 and 98 stock units. The second is 'Two red garden gnomes' with a price of 92 and 4 stock units. The third is 'One sat gnome' with a price of 34 and 34 stock units. All products are from the brand 'Home & Pro tools'.

```
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 ]
```



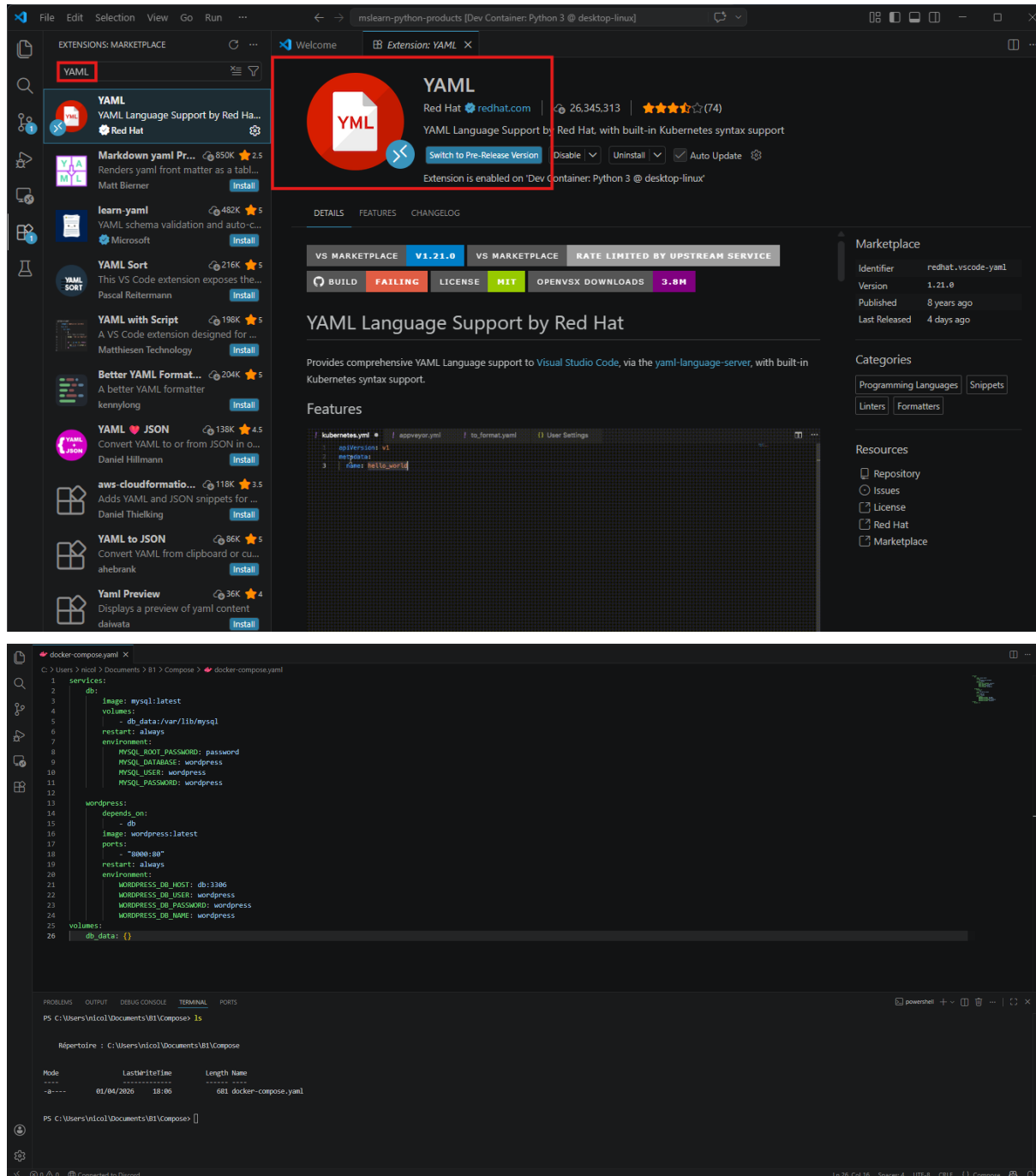
This screenshot shows the VS Code interface with the same Dev Container, but now displaying the 'index.html' file. The code is a basic HTML template using Bootstrap 5. It includes a header section with a hero area containing the heading 'Products' and a main section with a table of products. The table has columns for Name, Brand, Price, and Units in Stock. The table body uses a Jinja2 loop to iterate over the 'products' data from the 'products.json' file. The status bar at the bottom indicates the file is at line 1, column 1, with 2 spaces.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bulma@0.9.1/css/bulma.min.css">
7   <title>Products Viewer</title>
8 </head>
9 <body>
10   <div id="app">
11     <header>
12       <div class="container">
13         <div class="hero is-info is-bold">
14           <div class="hero-body">
15             <h1 class="is-size-1">Products</h1>
16           </div>
17         </div>
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               {% for product in products %}
34                 <tr>
35                   <td>
36                     {{product.name}}
37                   </td>
38                   <td>
39                     {{product.brand.name}}
40                   </td>
41                   <td>
42                     {{product.price}}
43                   </td>
44                   <td class="has-text-centered">
45                     {{product.stockUnits}}
46                   </td>
47                 </tr>
48               {% endfor %}
```

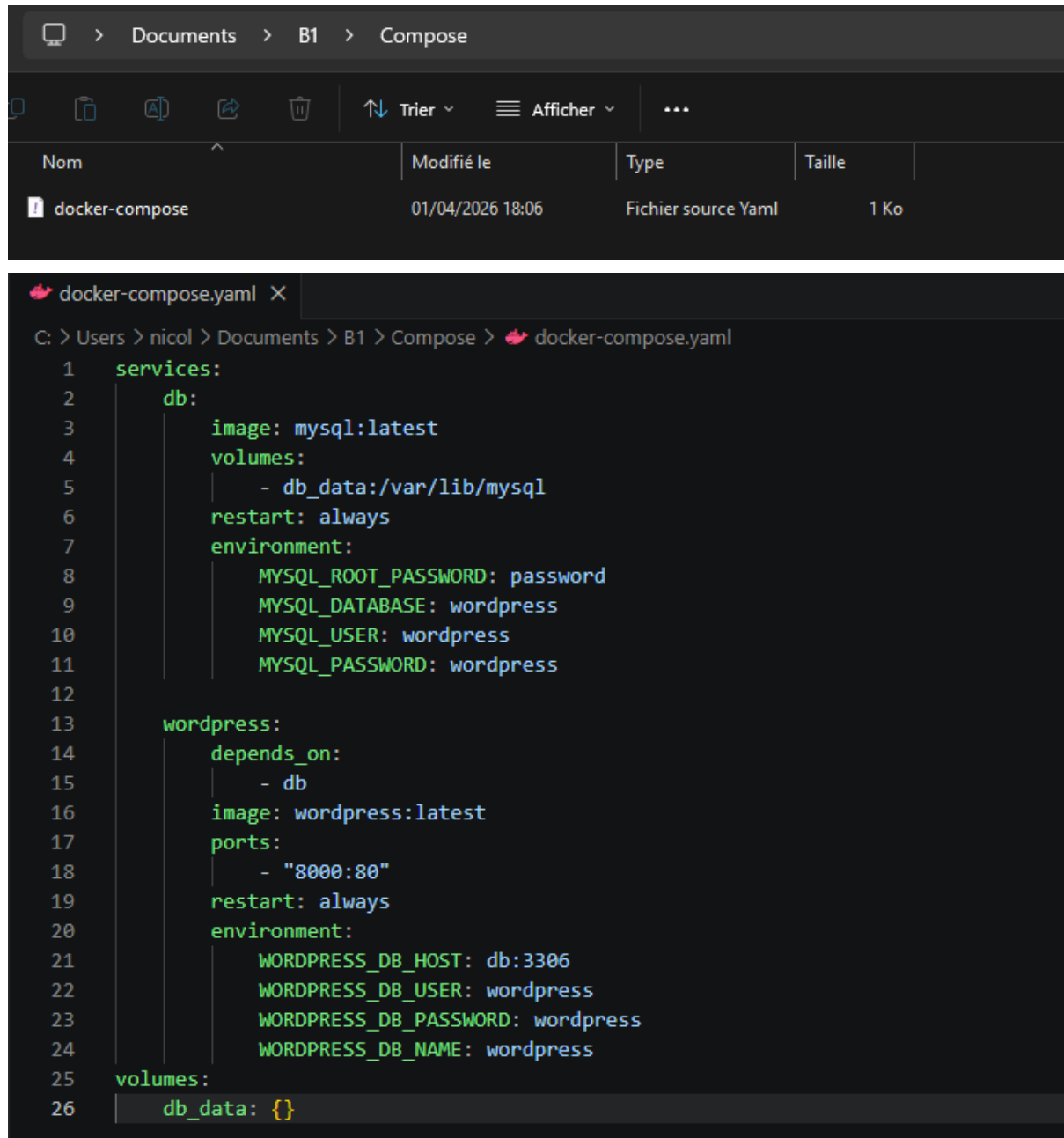


13. Monter une stack applicative avec Docker Compose.

- Installation de l'extension YAML dans VS Code :



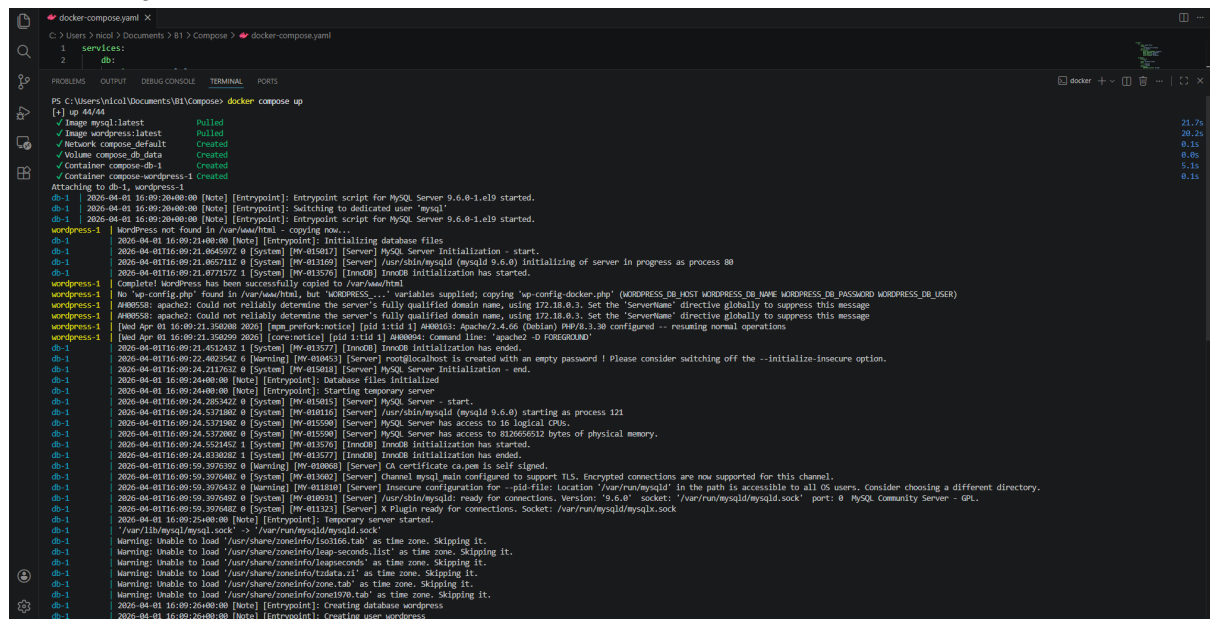
- Création du fichier docker-compose.yml :



The image shows a file explorer window at the top and a code editor window below it. The file explorer shows the path 'Documents > B1 > Compose' and a file named 'docker-compose' with a size of 1 Ko. The code editor shows the content of 'docker-compose.yml' with the following YAML configuration:

```
1  services:
2      db:
3          image: mysql:latest
4          volumes:
5              - db_data:/var/lib/mysql
6          restart: always
7          environment:
8              MYSQL_ROOT_PASSWORD: password
9              MYSQL_DATABASE: wordpress
10             MYSQL_USER: wordpress
11             MYSQL_PASSWORD: wordpress
12
13     wordpress:
14         depends_on:
15             - db
16         image: wordpress:latest
17         ports:
18             - "8000:80"
19         restart: always
20         environment:
21             WORDPRESS_DB_HOST: db:3306
22             WORDPRESS_DB_USER: wordpress
23             WORDPRESS_DB_PASSWORD: wordpress
24             WORDPRESS_DB_NAME: wordpress
25 volumes:
26     db_data: {}
```

- Nous lançons la stack applicative avec la commande docker compose up :



```
PS C:\Users\nicol\Documents> docker compose up
[+] Up All 0/0
✓ Image mysql:latest Pulled
✓ Image wordpress:latest Pulled
✓ Network compose_default Created
✓ Volume compose_db_data Created
✓ Container compose-db-1 Created
✓ Container compose-wordpress-1 Created
Attaching to db-1, wordpress-1
db-1 | 2026-04-01 16:09:20:40:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 9.6.0-1.el9 started.
db-1 | 2026-04-01 16:09:20:40:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
db-1 | 2026-04-01 16:09:20:40:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 9.6.0-1.el9 started.
wordpress-1 | WordPress not found in /var/www/html - copying now...
db-1 | 2026-04-01 16:09:21:40:00 [Note] [Entrypoint]: Initializing database files
db-1 | 2026-04-01 16:09:21:40:00 [System] [NW-015017] [Server] MySQL Server Initialization - start.
db-1 | 2026-04-01 16:09:21:40:00 [System] [NW-013169] [Server] /usr/sbin/mysqld (mysqld 9.6.0) Initializing server in progress as process 80
db-1 | 2026-04-01 16:09:21:40:00 [System] [NW-013170] [InnoDB] InnoDB Initialization has started.
wordpress-1 | Complete! WordPress has been successfully copied to /var/www/html
wordpress-1 | No 'wp-config.php' found in /var/www/html, but 'WORDPRESS...' variables supplied; copying 'wp-config-docker.php' (WORDPRESS_DB_HOST WORDPRESS_DB_NAME WORDPRESS_DB_PASSWORD WORDPRESS_DB_USER)
wordpress-1 | #480550: 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 message
wordpress-1 | #480550: 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 message
wordpress-1 | [Wed Apr 01 16:09:21.350080 2026] [warn:prefork:notice] [pid 1:tid 1] #480163: Apache/2.4.66 (Debian) PHP/8.3.30 configured -- resuming normal operations
wordpress-1 | [Wed Apr 01 16:09:21.350080 2026] [core:notice] [pid 1:tid 1] #480806: Command Line: 'apache2 -D FOREGROUND'
db-1 | 2026-04-01 16:09:21:40:00 [System] [NW-013177] [InnoDB] InnoDB Initialization has ended.
db-1 | 2026-04-01 16:09:22:40:00 [Warning] [NW-010453] [Server] root@localhost is created with an empty password ! Please consider switching off the --initialize-insecure option.
db-1 | 2026-04-01 16:09:24:21:00 [System] [NW-010418] [Server] MySQL Server Initialization - end.
db-1 | 2026-04-01 16:09:24:40:00 [Note] [Entrypoint]: Database files initialized
db-1 | 2026-04-01 16:09:24:40:00 [Note] [Entrypoint]: Starting temporary server
db-1 | 2026-04-01 16:09:24:40:00 [System] [NW-015015] [Server] MySQL Server - start.
db-1 | 2026-04-01 16:09:24:53:7180 [System] [NW-010116] [Server] /usr/sbin/mysqld (mysqld 9.6.0) starting as process 121
db-1 | 2026-04-01 16:09:24:53:7180 [System] [NW-015090] [Server] MySQL Server has access to 16 logical CPUs.
db-1 | 2026-04-01 16:09:24:53:7200 [System] [NW-015090] [Server] MySQL Server has access to 812606552 bytes of physical memory.
db-1 | 2026-04-01 16:09:24:53:2452 [System] [NW-013170] [InnoDB] InnoDB Initialization has started.
db-1 | 2026-04-01 16:09:24:41:30282 [System] [NW-013177] [InnoDB] InnoDB Initialization has ended.
db-1 | 2026-04-01 16:09:25:20:382 [Warning] [NW-010600] [Server] CA certificate ca.pem is self signed.
db-1 | 2026-04-01 16:09:25:20:382 [System] [NW-013401] [Server] Channel mysql main configured to support TLS. Encrypted connections are now supported for this channel.
db-1 | 2026-04-01 16:09:25:20:382 [Warning] [NW-011810] [Server] Insecure configuration for --pid-file: Location '/var/run/mysqld/' in the path is accessible to all OS users. Consider choosing a different directory.
db-1 | 2026-04-01 16:09:25:20:382 [System] [NW-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '9.6.0' socket: '/var/run/mysqld/mysqld.sock' port: 0 MySQL Community Server - GPL.
db-1 | 2026-04-01 16:09:25:20:382 [System] [NW-011152] [Server] * Plugin ready for connections. Socket: /var/run/mysqld/mysqld.sock
db-1 | 2026-04-01 16:09:25:40:00 [Note] [Entrypoint]: Temporary server started.
db-1 | /var/lib/mysql/mysql.sock -> /var/run/mysqld/mysqld.sock
db-1 | Warning: Unable to load '/usr/share/zoneinfo/iso3166.tab' as time zone. Skipping it.
db-1 | Warning: Unable to load '/usr/share/zoneinfo/leap-seconds.list' as time zone. Skipping it.
db-1 | Warning: Unable to load '/usr/share/zoneinfo/leapseconds' as time zone. Skipping it.
db-1 | Warning: Unable to load '/usr/share/zoneinfo/leapseconds' as time zone. Skipping it.
db-1 | Warning: Unable to load '/usr/share/zoneinfo/leapseconds' as time zone. Skipping it.
db-1 | Warning: Unable to load '/usr/share/zoneinfo/zone.tab' as time zone. Skipping it.
db-1 | Warning: Unable to load '/usr/share/zoneinfo/zone.tab' as time zone. Skipping it.
db-1 | 2026-04-01 16:09:26:40:00 [Note] [Entrypoint]: Creating database wordpress
db-1 | 2026-04-01 16:09:26:40:00 [Note] [Entrypoint]: Creating user wordpress
```

- Nous accédons au blog en saisissant l'url <http://localhost:8000> :



