NAME: **AMAN AGRAWAL**
ROLL NO: **G24AIT172**
SUB: **ML-OPs**
TASK : **ASSIGNMENT-2 (Docker Fundamentals)**

Docker Repository: https://hub.docker.com/repository/docker/aman007agg/docker-assignment-python-hello/general

1. Introduction:

This report describes the process of setting up a custom Docker image containing a Python script. The script prints a message with my roll number **(G24AIT172)** when executed inside a container.

The goal is to pull an official Python image, customize it with a simple script, build a Docker image, and push it to Docker Hub.

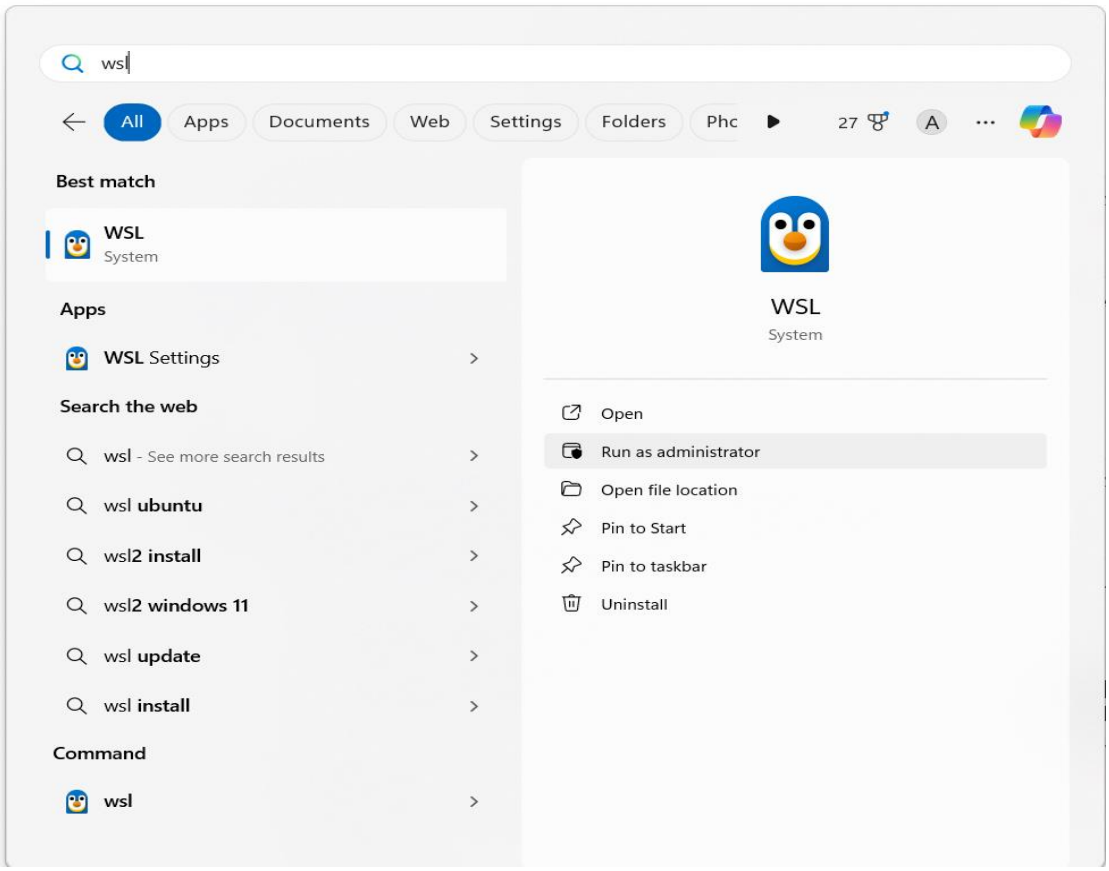
2. Setup Process

2.1. Install Docker Desktop on Windows.

Pre-requisite:

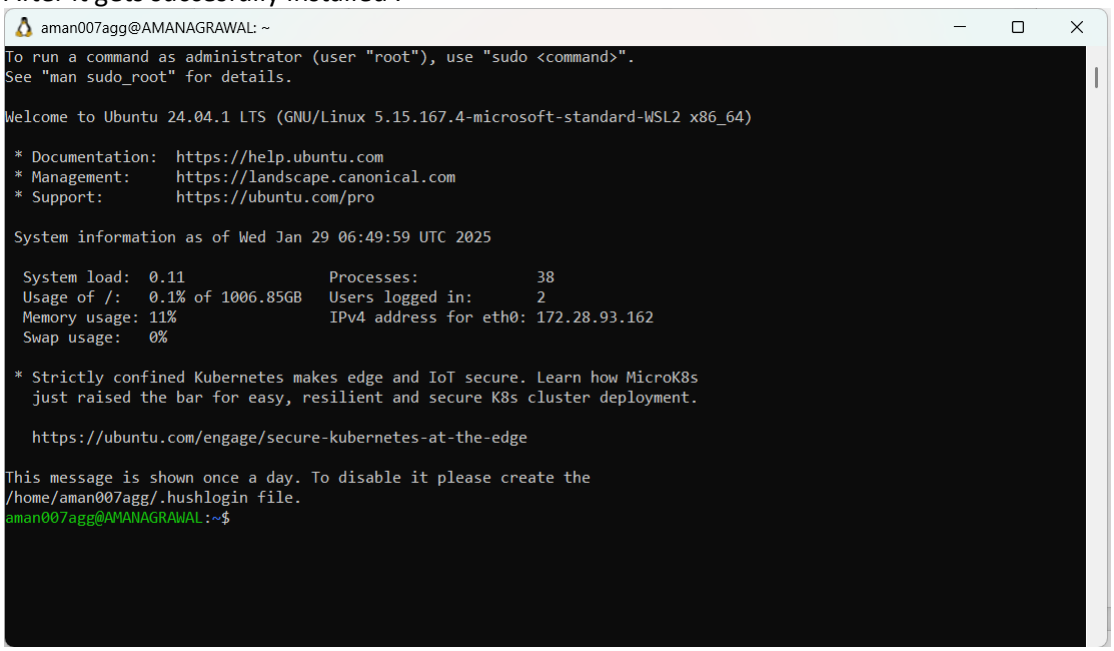
(A.) Install WSL (Windows Subsystem for Linux)-WSL

Search for WSL on your local system , and click run as administrator



Run the command: **wsl.exe --install**

After it gets succesfully installed .



(B.) To set WSL2 as the default version for future installations:
Run the command: **wsl --set-default-version 2**

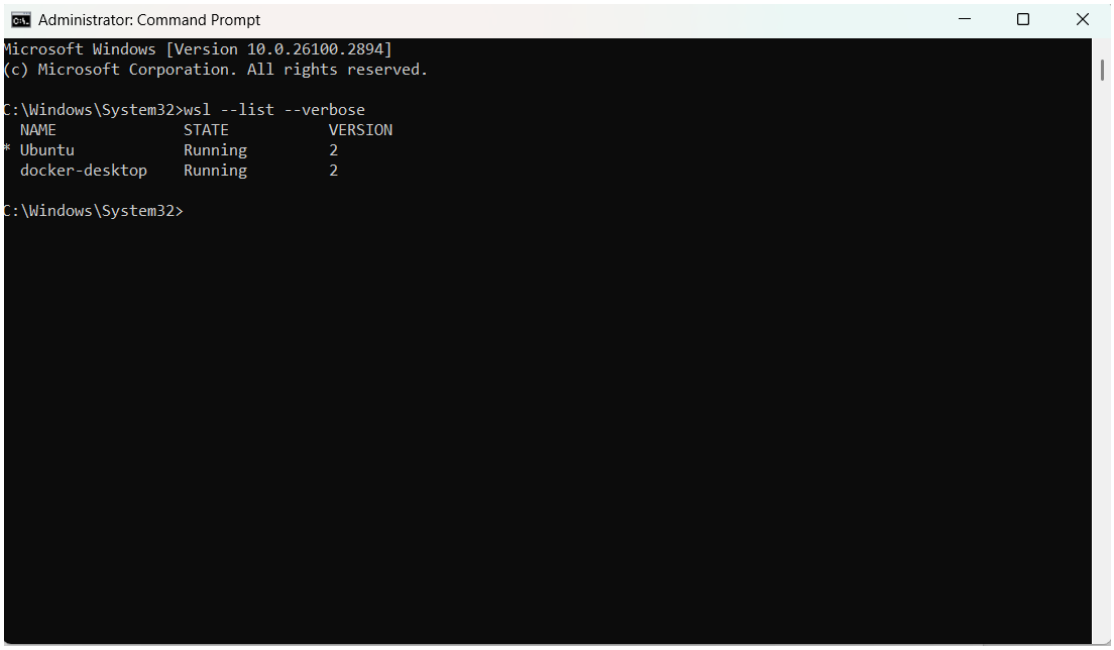
Kindly Note:
When prompted to create a UNIX username during the WSL (Windows Subsystem for Linux) installation process, you simply need to:

1. Enter a preferred username. This username does not need to match your Windows username, but it will be your default username for logging into the WSL environment.
2. Press **Enter**.
3. You'll then be asked to create a password for this user. Ensure the password is something you'll remember, as it will be needed to authenticate commands requiring elevated privileges (via sudo).

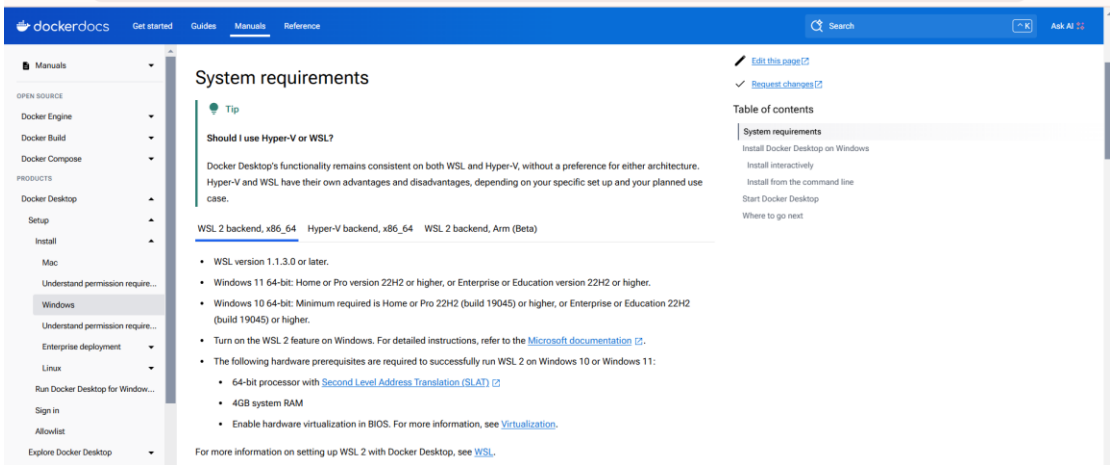
Enter new UNIX username: yourusername
New password: ********
Retype new password: ********

(C.) To check the version of WSL you are using, you can run:
Run the command: **wsl --list --verbose**

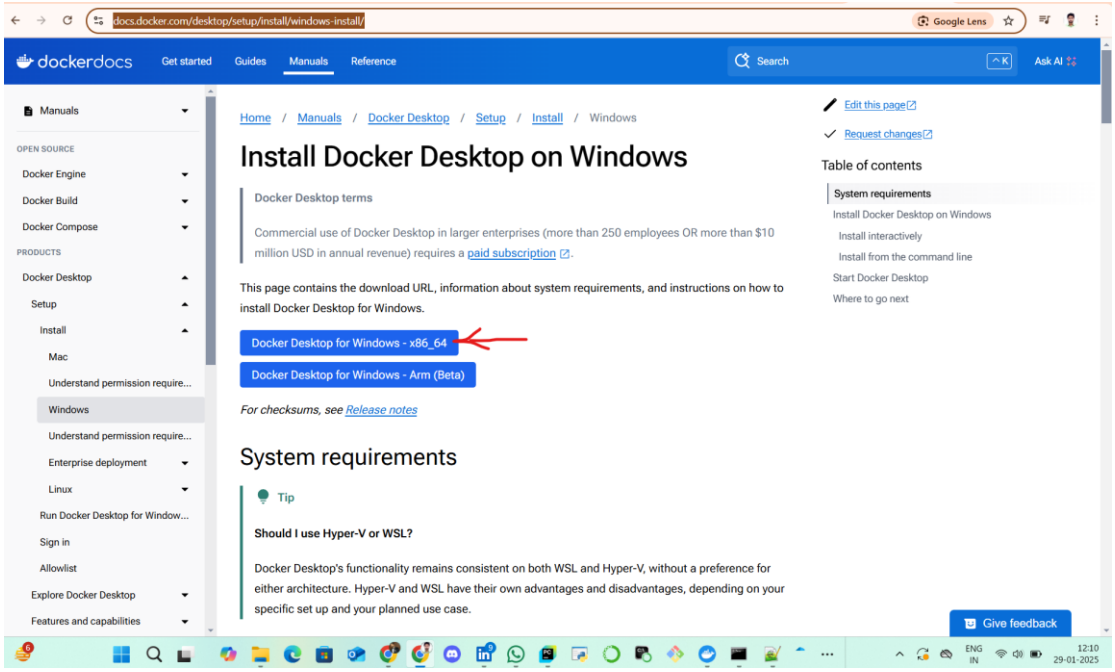
You can run the command on cmd as an administrator mode



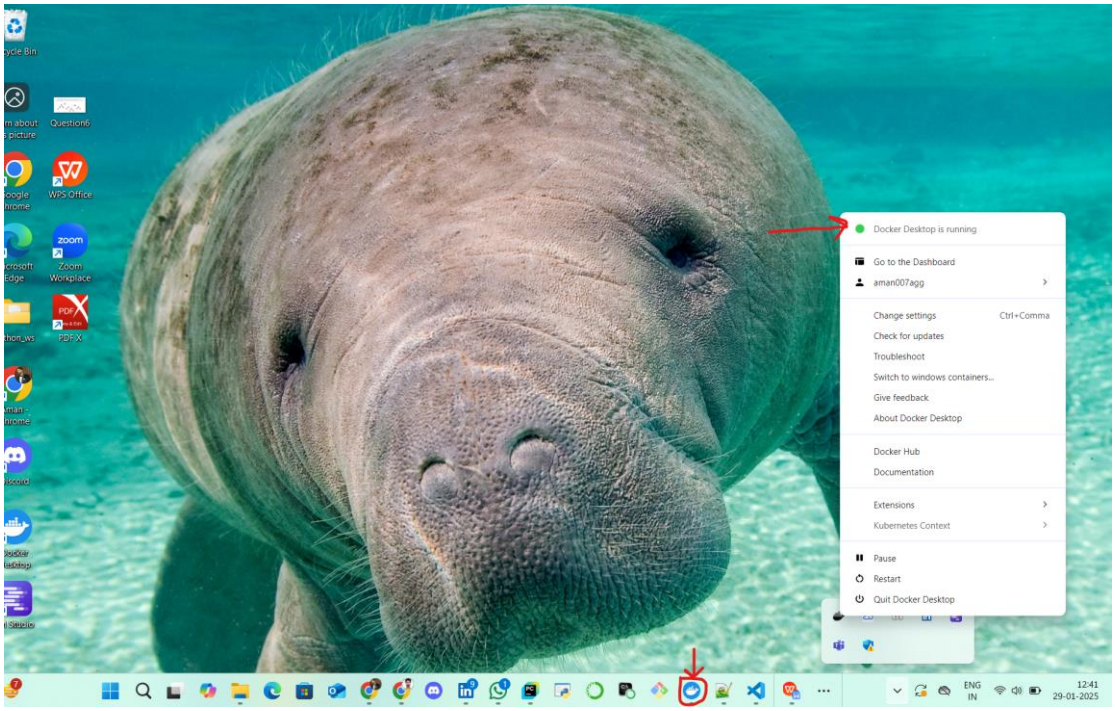
(D.) You can check the system requirements before setting up the Docker Desktop for windows.



(E.) Go to the specfied url and download the Docker Desktop on Windows-
https://docs.docker.com/desktop/setup/install/windows-install/

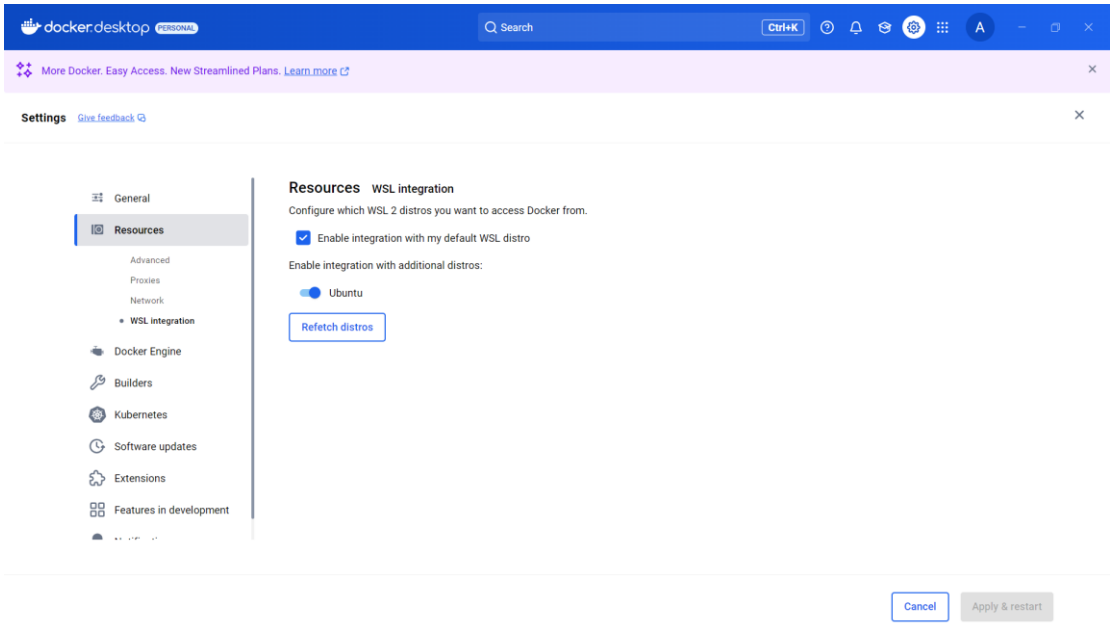


(F.) Once it gets successfully installed, make sure that docker is in running mode.



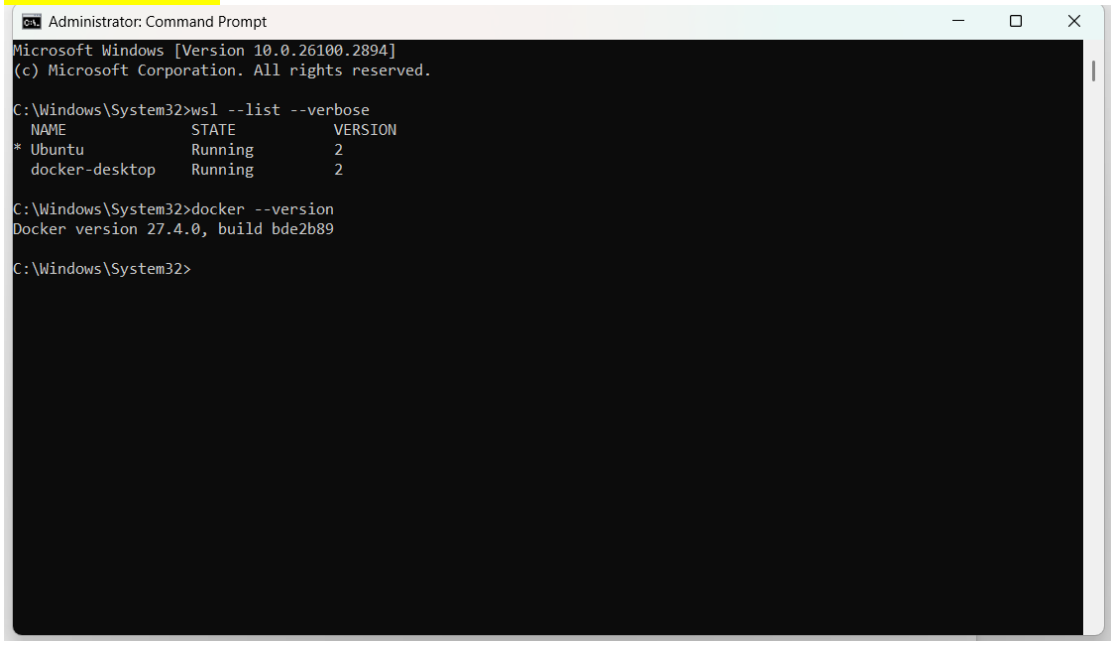
(G.) Enable WSL 2 Integration in Docker: During the installation, Docker Desktop will ask you to enable WSL 2 integration. Make sure that you check the option to use WSL 2 as the backend for Docker. Docker will automatically set this up for you.

- Once Docker Desktop is installed, open Docker Desktop from the Start menu.
- Go to **Settings** (gear icon) > **Resources** > **WSL Integration**.
- Ensure that **Enable integration with my default WSL distro** is checked. You can also enable integration with other WSL distributions if you have more than one.



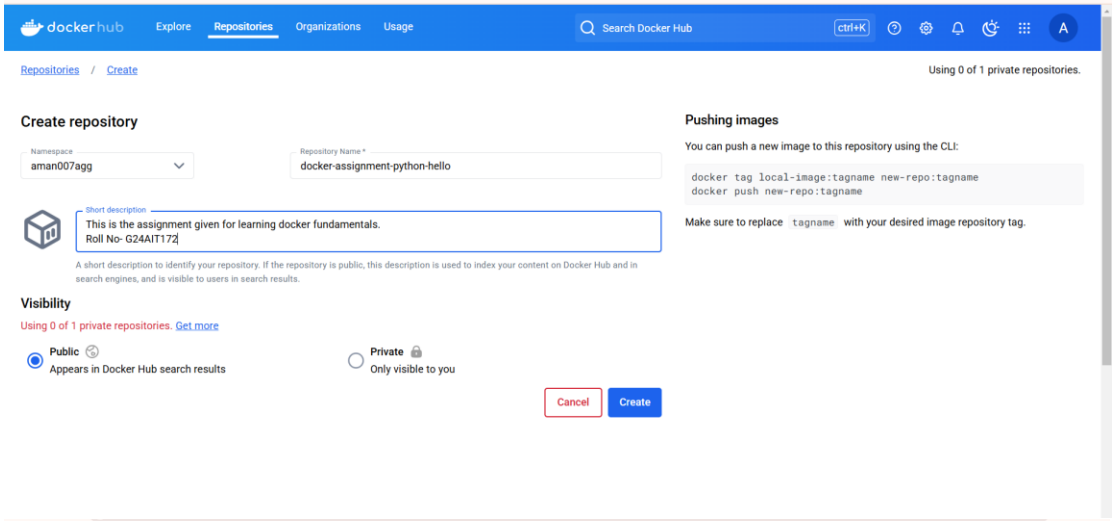
(H.) Test Docker Installation: Once Docker Desktop is running, open a command prompt or PowerShell window and run the following command to verify Docker is installed and running:

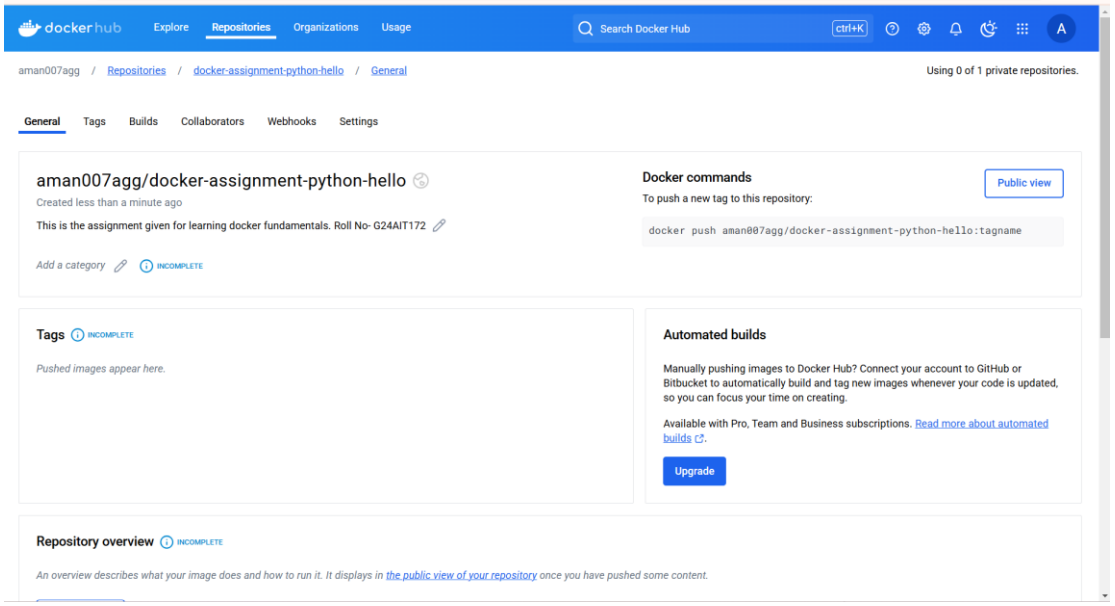
docker --version



3. Step-by-Step Execution

3.1 Create a new docker hub repository-



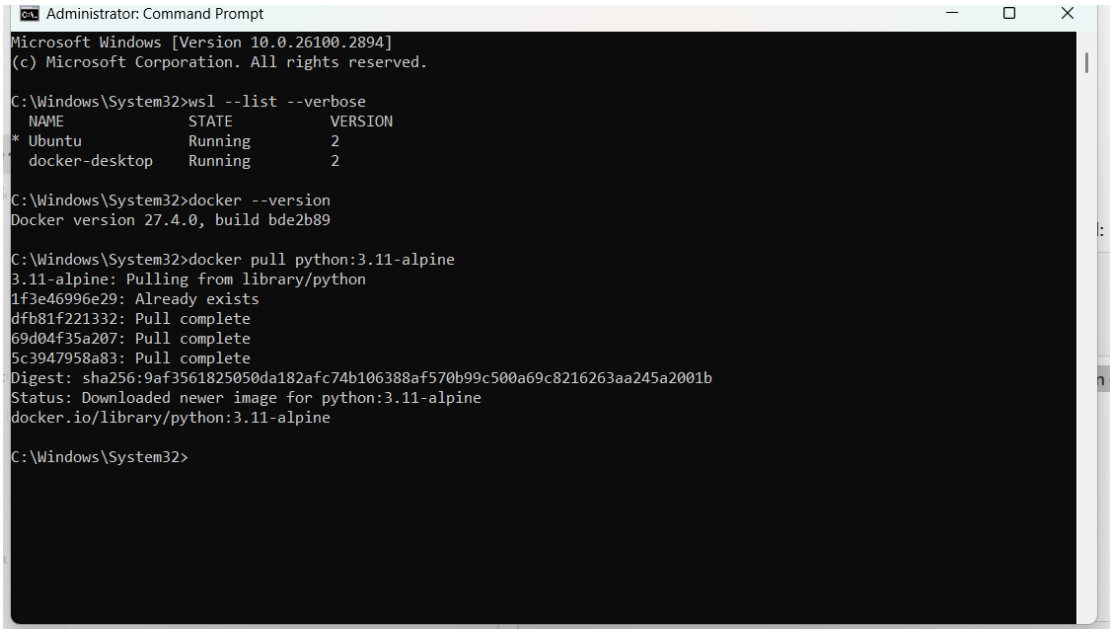


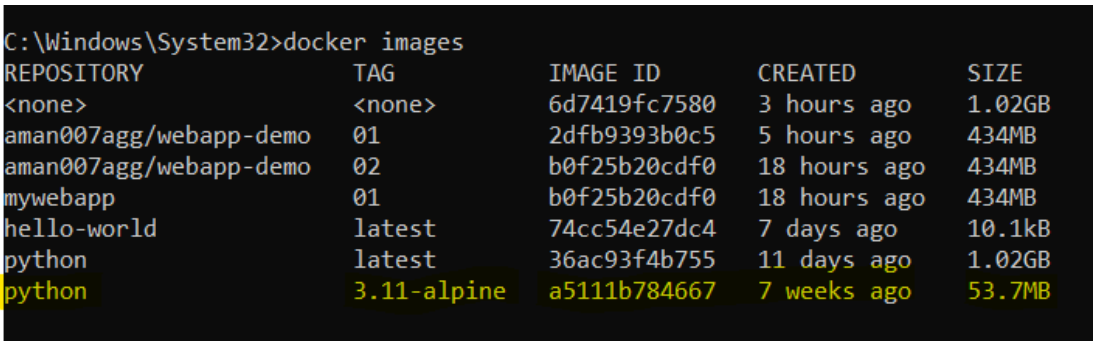
Pull command - *docker pull aman007agg/docker-assignment-python-hello*

A. Pull the official Python image, python:-alpine (the smallest version).

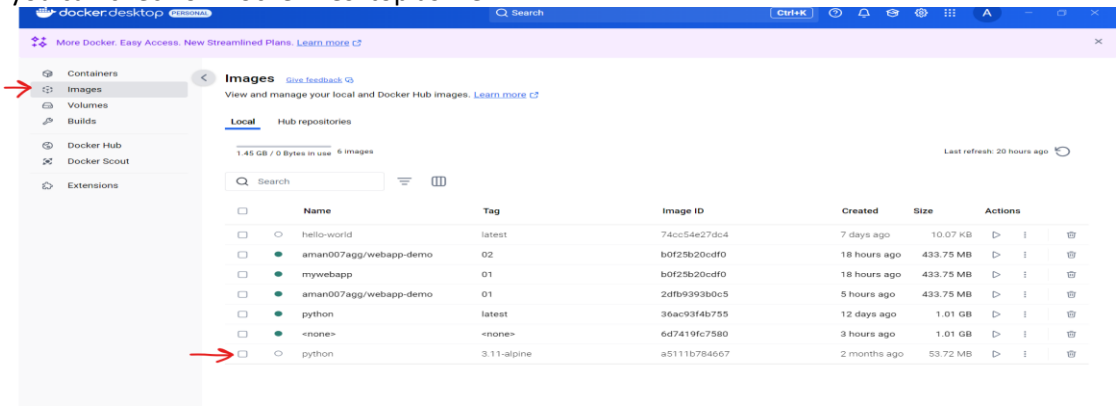
docker pull python:3.11-alpine

The Alpine version is chosen because it is a lightweight Linux distribution optimized for containers.





you can check on Docker Desktop as well



B. Build a custom Docker image by creating a directory, and adding a hello.py file that prints "Hello from the container, Your Roll No.(replace this with your roll no.)". Use a Dockerfile to create the image.

1. Create a Directory -

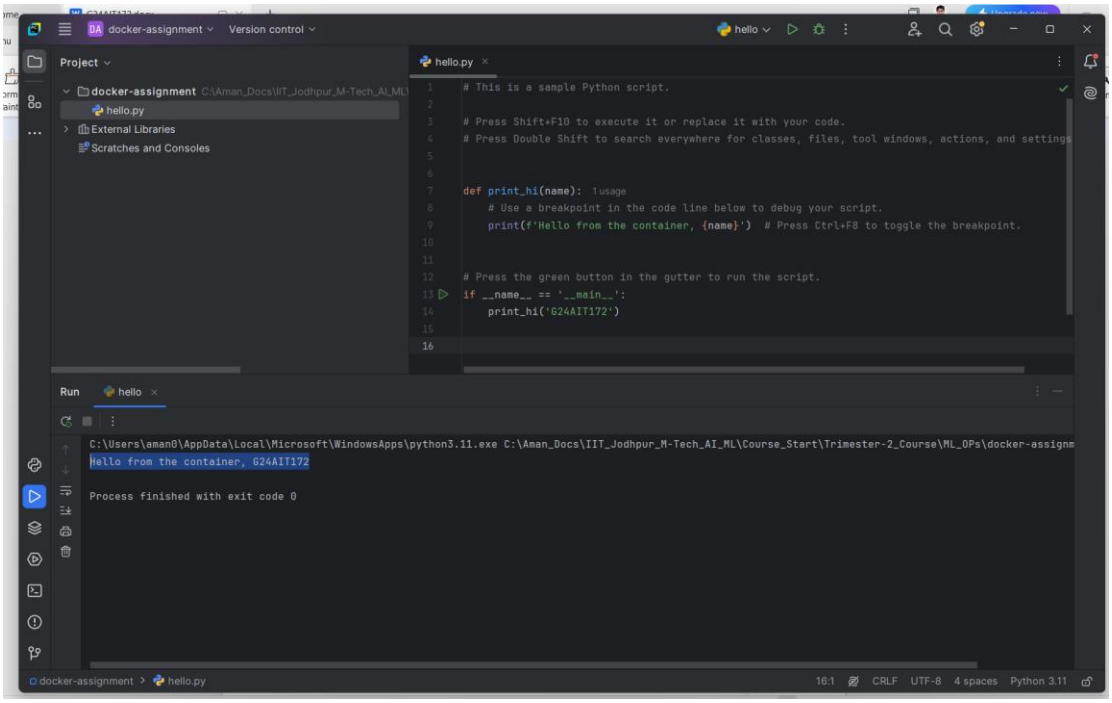
Have created a directory named - **"docker-assignment"**

```
C:\Aman_Docs\IIT_Jodhpur_M-Tech_AI_ML\Course_Start\Trimester-2_Course\ML_OPs>mkdir docker-assignment

C:\Aman_Docs\IIT_Jodhpur_M-Tech_AI_ML\Course_Start\Trimester-2_Course\ML_OPs>cdc docker-assignment
'cdc' is not recognized as an internal or external command,
operable program or batch file.

C:\Aman_Docs\IIT_Jodhpur_M-Tech_AI_ML\Course_Start\Trimester-2_Course\ML_OPs>cd docker-assignment

C:\Aman_Docs\IIT_Jodhpur_M-Tech_AI_ML\Course_Start\Trimester-2_Course\ML_OPs\docker-assignment>
```

Note : I am using Pycharm to create custom Docker Image

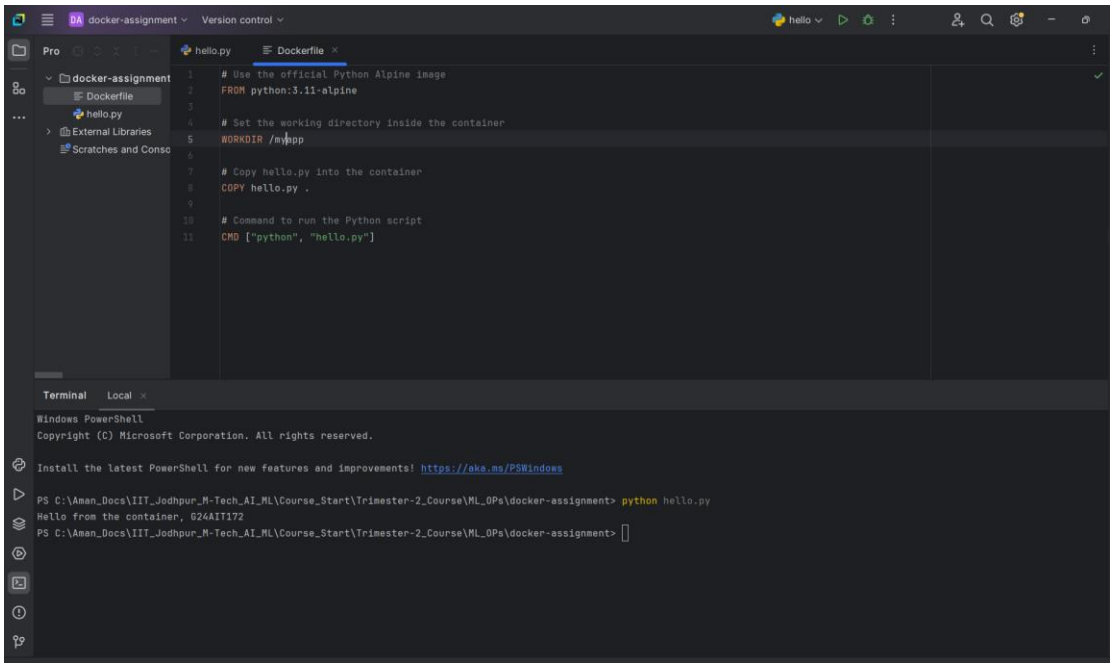
2. Create a hello.py (python) file-

So , first I am creating a python file - **hello.py** with the following content:
"Hello from the container,G24AIT172"



3. Now creating a Dockerfile

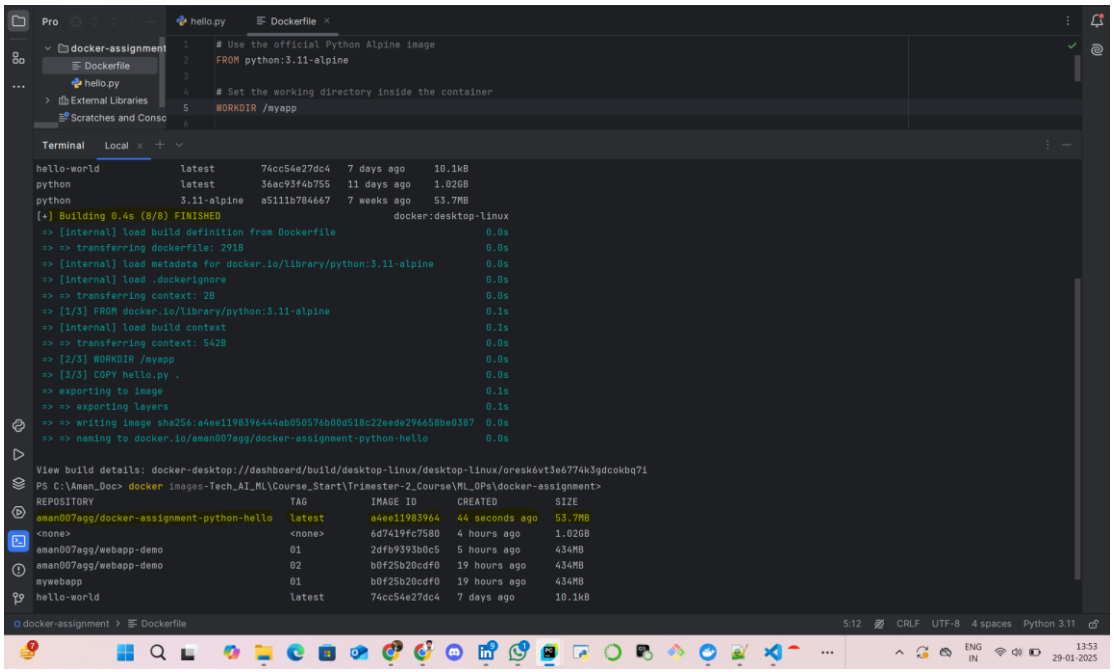
```
# Use the official Python Alpine image
FROM  python:3.11-alpine

# Set the working directory inside the container
WORKDIR  /myapp

# Copy hello.py into the container
COPY  hello.py .

# Command to run the Python script
CMD  ["python", "hello.py"]
```



- **FROM python:3.11-alpine:** Specifies the base image.
- **WORKDIR /myapp:** Sets the working directory inside the container.
- **COPY hello.py .:** Copies the script into the container.
- **CMD ["python", "hello.py"]:** Runs the script when the container starts.

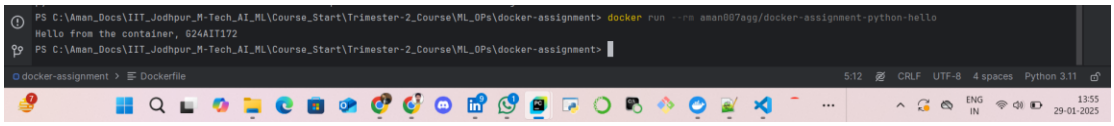
4. Building and Running the Docker Image

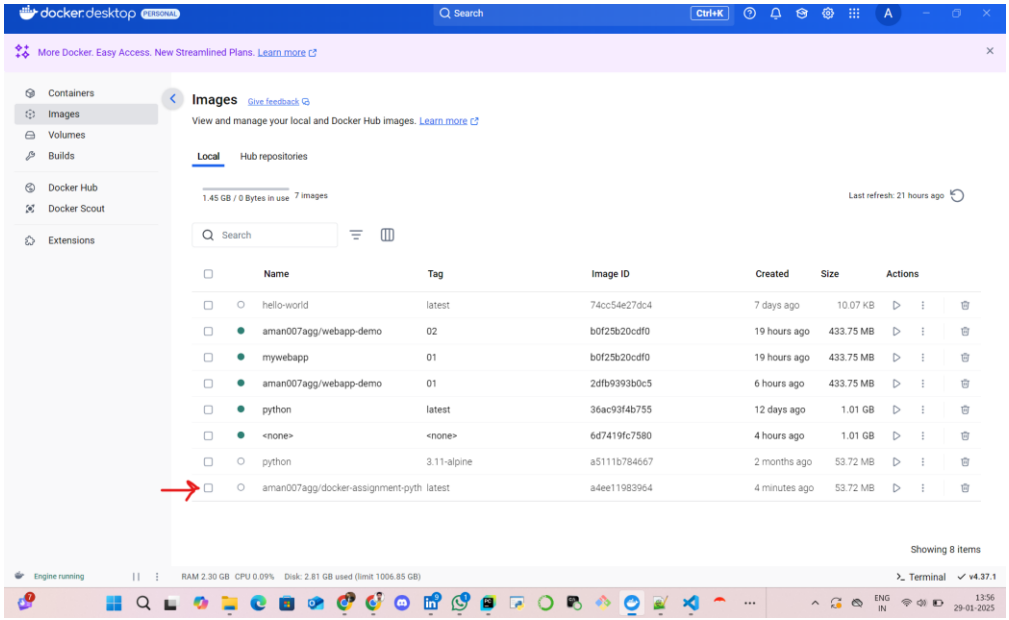
docker build -t aman007agg/docker-assignment-python-hello .



To verify that my image works correctly, I ran:

docker run --rm aman007agg/docker-assignment-python-hello



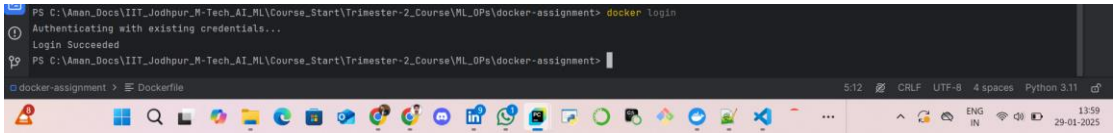


C. Push the created Docker image to DockerHub. The image should print the above statement when pulled and run. .

1. Pushing the Image to Docker Hub

First, I logged into Docker Hub using:

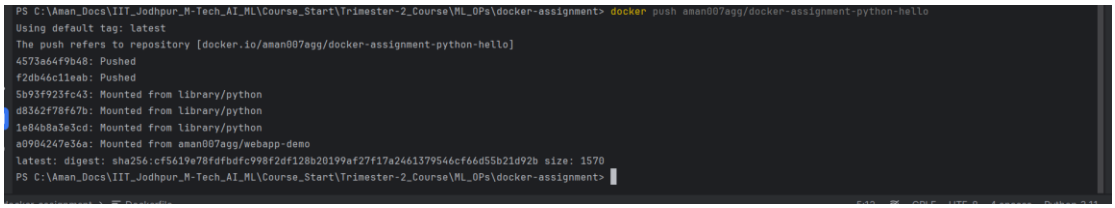
docker login

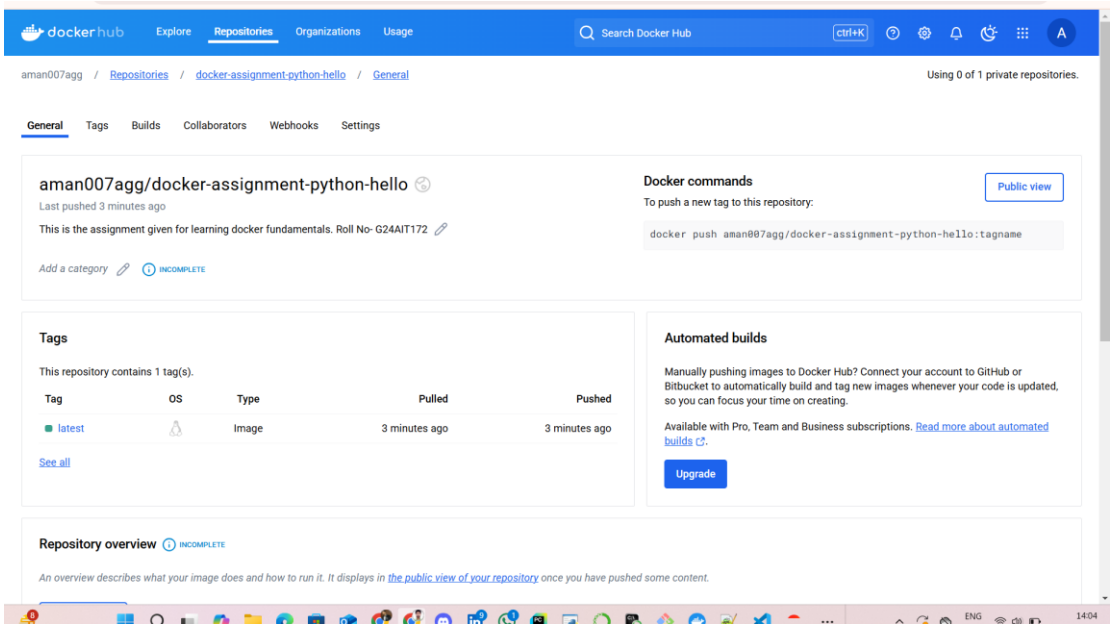


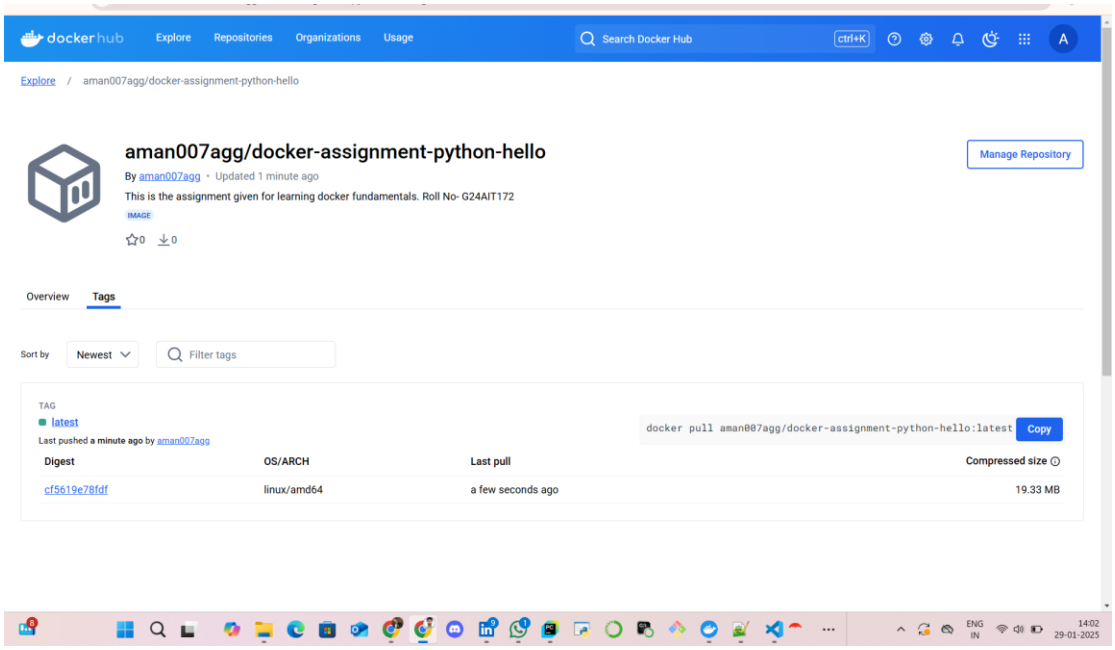
after it gets Authenticated successfully.

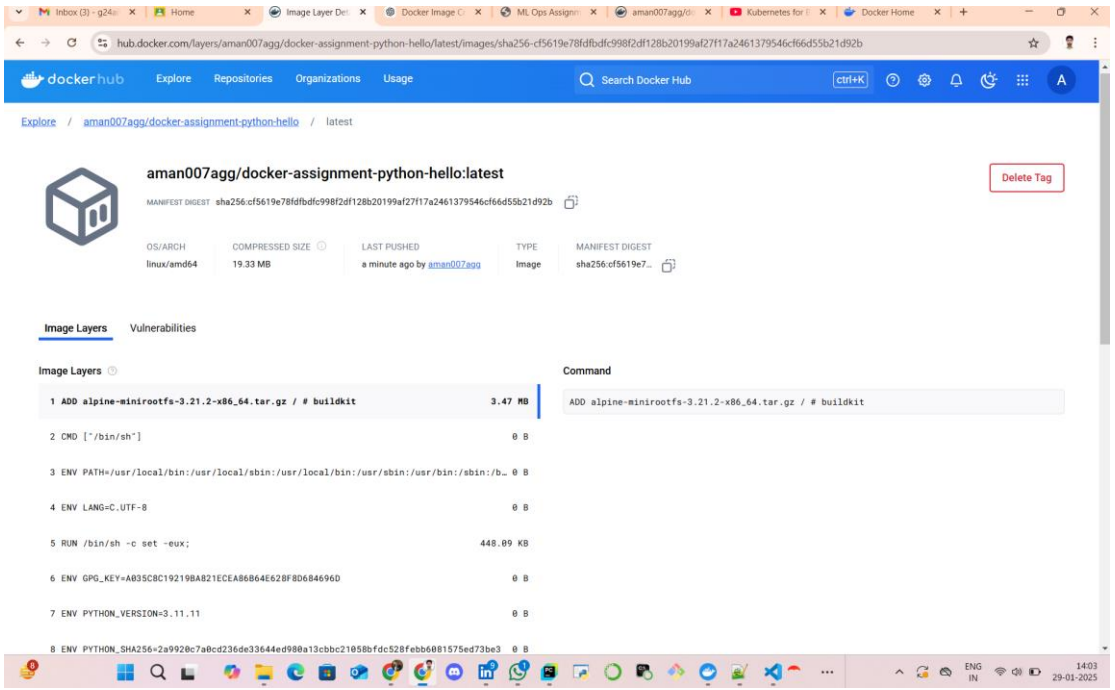
I pushed my image to my repository: **aman007agg/docker-assignment-python-hello**

docker push aman007agg/docker-assignment-python-hello



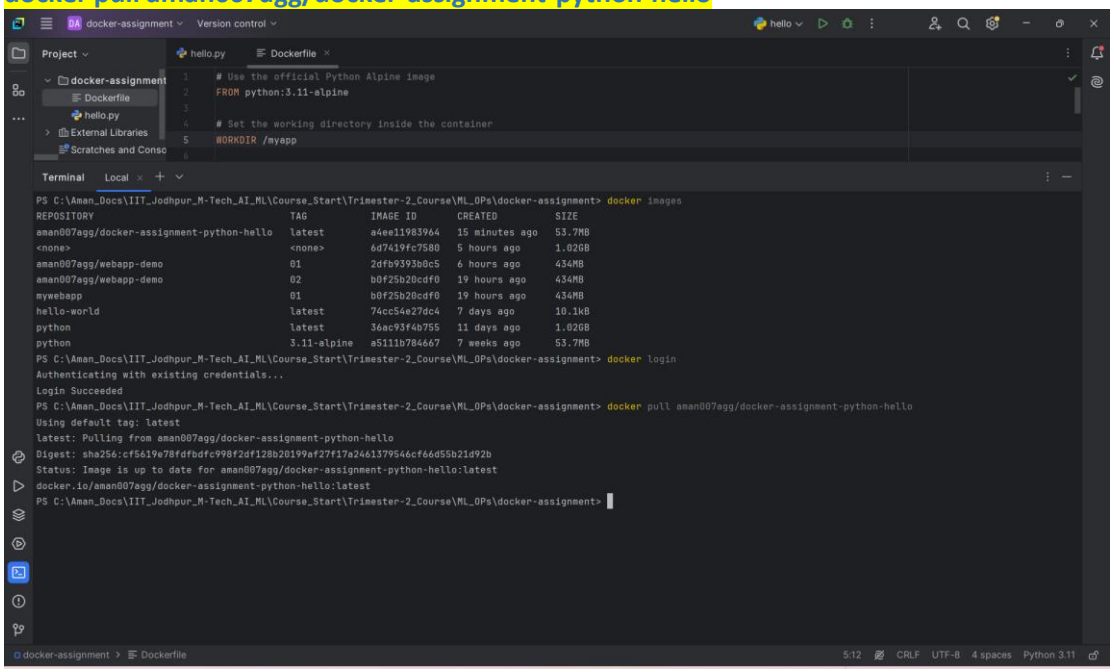



docker hub
Explore
Repositories
Organizations
Usage
Search Docker Hub
ctrl+K
Explore / aman007agg/docker-assignment-python-hello
aman007agg/docker-assignment-python-hello
By aman007agg · Updated 1 minute ago
This is the assignment given for learning docker fundamentals. Roll No- G24AIT172
IMAGE
0 0
Manage Repository
Overview
Tags
Sort by
Newest
Filter tags
TAG
latest
Last pushed a minute ago by aman007agg
docker pull aman007agg/docker-assignment-python-hello:latest
Copy
Digest
OS/ARCH
Last pull
Compressed size
cf5619e78fdf
linux/amd64
a few seconds ago
19.33 MB


hub.docker.com/layers/aman007agg/docker-assignment-python-hello/latest/images/sha256-cf5619e78fdfbdfc998f2df128b20199af27f17a2461379546cf66d55b21d92b
docker hub
Explore
Repositories
Organizations
Usage
Search Docker Hub
ctrl+K
Explore / aman007agg/docker-assignment-python-hello / latest
aman007agg/docker-assignment-python-hello:latest
MANIFEST DIGEST sha256:cf5619e78fdfbdfc998f2df128b20199af27f17a2461379546cf66d55b21d92b
Delete Tag
OS/ARCH
linux/amd64
COMPRESSED SIZE
19.33 MB
LAST PUSHED
a minute ago by aman007agg
TYPE
Image
MANIFEST DIGEST
sha256:cf5619e7...
Image Layers
Vulnerabilities
Image Layers
1 ADD alpine-minirootfs-3.21.2-x86_64.tar.gz / # buildkit 3.47 MB
2 CMD ["/bin/sh"] 0 B
3 ENV PATH=/usr/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/b… 0 B
4 ENV LANG=C.UTF-8 0 B
5 RUN /bin/sh -c set -eux; 448.09 KB
6 ENV GPG_KEY=A035C8C19219BA821ECEA86B64E628F8D684696D 0 B
7 ENV PYTHON_VERSION=3.11.11 0 B
8 ENV PYTHON_SHA256=2a9920c7a0cd236de33644ed980a13cbbc21058bfdc528febb6081575ed73be3 0 B
Command
ADD alpine-minirootfs-3.21.2-x86_64.tar.gz / # buildkit

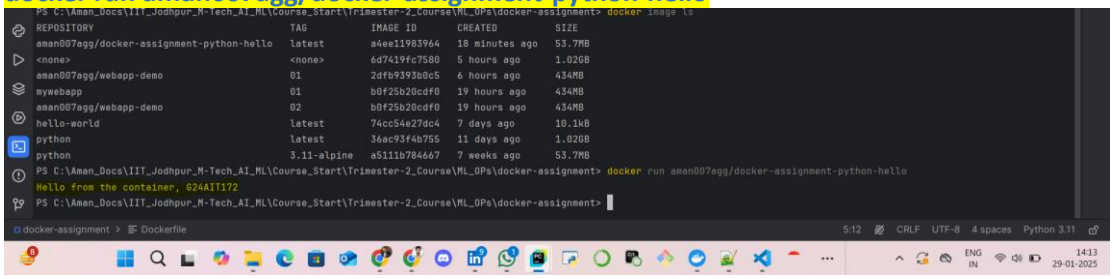
2. Steps to Pull the Image and Run the Container:

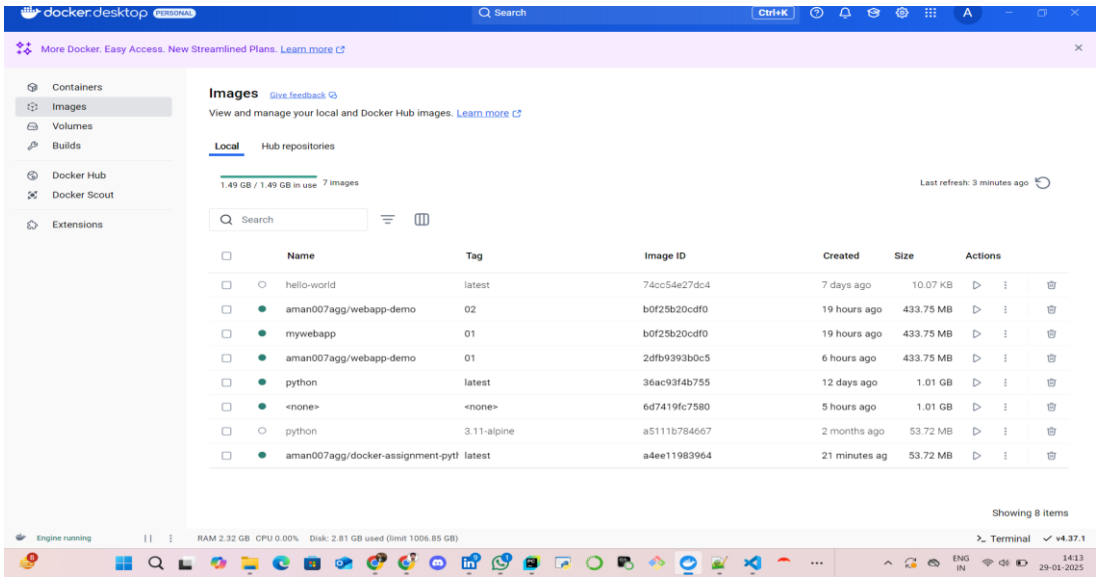
docker pull aman007agg/docker-assignment-python-hello

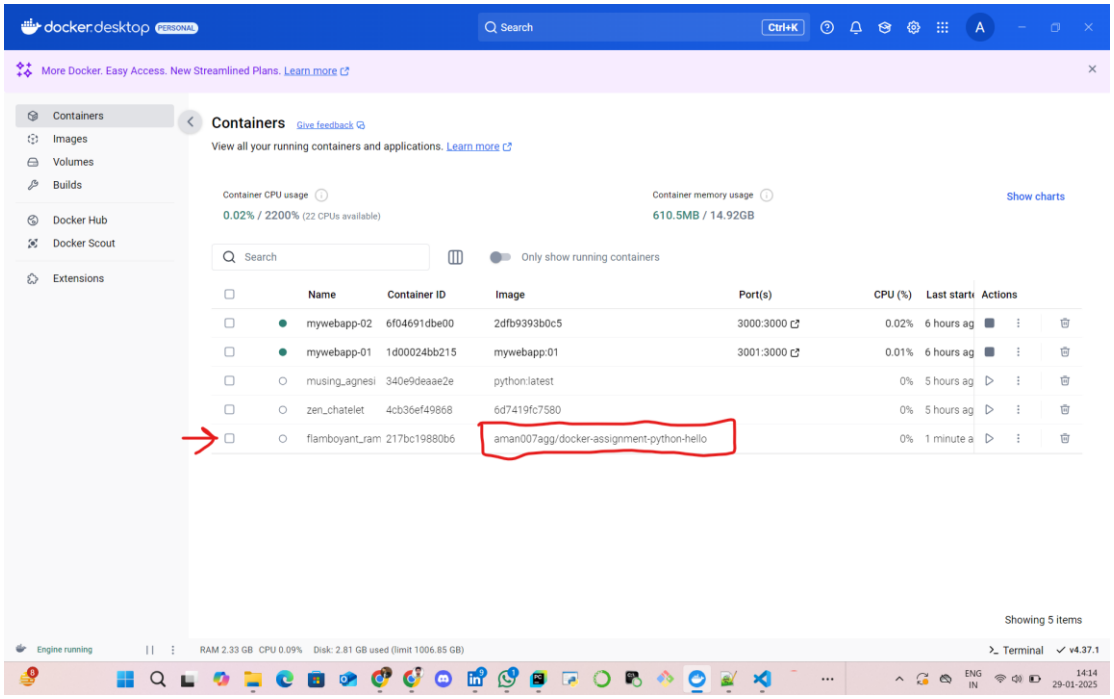


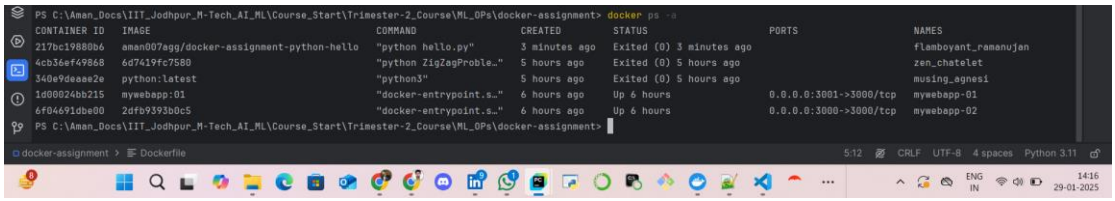
Run the Container:

docker run aman007agg/docker-assignment-python-hello









4. Conclusion

This assignment helped me understand the basics of Docker image creation and deployment. I successfully built a custom Docker image, tested it locally, and pushed it to Docker Hub.

Through this process, I became familiar with working with Dockerfile, docker build, docker run, and docker push commands.