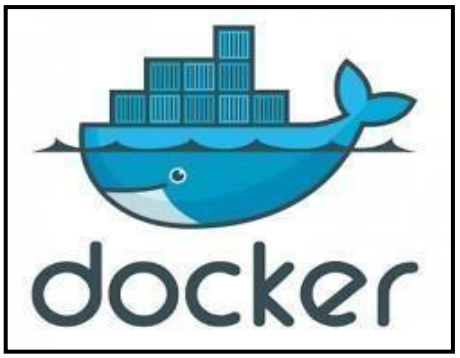
Assignments No. 7

Assignments Name	: Deploy a static website using Docker.		
Students Name	: Rohan N	**Roll-No:**	333047
Division	: TYIT –C2	**GR-No.**	: 22010064
Professor Name	: Prof. Vishal Meshram		

Theory:

1) **What is Docker :**
2)



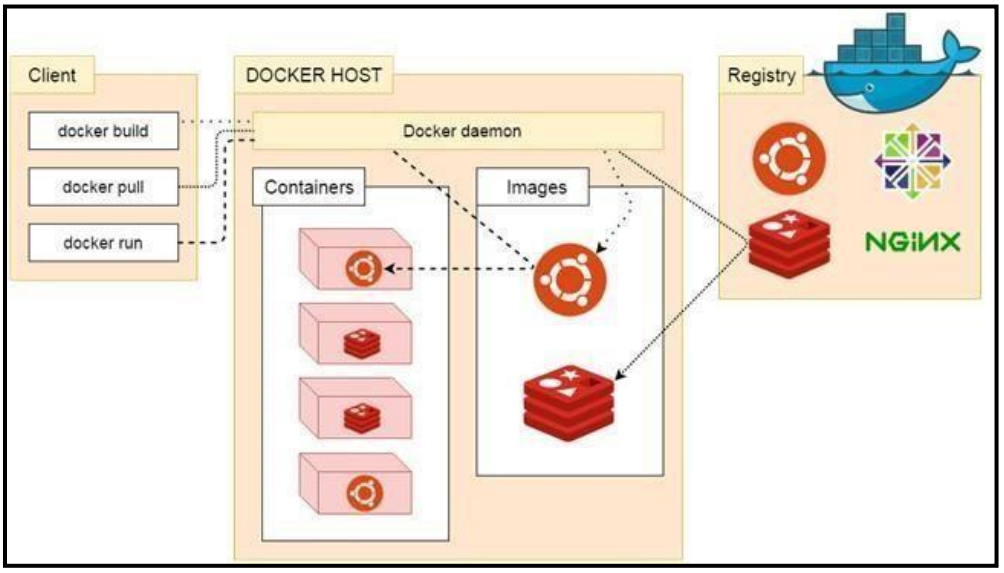
Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications. By taking advantage of Docker's methodologies for shipping, testing, and deploying code quickly, you can significantly reduce the delay between writing code and running it in production.

Docker provides the ability to package and run an application in a loosely isolated environment called a container. The isolation and security allows you to run many containers simultaneously on a given host. Containers are lightweight and contain everything needed to run the application, so you do not need to rely on what is currently installed on the host. You can easily share containers while you work, and be sure that everyone you share with gets the same container that works in the same way.

Docker provides tooling and a platform to manage the lifecycle of your containers:

- Develop your application and its supporting components using containers.
- The container becomes the unit for distributing and testing your application.
- When you're ready, deploy your application into your production environment, as a container or an orchestrated service. This works the same whether your production environment is a local data center, a cloud provider, or a hybrid of the two.

2) Docker Architecture :



Docker Daemon :
Docker daemon runs on the host operating system. It is responsible for running containers to manage docker services. Docker daemon communicates with other daemons. It offers various Docker objects such as images, containers, networking, and storage.

Docker client
Docker client uses **commands** and **REST APIs** to communicate with the Docker Daemon (Server). When a client runs any docker command on the docker client terminal, the client terminal sends these docker commands to the Docker daemon. Docker daemon receives these commands from the docker client in the form of command and REST API's request.

Docker Host :
Docker Host is used to provide an environment to execute and run applications. It contains the docker daemon, images, containers, networks, and storage.

Docker Registry :
Docker Registry manages and stores the Docker images.
There are two types of registries in the Docker -
Pubic Registry - Public Registry is also called as Docker hub.
Private Registry - It is used to share images within the enterprise.

Docker Objects :
There are the following Docker Objects -
Docker Images :

Docker images are the read-only binary templates used to create Docker Containers. It uses a private container registry to share container images within the enterprise and also uses public container registry to share container images within the whole world. Metadata is also used by docket images to describe the container's abilities.

Docker Containers :

Containers are the structural units of Docker, which is used to hold the entire package that is needed to run the application. The advantage of containers is that it requires very less resources.

In other words, we can say that the image is a template, and the container is a copy of that template.

3) Difference between Docker and Virtual machine

Features	VM (Virtual Machines)	Docker
Boot-Time	VM boots in a few minutes.	Docker takes a few seconds to boot.
Runs on	A virtual machine uses a hypervisor.	Dockers use an execution engine.
Memory Efficiency	It is less efficient because it requires the whole operating system to be loaded before beginning the surface.	No space will be required for virtualization, so less memory.
Isolation	Interference possibility will be minimum because of its isolation mechanism.	Dockers are prone to adversities. No provisions for many isolation systems.
Deployment	VM contains lengthy deployment because it isolated instances are liable for execution.	Docker contains easy deployment because of an individual image. It is containerized and can be applied beyond each platform.
Usage	A virtual machine has tools that are simpler and easy-to-use to implement.	Docker has a convoluted usage mechanism. It consists of docker managed tools and third party both.
OS support	All virtual machines have an isolated OS.	All the containers can distribute OS.
Storage	It requires a few GBs.	Its container is lightweight (MBs/KBs).
Availability	Ready-made virtual machines are available but complex to find.	Pre-built containers of dockers are available.
Resource Usage	More usage of resources.	Less usage of resources.
Creation Time	Creating a virtual machine will take a longer time relatively.	The container of the docker can be made in seconds.

4) Docker Commands

docker --version :
docker pull :
docker run :
docker ps :
docker exec :
docker stop :
docker restart
: docker kill :
docker push :

5) Dockerfile :

Dockerfile uses DSL (Domain Specific Language) and contains instructions for generating a Docker image. Dockerfile will define the processes to quickly

produce an image. While creating your application, you should create a Dockerfile in order since the Docker daemon runs all of the instructions from top to bottom.

Steps To Create a Dockerfile •

Create a file named Dockerfile.

- Add instructions in Dockerfile.
- Build Dockerfile to create an image.
- Run the image to create a container.

Docker file instructions :

- FROM <IMAGE_NAME>
- COPY <SOURCE> <DESTINATION>
- ADD <URL>
- RUN <COMMANDS + ARGS>
- CMD <COMMANDS + args>
- ENTRYPOINT <COMMANDS + args>
- MAINTAINER <NAME>

Implementation:

Step 1: Install nginx on windows follow the link:

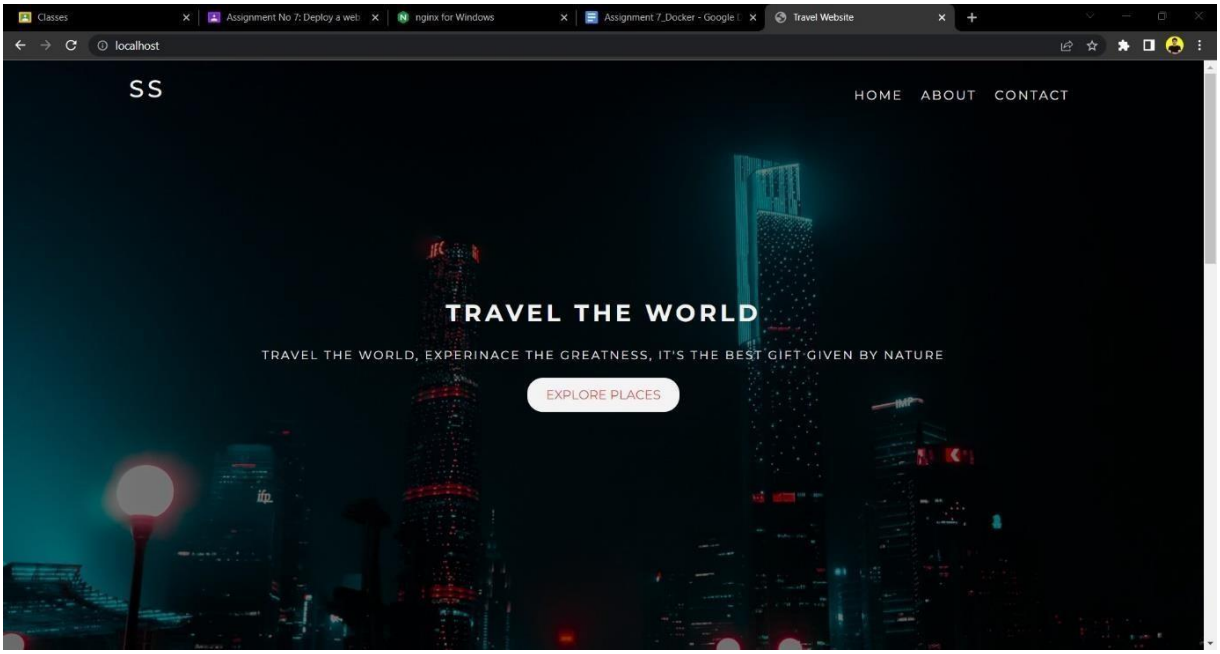
http://nginx.org/en/docs/windows.html

cd c:\

unzip nginx-1.23.4.zip cd nginx-1.23.4 start nginx

Step 2: Copy the sample-website in "C:\nginx\html\" folder

Step 3: open browser and run "localhost:80"



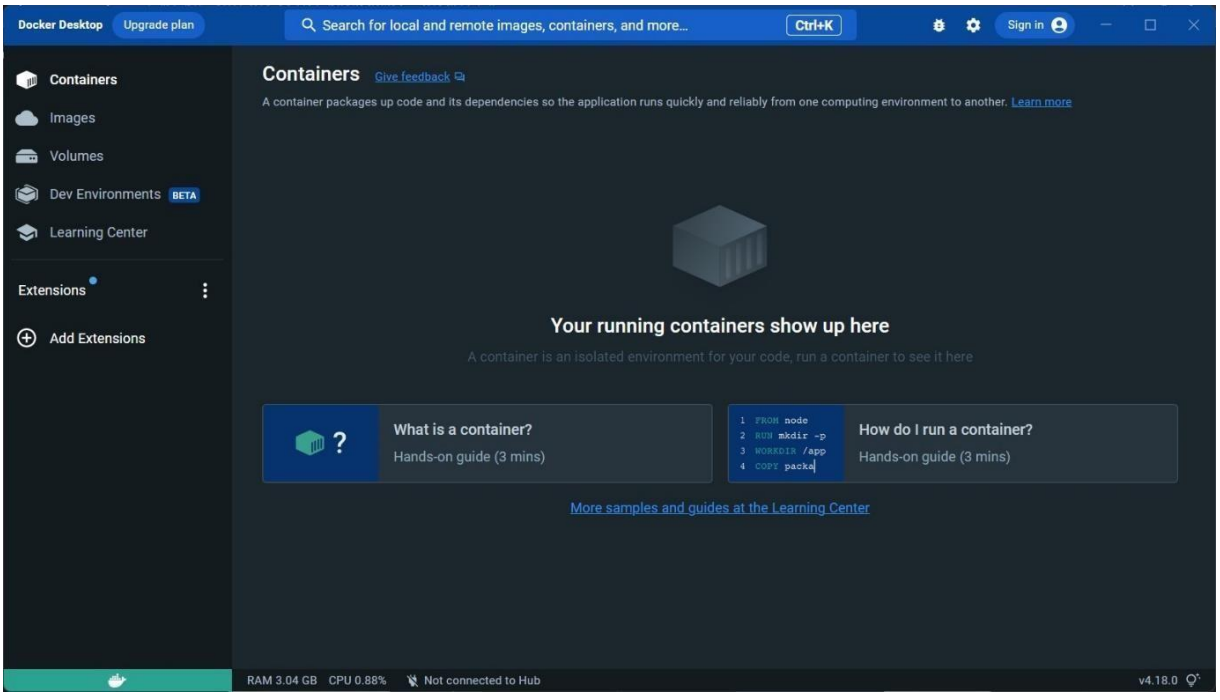
Step 4: Download Docker for windows, follow the link

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

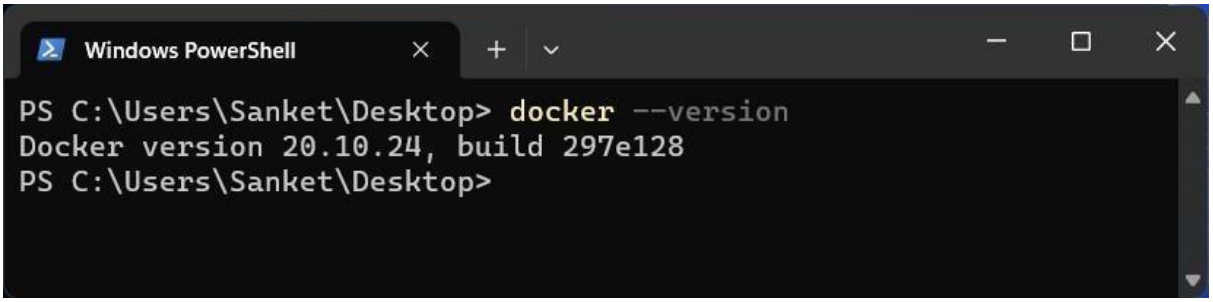
Step 5: Start Docker Desktop

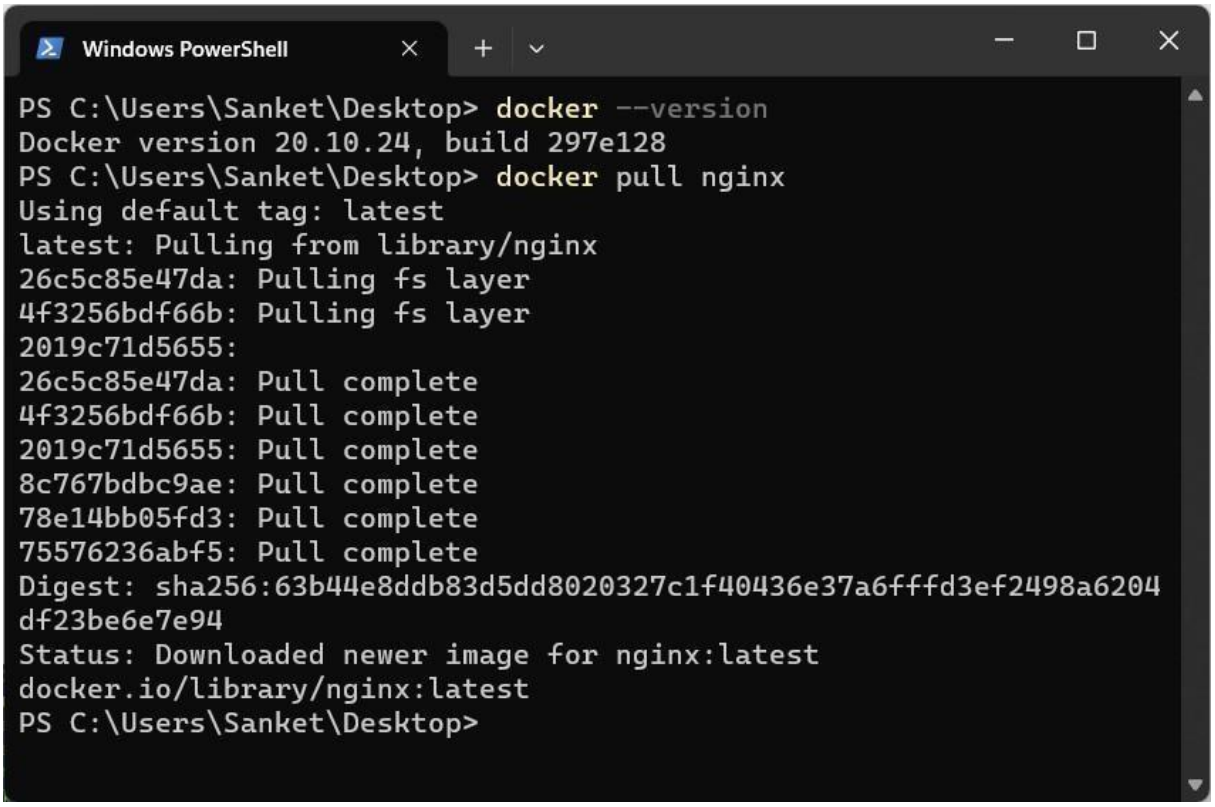
Docker Desktop does not start automatically after installation. To start Docker Desktop:

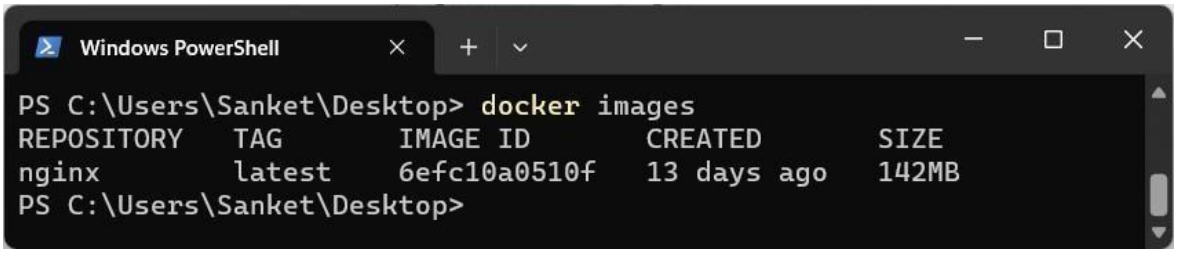
1. Search for Docker, and select **Docker Desktop** in the search results.



Step 6: Open Powershell and check Docker installation using commands:


Windows PowerShell
PS C:\Users\Sanket\Desktop> docker --version
Docker version 20.10.24, build 297e128
PS C:\Users\Sanket\Desktop>

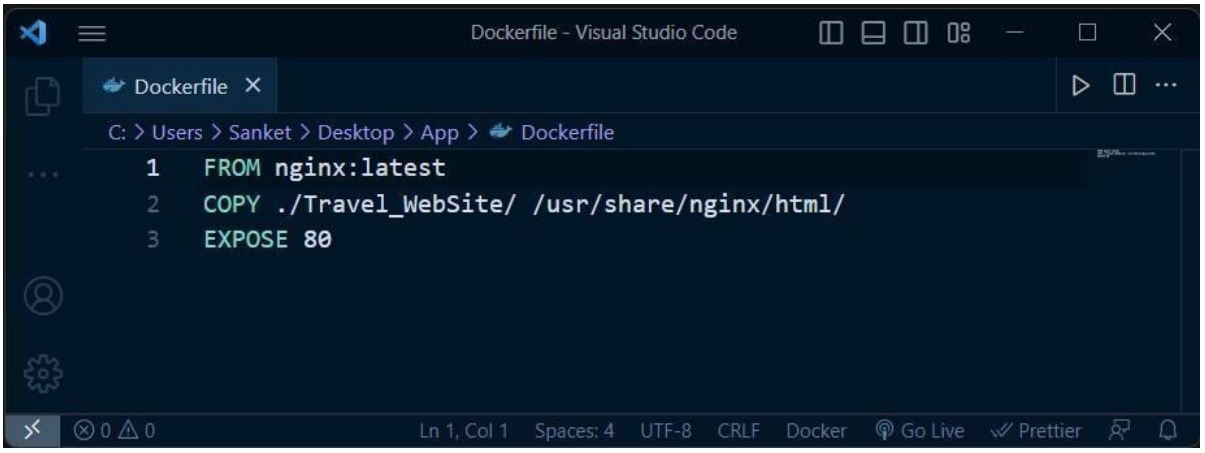

Windows PowerShell
PS C:\Users\Sanket\Desktop> docker --version
Docker version 20.10.24, build 297e128
PS C:\Users\Sanket\Desktop> docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
26c5c85e47da: Pulling fs layer
4f3256bdf66b: Pulling fs layer
2019c71d5655:
26c5c85e47da: Pull complete
4f3256bdf66b: Pull complete
2019c71d5655: Pull complete
8c767bdbc9ae: Pull complete
78e14bb05fd3: Pull complete
75576236abf5: Pull complete
Digest: sha256:63b44e8ddb83d5dd8020327c1f40436e37a6fffd3ef2498a6204
df23be6e7e94
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
PS C:\Users\Sanket\Desktop>


Windows PowerShell
PS C:\Users\Sanket\Desktop> docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
nginx latest 6efc10a0510f 13 days ago 142MB
PS C:\Users\Sanket\Desktop>

Docker file:

Steps to run the "Sample website" in Docker container

Step 1) visit to Docker hub web site: https://hub.docker.com/



Step 3) pull the latest image of nginx using command "docker pull nginx"

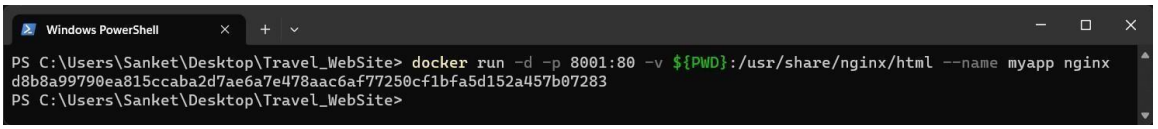
Step 4) check the docker images on your desktop by using command:

"docker images"

```
C:\Users\Sanket\Desktop\App>docker images
REPOSITORY   TAG       IMAGE ID       CREATED          SIZE
my-app       v1        e924c6543ac7   2 minutes ago    157MB
nginx        latest    6efc10a0510f   2 weeks ago      142MB
```

Step 5) go in the "SampleWebsite" folder and then Create a container using the docker command and sync the "SampleWebsite" folder with folder inside the container folder. (This is called Mount Bind")

"docker run -d -p 8001:80 -v ${PWD}:/usr/share/nginx/html --name web-site nginx"



Step 6)verify the website open browser and chec "localhost:8001". Now this website is running inside your container.

DockerFile

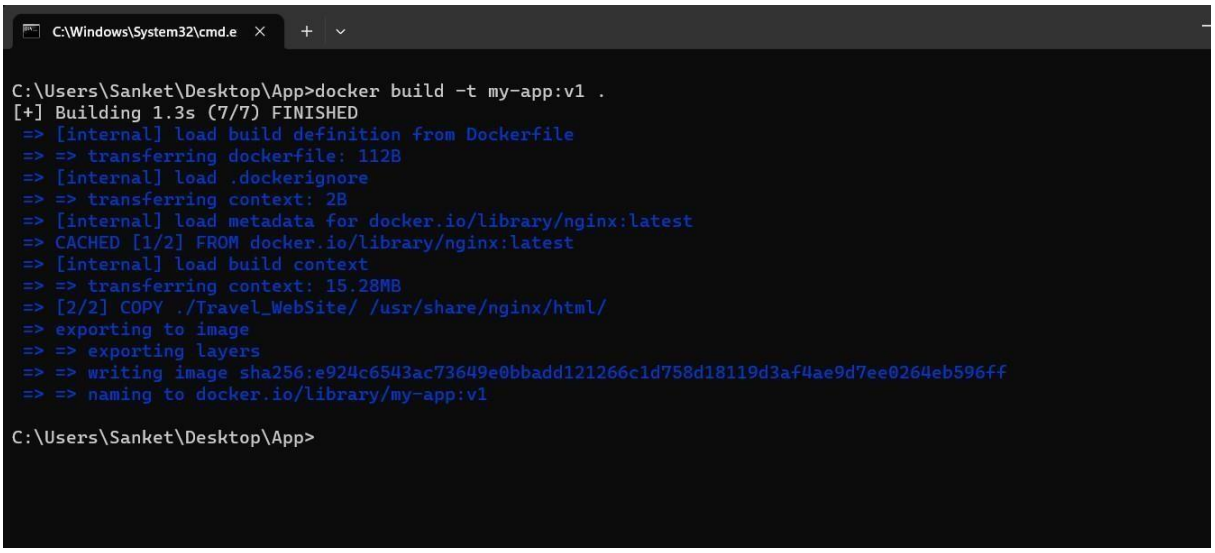
Step 1) Create a Directory structure like

Step 2) Write a following script into "Dockerfile"

Step 3) build image from docker file using command

"docker build -t my-app:v1 . "

Step 4) check images using command: docker images

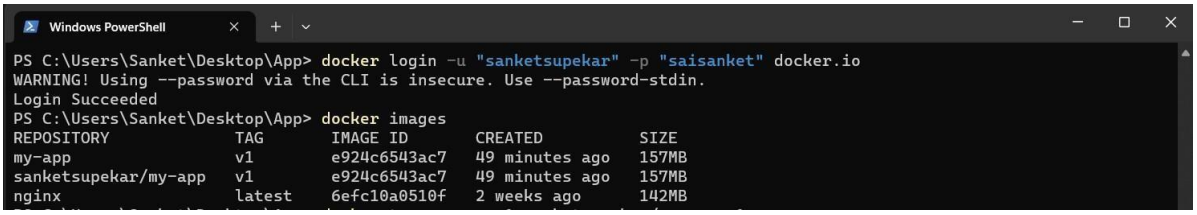


PUSH Image to "DockerHub"

Step 1) login to docker hub using command

1) docker login -u username

2) docker images



3) docker tag (old image name) username/newname

4) docker push