**1. Alset and Atem are two software companies that are competitors. Alset uses a traditional software development lifecycle, while Atem uses containers. Which one of them would have a faster development and deployment cycle and why?**

Answer: Considering that Atem uses containers, I believe that their development and deployment cycles will be quicker. Customers would have to wait a lot longer for new version features because it would take years for the developer to create a new version of an application. Applications for a single environment were used to build them all. Long development cycles characterize traditional software. There aren't many ecosystems out there; this goes for the hardware as well as the operating system. There will be a gradual expansion of the conventional software development life cycle. But today, a lot has changed because services are now disconnected. We do not cram every component, including a network application, filesystem, and database, into a single program. For our work, we require a different setting. The methodology for developing software has evolved. Customers anticipate the release of new versions of the application quickly. Software that is adaptable to different situations can be produced by the company. Containers are the answer to the deployment issue. Where the software is built, there are numerous hardware environments. All systems will be compatible with the new version, and leveraging containers makes testing simple.

2: **Dr. Doom claims that his company can use VM images instead of Containers to do the same task. Is he correct in claiming so? Explain your answer.**

**Answer: The assertion made by Dr. Doom that he can complete the same operation using VM images is untrue because containers are easier to utilize than VM images. Software development and deployment would be more time-consuming if VM images were used. As compared to VM images, containers are substantially lighter. More resources are used by VM images than by containers. Containers use less disc space than virtual machine images. Due to the additional storage needed, this is a bad choice. Much faster than VM images are containers. Start-up times are quicker. Compared to containers, virtual machine images scale up slowly. Unlike VM images, containers boot up faster. Containers guarantee that apps launch in the same manner regardless of the infrastructure. Software deployment and development may become simpler as a result. Compared to VM images, containers can handle more traffic and workloads. VM images are less portable than containers.**