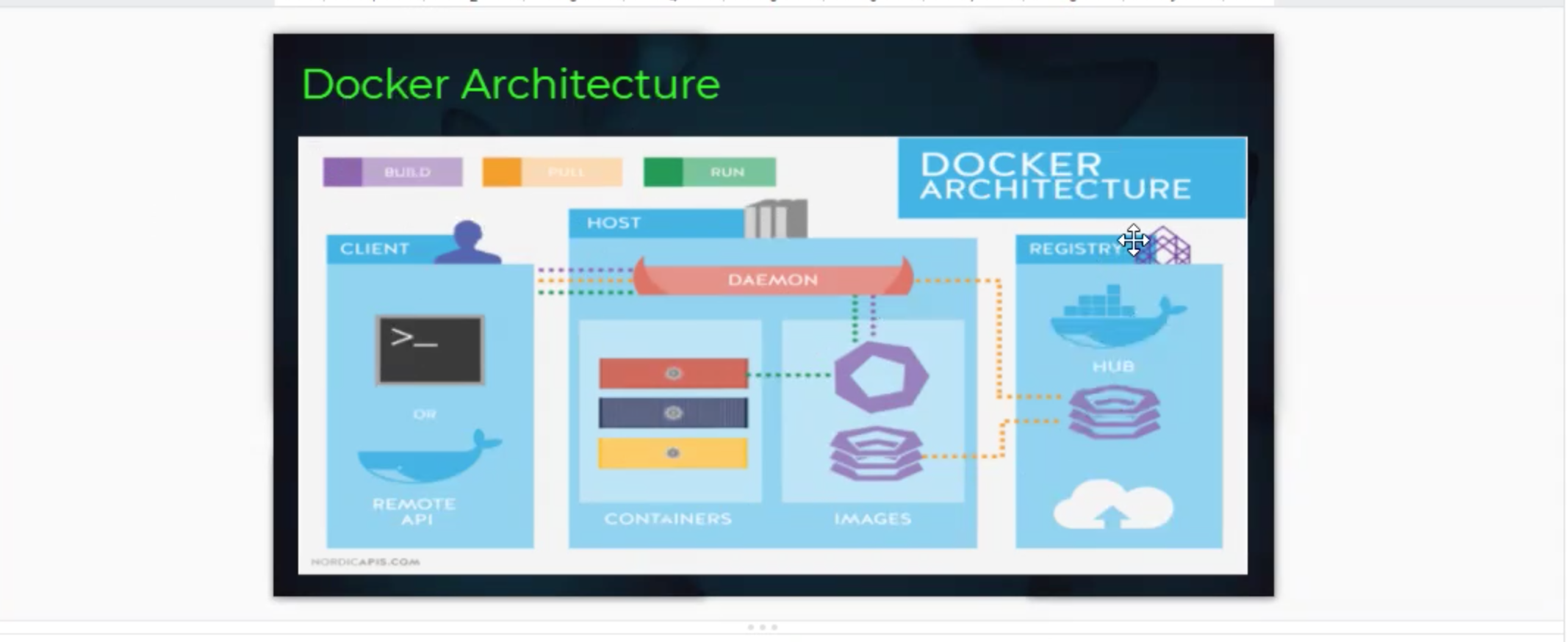
**Docker Architecture**:



1.Docker client:

Performs docker build, pull and run operations to building communication with the Docker host.

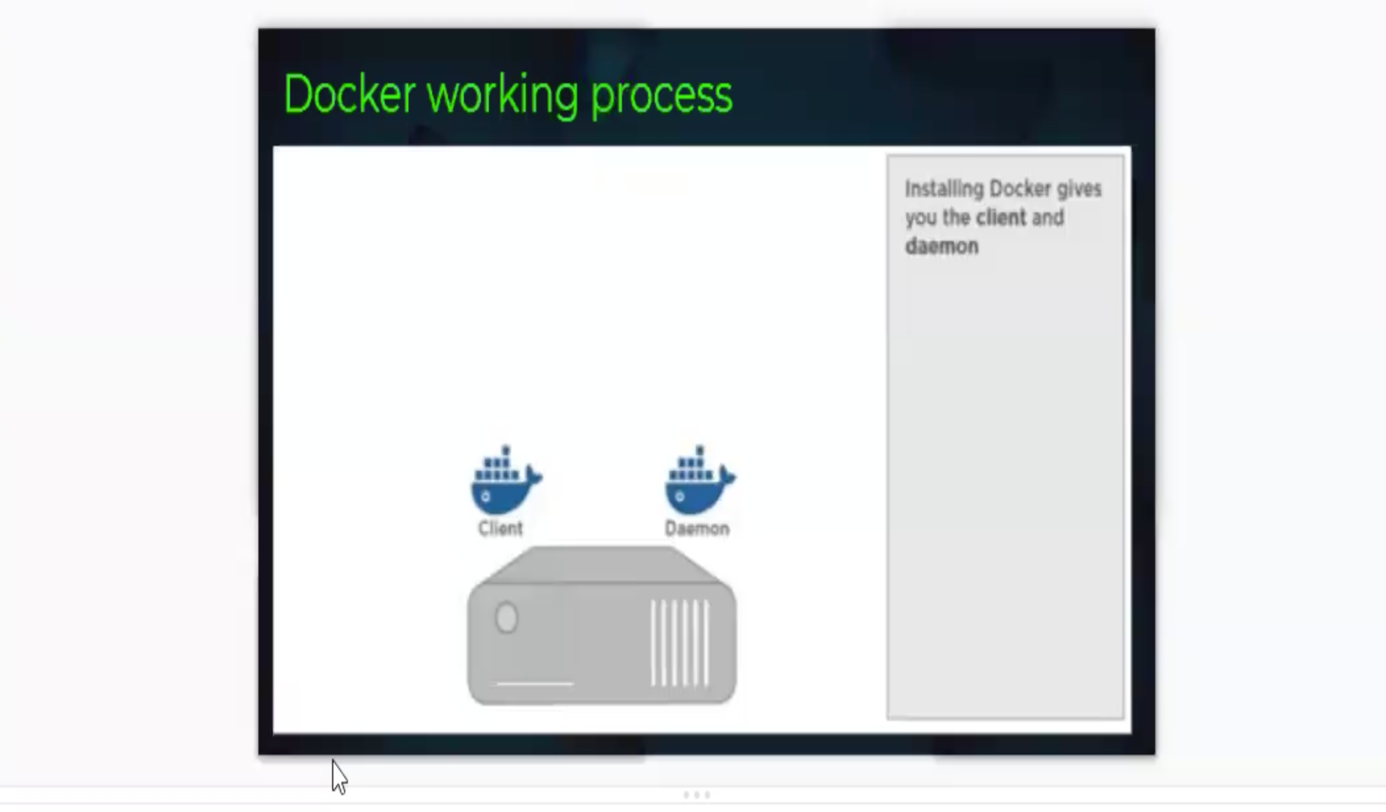
2.Docker host:

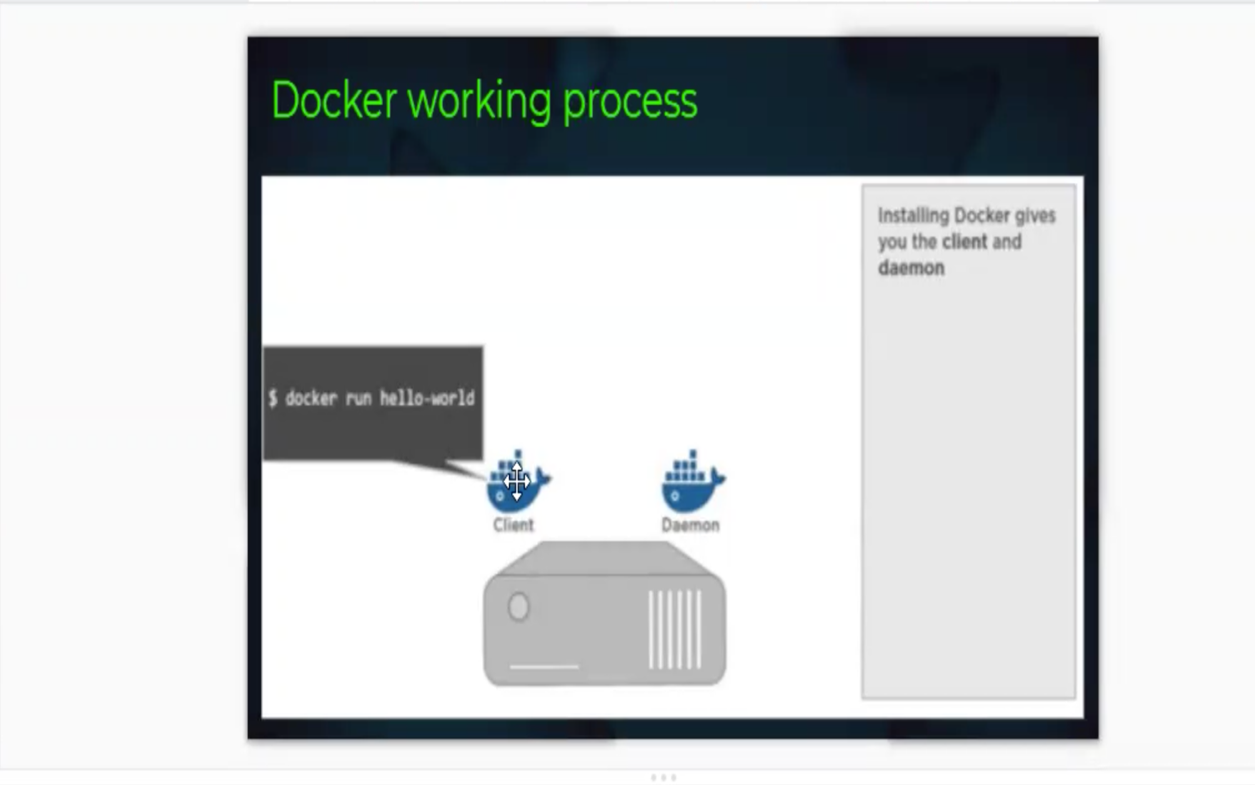
* Components contains Docker Daemon, Container and images.
* Images will be the kind of metadata for the applications which are the containerized in the container.
* The docker Daemon build a connection with Registry.

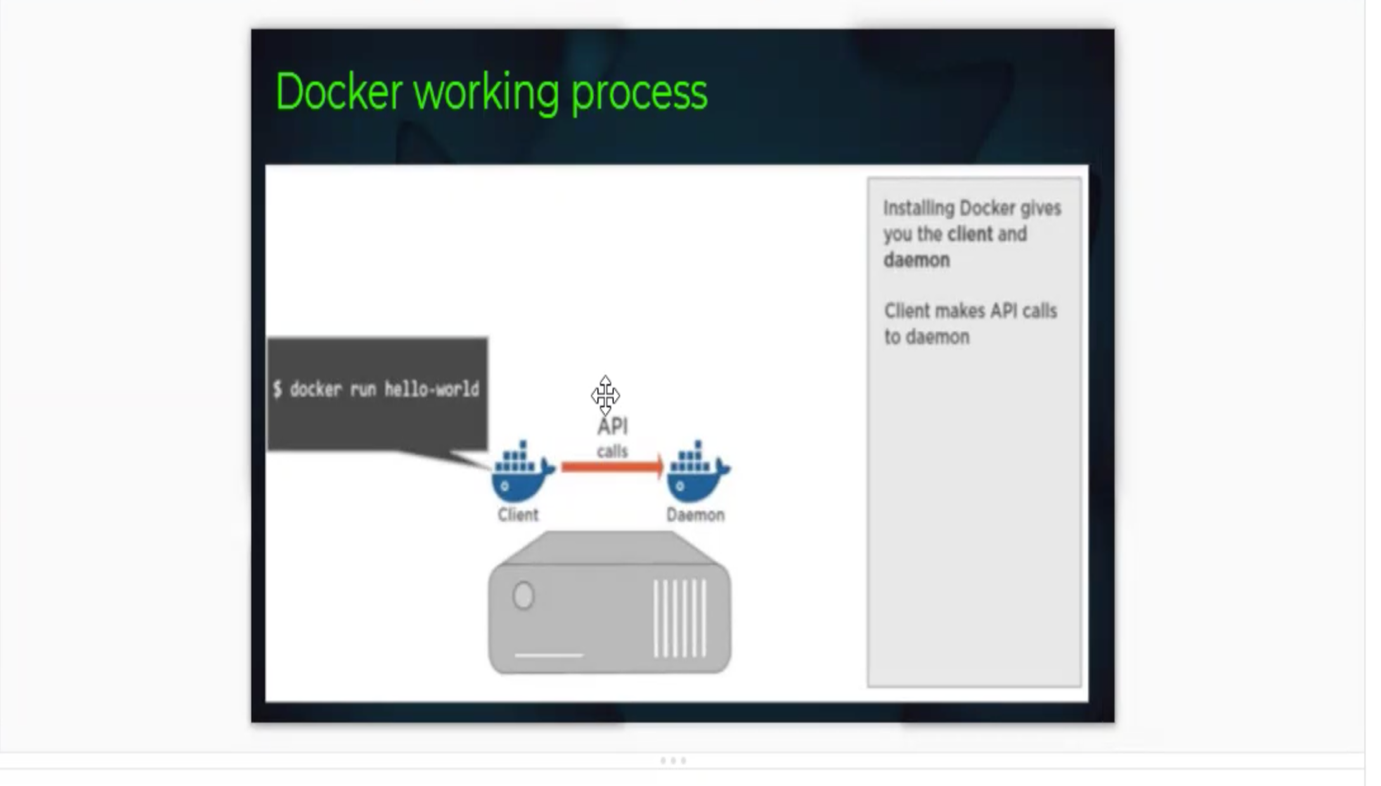
3.Regisrty**:**

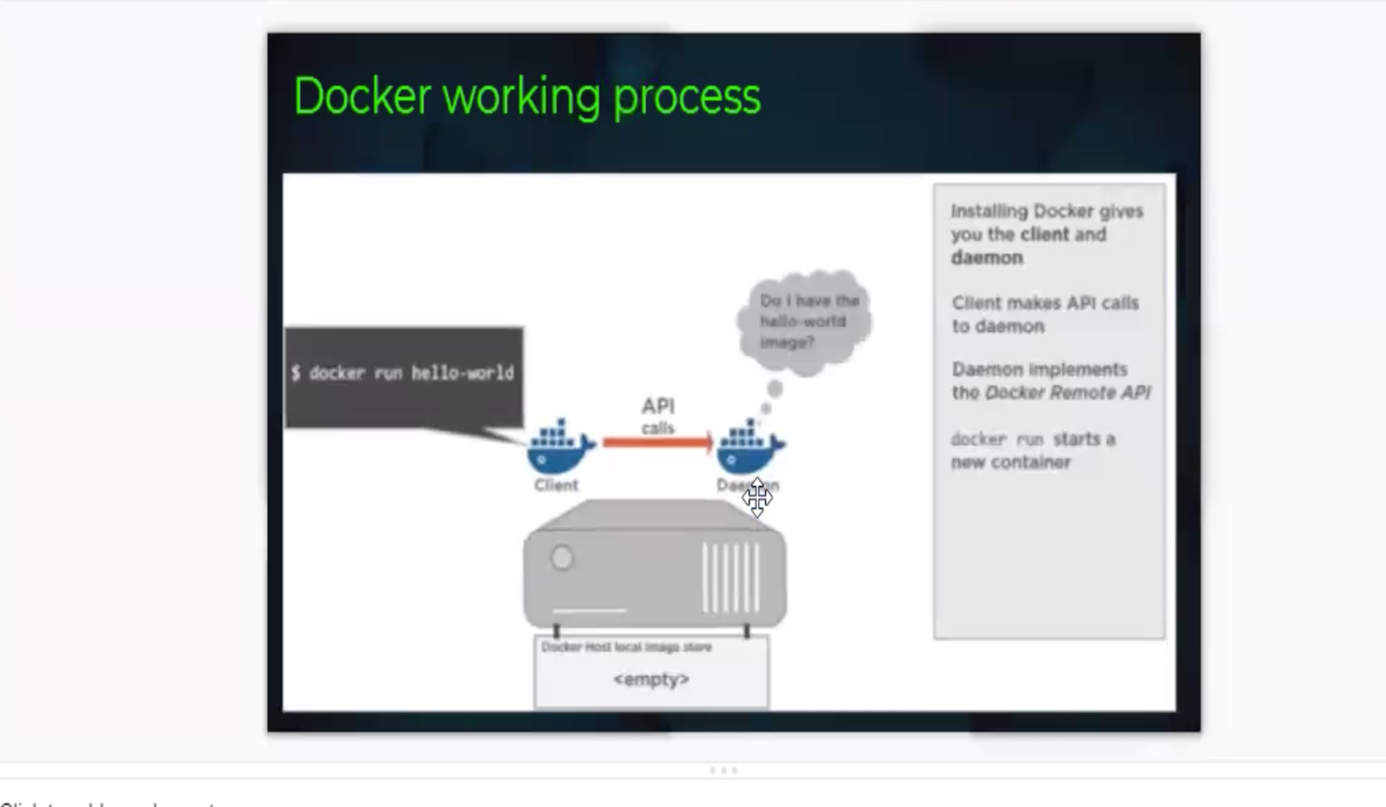
Component will be storing the Docker images.

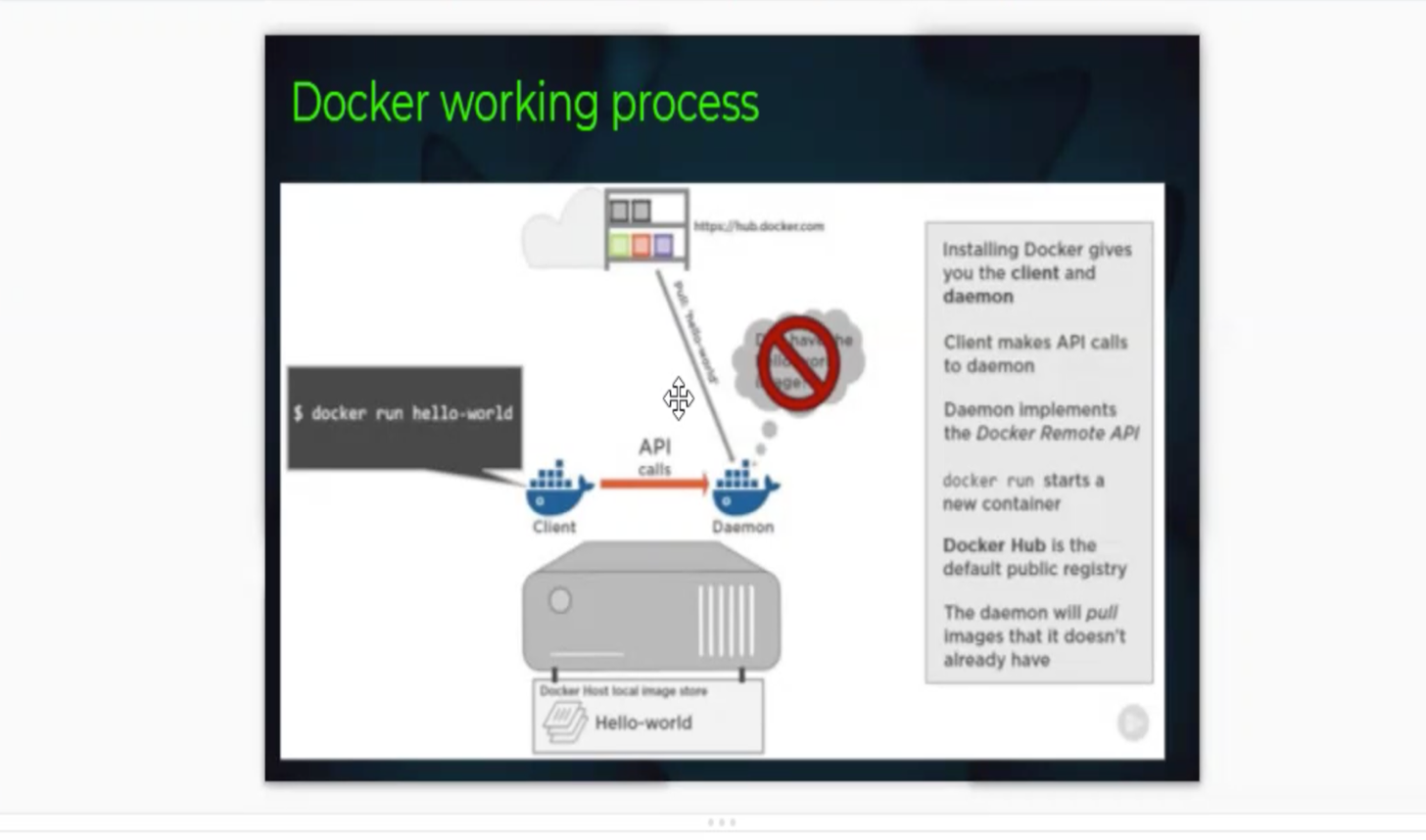
Docker working process:

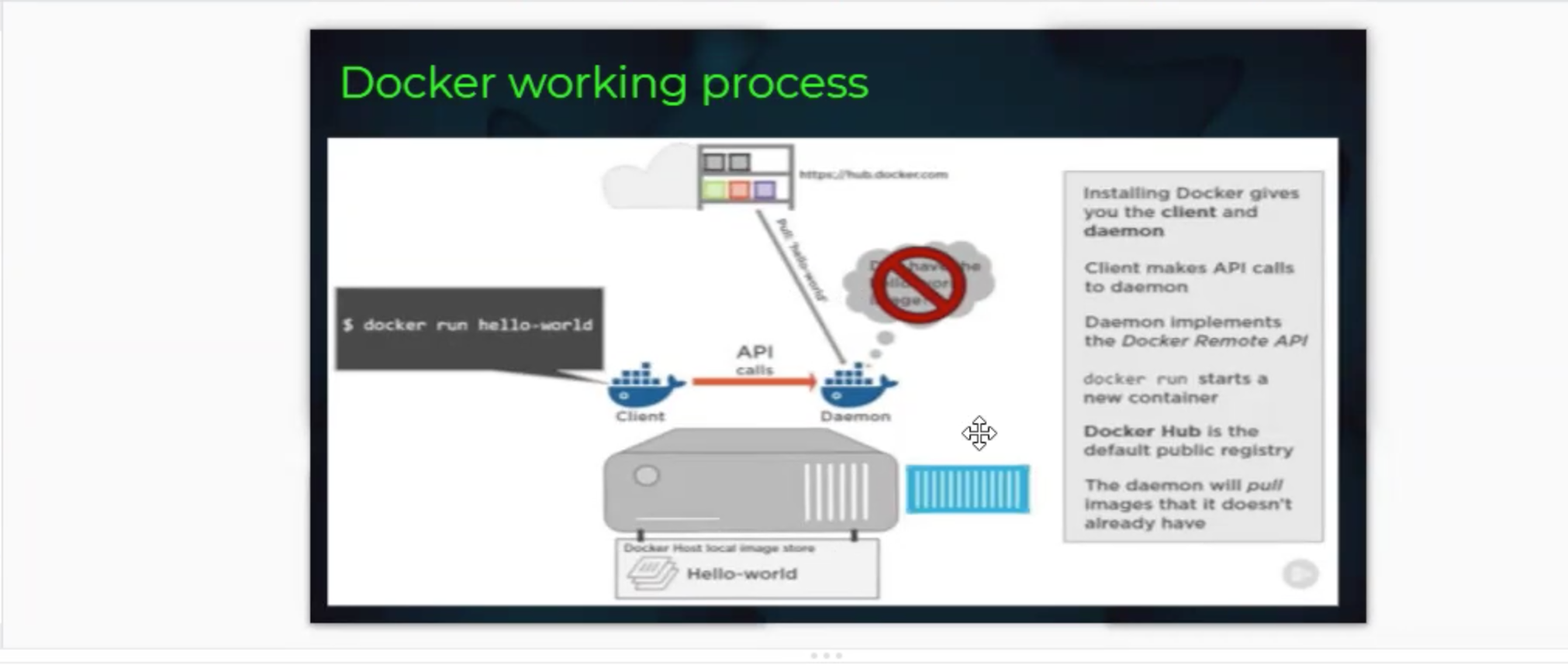


****

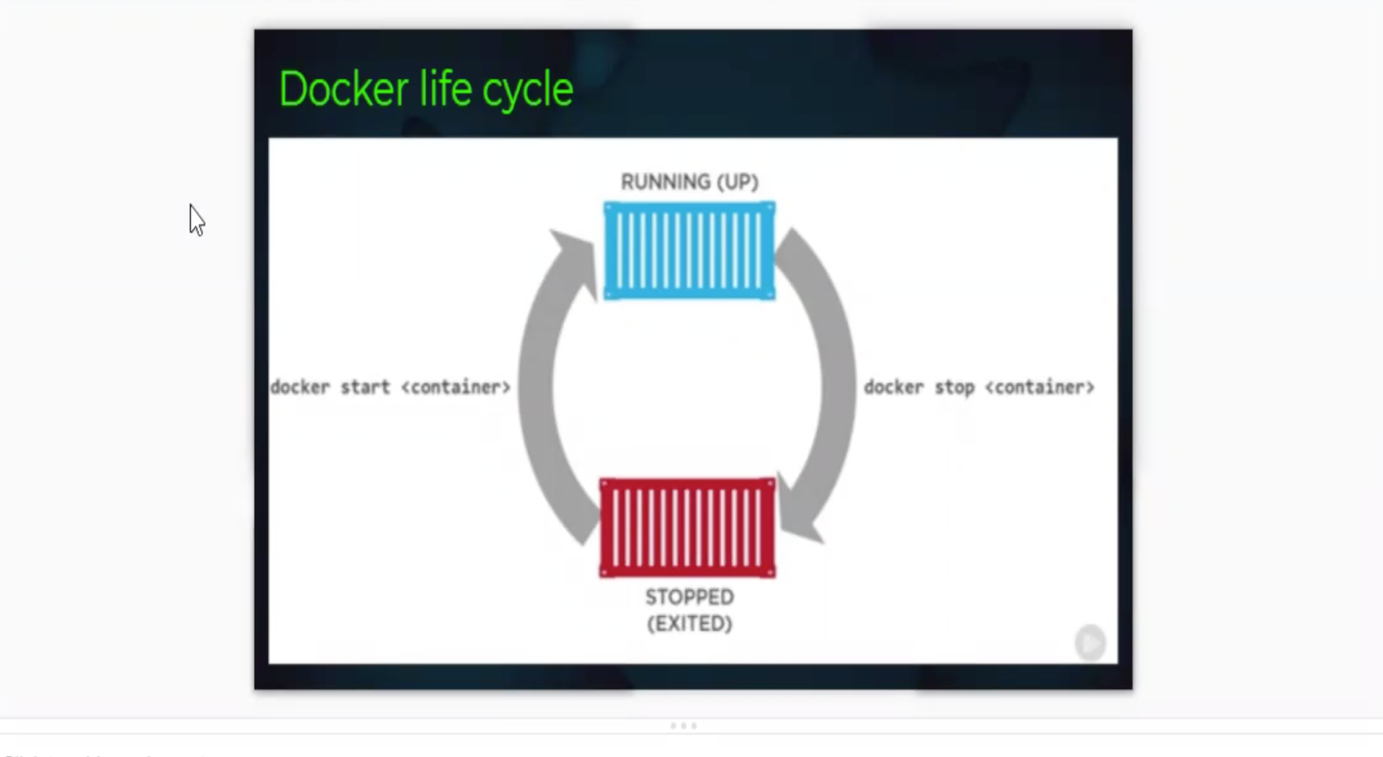








Docker life cycle:



**Create a Container Commands:**

1. Syntax:

Docker run --name <container name> -d -p host port number: guest port number <image name>

* d means detach mode
* p means port forwarding

**note:** docker run=search + pull + create

**Example 1:**

1. Open AWS console and Select the instance and click on connect and click on SSH client tab click on copy symbol under example.
2. Open gitbash terminal where our pem file is located.
3. Now paste here and it asks for yes/no/fingerprint. , type YES.
4. Type sudo -i
5. Type the command like below:

docker run --name myc1 -d -p 8081:80 nginx

6.Type docker ps

7.Type docker images

8.Open aws console and select docker server instance

9.Copy the public ip and paste in browser like below:

< public ip >:8081

**Note**: Now we will able to see nginx webserver

**Basic Container related commands**:

1. To list all the running containers

**docker ps(process status**)

1. To list all running containers and including stopped

**docker ps -a**

1. To show only container id

**docker ps -a -q**

1. To find all details about a docker container

**docker inspect <container id/name>**

1. To navigate inside the container

**docker exec -it <container id/ name>/bin/bash**

**it** means interactive mode

1. To see the logs of an container

**docker logs <container id /name>**

1. To stop the container

**docker stop <container id /name>**

1. To start the container

**docker start <container id/name>**

1. To show docker disk usage

**docker system df**

1. To display live stream of containers resource usage statics

**docker stats --no-stream**

1. To delete a single running container

**docker rm <container id/name> -f**

1. To delete all running containers at a time

**docker rm $(docker ps -a -q) -f**

**BASIC IMAGES RELATED COMMANDS:**

1. To list all images available in our local system

**docker images**

1. To search images on docker hub

**docker search <image name>**

1. To download any image from docker hub

**docker pull <image name>**

1. To show only image id

**docker images -q**

1. To see the history of an image

**docker history <image id /name>**

1. To remove single docker image in our local system

**docker rmi <image id/name>**

1. To remove all docker image in our local system at a time

docker rmi -f $(docker images)

Example 2: (*Jenkins installation as a application)*

1. Open AWS console and Select the instance and click on connect and click on SSH client tab click on copy symbol under example.
2. Open gitbash terminal where our pem file is located.
3. Now paste here and it asks for yes/no/fingerprint. , type YES.
4. Type sudo -i
5. Type the command like below:

docker run --name myc1 -d -p 8082:8080 jenkins:2.60.3

6.Type docker ps

7.Type docker images

8.Open aws console and select docker server instance

9.Copy the public ip and paste in browser like below:

< public ip >:8082

10. A browser tab will pop-up asking for the initial administrator password

11.Copy the path from browsers pop-up tab

12.Now open docker server terminal and type like below:

**docker exec -it <container id/ name>/bin/bash**

**13**.Type java -version

**Docker run --name madhu -d -p 8090:8080 jenkins:**