DEVOPS ASSIGNMENT 2

By: Nithyashree Arunachalam(4NI19IS058)
Section - A

1. Difference between hypervisor and Docker

Hypervisor

It’s a visualization software in in cloud to host and allocate resources on various pieces of hardware. A hypervisor is a software that created and runs virtual machines. Its known as a virtual machine manager. The hypervisor separated the hypervisor OS and resources from the VMs and enables management of those VM.

It is a hardware virtualization technique that allows multiple guest operating system to run on a single host system is also known as Virtual machine manager. Multiple different operating systems can run alongside each other and share the same virtualized hardware resources with a hypervisor.

The hypervisor gives each virtual machine the resources that have been allocated and manages the scheduling of VM resources against the physical resources. The physical hardware still does the execution, so CPU is still executing CPU instructions as requested by the VMs, for example, while the hypervisor manages the schedule.

Docker

Docker is an open source platform which helps developers to build, deploy, run, update applications. It also allows management of containers which are standardized, executable components that combine application source code with the operating system libraries and dependencies required to run that code in any environment. Containers are made possible by process isolation and virtualization capabilities built into the Linux kernel.

S.no	Basis	Hypervisor	Docker
1.	**Mechanism**	Can be hardware as well as software. It can work on top of the OS.	Docker works on host kernel itself.Does not allow user to create multiple Operating systems.
2.	**Number of instances supported**	Allows user to generate multiple instances to the complete OSs.	Can run multiple applications or multiple instances of a single application.
3.	**Memory requirement**	Enables users to run multiple instances of complete operating systems. This makes them resource hungry and needs dedicated resources for any particular instance among the shared hardware.	Does not have any such requirements. One can create as many containers as needed,based on the application requirement and availability of processing power.
4.	**Boot-time**	A hypervisor might consume up to a minute to boot the OS and get up and running.	Docker can create containers in seconds, and users can get started in no time.
5.	**OS support**	OS agnostic- they can run across Windows, Mac, and Linux.	Dockers, on the other hand, are limited to Linux only. That, however, is not a deterrent for Dockers since Linux is a strong eco-system.

2. Difference between container and Virtual Machines

Virtual Machines

A **Virtual Machine** (VM) is a compute resource that uses software instead of a physical computer to run programs and deploy apps. One or more virtual "guest" machines run on a physical "host" machine. Each virtual machine runs its own operating system and functions separately from the other VMs, even when they are all running on the same host.

Container

Containers are a form of operating system virtualization. A single container might be used to run anything from a small microservice or software process to a larger application. Inside a container are all the necessary executables, binary code, libraries, and configuration files.

S.no	Container	Virtual Machine
1.	facilitate a way for virtualizing the operating system so that several workloads can execute on an individual operating system instance	The hardware is virtualized to execute several Operating system instances with VMs.
2.	Containers give services of OS from an underlying host and also separate the applications utilizing virtual-memory hardware.	VM is managed via hypervisor and uses VM hardware.
3.	Container facilitates the abstract operating system	VM facilitates the abstract machine which utilizes device drivers addressing an abstract machine.
4.	Lower overhead	Higher overhead
5.	Container only virtualizes the OS.	VM facilitates a way for virtualizing any computer system.
6.	Applications executing within the container environment contribute to an individual OS.	Applications executing on virtual machine system can execute distinct OS.
7.	Containers are very light	VMs have a large size.
8.	It is less secure.	It is highly secured
9.	Container examples: Containers via Docker, PhotonOS, RancherOS.	VM examples: VMware, Xen, KVM