Juhwan Lee
CS300: Elements of Software Engineering
Professor Fei Xie

Exercise 5.2 (Textbook Pg. 149):
Explain the fundamental difference between virtualization using a VM and virtualization using a container.

A hypervisor virtualizes physical hardware in traditional virtualization. As a result, each virtual machine has a guest OS, a virtual copy of the hardware that the OS needs to run, and an application with all of its associated libraries and dependencies. On the same physical server, multiple virtual machines running different operating systems can coexist. A VMware VM, for example, can coexist with a Linux VM, which in turn can coexist with a Microsoft VM, and so on.

Containers virtualize the operating system (typically Linux or Windows) rather than the underlying hardware, so each container only contains the application and its libraries and dependencies. Containers are small, fast, and portable because, unlike virtual machines, they do not require a guest OS in every instance and can instead rely on the host OS's features and resources.

Similar to virtual machines, containers allow developers to increase CPU and memory utilization on physical machines. However, containers go even further because they also enable microservices architectures that can be deployed and scaled with more granular application components. This is an attractive alternative to having to scale an entire monolithic application because a single component is struggling with the load.