-What is Docker?
Docker is a platform that helps to run and develop applications by packing the code into an immutable unit called IMAGE.
BUILD ONCE, RUN EVERYWHERE

-Can we consider Docker as a Virtual machine?
Docker is mainly based on Containerization which makes it more flexible and easy to manage. It uses OS-level Virtualization which makes it lightweight and disposable than a virtual machine working on the hypervisor. It is much faster to initialize the applications.

-What are the advantages of using Docker?
Efficiency: In Docker, there is no need of assigning memory as it is immutable reflecting it is what it is.
Cost-effective: Running An OS over another OS is a wasteful criterion that is eliminated by docker and that's what makes it cost-effective.
Portability: The standardized format makes docker and containers portable.

-Image
If one is familiar with Object Oriented Programming, we know that there exist Classes(Blueprint) and Objects(states or behavior) and we can relate Images with Class and Containers with Objects such that every Image describes a Container.
Under this, we also have access to look up the Image built by others.

-How does it work?
docker file:
Firstly, we take a base image and introduce an entry point that defines the command that will run when started.
Secondly, we declare an environment variable which we can override whenever we want and tell the container the command run time, what directory the command is.
Thirdly, we copy files into images and applications.
Lastly, we give a command for execution.
docker build: This command is run by default and takes Docker File out to create an Image.
docker run: Starts Running the Images
docker ps: Lists out the container that is running.
Docker rm: Deletes the container.
Docker Image rm: Deletes the image.

-Registry: A place where the copy of images is stored.

-Docker Hub: It is a public registry. The main registry is hub.docker.com

-Working with the registry?
Docker Pull: Grabs the Image from a repository.

Docker Tag: It puts an identifier or tag over an image like a version number that segregates it from others.
Docker Login: To interact with Docker we need some authentication like with any other system, basically permission for performing operations like Docker Push.

Docker - Compose:
It describes the layout of an application. It's like Infrastructure as a Code.
It is written down just once and avoids repetition.
Therefore, instead of copy-pasting the commands from any Project Docs, we can collect them from the docker-compose .yaml file.
Once we have a .yaml file, we can then run the entire application using a single command.

It was a powerful session and I enjoyed learning about Docker. Thank you:)