08/05/2023

Query retrieve join

CustomerName

OrderDate

ProductName

List<Object>

List<JoinWrapper>

Class JoinWrapper {

CustomerName

OrderDate

ProductName

}

CName orderDate ProductName

Steven 08/05/2023 TV

Class JoinWrapper {

Constructor(CustomerName:string

OrderDate:Date

ProductName:String ){}

}

VM ware software :

Virtual Machine

Base Machine with 16 RAM

1TB hard disk

Linux or Unix OS.

4GB

50GB

10OS

Docker : Docker is an advanced OS virtualization software platform

That makes it easier to create, deploy and run the application in Docker container.

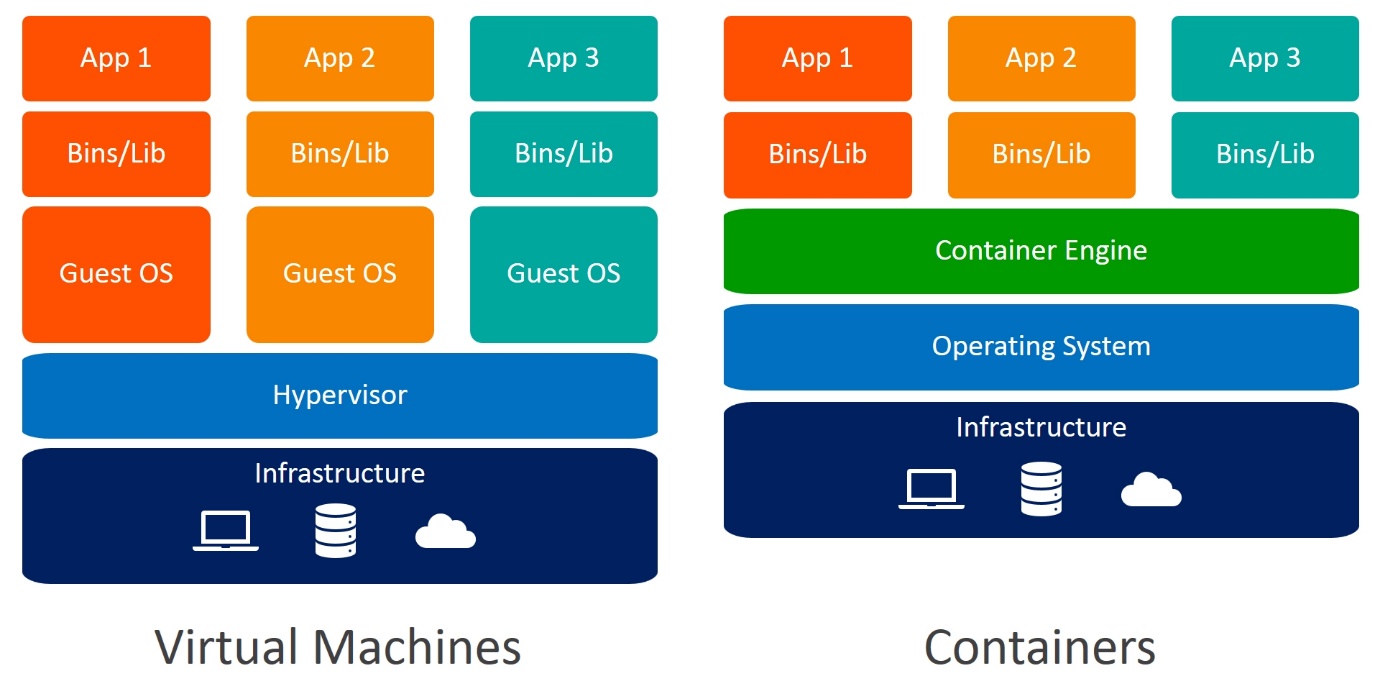
Virtualization lets you divide a system into a series of separate section, each one acting as distinct individual system or machine.

Using Docker we can create containerization application.

Virtualization is an abstract version of physical machine.

Containerization is an abstract version of an application or tool or software.

Docker container is responsible to run the application. Docker container is a part of docker engine.



@OneToMany(cascade=CascadeType.ALL)

List<Student> listofStd;

docker --version it is use to check the version of docker

docker images : it is use to check all images details present in local machine.

docker pull imagename this command is use to pull the image

docker pull hello-world

docker run imagename/imageId this command is use to run the image

docker hub account :

Docker hub is a like a git hub which help to publish our own images as well as

We can pull pre defined and user defined images in local machine.

In git hub we can push any type of data in Docker hub we need to publish or push

Docker images.

Docker Container : this is a running process or / instance of a images.

Docker Images : This file system and configuration of our application which are used to create a container.

Or

Docker images are the source code for our container.

Once we run the Docker images container become up and Container is responsible to execute the specific application mention in Docker file.

Docker file A docker file is a blue print or set of instruction that defines how our images is built.

.java

Docker file to display echo message through busybox images

Dockerfile

FROM busybox

CMD ["echo","Welcome to Docker Training"]

docker build -t imageName . -f Dockerfile

docker build -t cbfsd-busybox . -f Dockerfile

docker images

docker run cbfsd-busybox

Docker images to run simple java program

Demo.java

public class Demo {

    public static void main(String[] args) {

        System.out.println("Welcome to Java through Docker");

    }

}

Dockerfile

FROM openjdk:11

COPY Demo.java .

RUN javac Demo.java

CMD ["java","Demo"]

docker build -t cbfsd-java . -f Dockerfile

Creating docker image to run spring boot project

First create spring boot project and then

Create more than one rest api

Then using mvc package command create jar or war file.

Then create image with the help of

**Dockerfile**

FROM openjdk:11

COPY ./target/spring-boot-with-docker-0.0.1-SNAPSHOT.jar .

CMD ["java","-jar","spring-boot-with-docker-0.0.1-SNAPSHOT.jar"]

Command to create the images

**docker build -t cbfsd-springboot . -f Dockerfile**

To run the image if image is responsible to execute or run web app

**docker run -d -p 8080:8080 imageName/imageId**

docker run -d -p 8080:8080 cbfsd-springboot

docker run -d -p 8181:8080 --name my-spring-container cbfsd-springboot

docker ps This command is use to check all running container

docker ps -a This command display all container present in local machine (it may be running or stopped)

docker stop containerId/containerName : to suspend the running container

docker start containerId/containerName : to resumes the container

docker rm containerId –f

docker rmi imageName/imageId

creating the images for static web page

index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <style>

        div{background-color: burlywood;}

    </style>

</head>

<body>

    <div>

        Welcome to simple web page running throug Docker

    </div>

</body>

</html>

Dockerfile

FROM nginx

COPY index.html /usr/share/nginx/html

**docker build -t cbfsd-web-app . -f Dockerfile**

by default nginx server run on default port number 80.