**Docker :**

Containerization platform all the necessary dependencies combined to form containers. to ensure that your application works in any environment.   
Containers run on the same machine sharing the same Operating system Kernel, this makes it

**Docker Vs VM (Virtual Machine)**

**Docker :**

* Docker shares the OS.
* Booting time will reduced.
* Process isolation is done at OS level.

**VM :**

* Each VM requiredsepate OS.
* Booting will take time.
* Process isolation is done at hardware level.

**Docker Images :** Docker images are source of container.Images are created with the build command, and they’ll produce a container when started with run.

**Docker Container :** Docker containers include the application and all of its dependencies, but share the kernel with other containers, running as isolated processes in user space on the host operating system.

**Docker Hub :** This is cloud registry, used to build , store and test the images .

**Docker Swarm :**

**Dockerfile :** Dockerfile is nothing but a set of instructions that have to be passed on to Docker itself, so that it can build images automatically.

**Can I use json instead of yaml for my compose file in Docker**?

Yes, we can with below format **docker-compose -f docker-compose.json u**

**Tell us how you have used Docker in your past position ?**

**Past experience**

**How far do Docker containers scale?**

Large web development like Google and twitter use at scale of hundreds , thousand or even millions of containers are running parallel.

**What platforms does Docker run on :** Runs on only linux and cloud platforms.

**Linux :**

* Ubuntu 12.04, 13.04 et al
* Fedora 19/20+
* RHEL 6.5+
* CentOS 6+

**Cloud :**

* AWS
* GCE

**Do I lose my data when the Docker container exits?**

No, data is saved in container until container deleted explicitly.

**What, in your opinion, is the most exciting potential use for Docker?**

Build Pipeline

**Why is Docker the new craze in virtualization and cloud computing?**

This is ultra weight containerization app.

**Why do my services take 10 seconds to recreate or stop?**

Sec is the default timeout for stop or recreate 10.

We can overcome as below

* Use Json form of command and also entry point in docker file.
* Use [“program”, “argument1”, “argument2”] instead of sending it as a plain string as like this – “program argument1 argument2”.
* Using the string form, makes Docker run the process using bash that can’t handle signals properly. Compose always uses the JSON form.
* Also set the stop\_signal to a proper signal that the application can understand and also know how to handle it

**How do I run multiple copies of a Compose file on the same host?**

**Docker compose use project name to create unique identifiers for all projects. In order to run multiple copies on same project need to set custom name using –p command line.**

**What’s the difference between up, run, and start?**

**Start : used to restart container which is previously created but stopped**

**Up :**  start or restart all the services defined in compose file.

**Run :** used to run adhoc task .  It requires the service name you want to run and only starts containers for services that the running service depends on. Use run to run tests or perform an administrative task such as removing or adding data to a data volume container.

**How many containers can run per host?**

The application size, available resources (like CPU, memory) will decide on the number of containers that can run on an environment.

**Is there a possibility to include specific code with COPY/ADD or a volume?**

You can add your code to the image using COPY or ADD directive in a Dockerfile. This is useful if you need to relocate your code along with the Docker image, for example when you’re sending the code to another environment (production, CI, etc).

You should use a volume if you want to make changes to your code and see them reflected immediately, for example when you’re developing code and your server supports hot code reloading or live-reload.

There may be cases where you’ll want to use both. You can have the image include the code using a COPY, and use a volume in your Compose file to include the code from the host during development. The volume overrides the directory contents of the image.

**What are the differences between the ‘docker run’ and the ‘docker create’?**

**What are the various states that a Docker container can be in at any given point in time?**

. Running  
• Paused  
• Restarting  
• Exited

**Can you remove a paused container from Docker?**

We cant remove the paused container, we need to stop then remove.

**Is there a possibility that a container can restart all by itself in Docker?**

No, The default –restart flag is set to never restart on its own.

**What is the preferred way of removing containers - ‘docker rm -f’ or ‘docker stop’ then followed by a ‘docker rm’?**

Docker stop and docker rm

**Difference between Docker Image and container?**

Docker images does not have any state and its state never changes , it is just an file and it is source of docker container.

Docker container is the runtime instance of docker image.

Docker compose : <https://docs.docker.com/compose/gettingstarted/#step-6-re-build-and-run-the-app-with-compose>

<https://www.edureka.co/blog/interview-questions/docker-interview-questions/>

Project : <https://www.oreilly.com/ideas/how-to-manage-docker-containers-in-kubernetes-with-java>

<https://www.joyent.com/blog/video-training-dockerize-applications>

<https://docs.oracle.com/cd/E24329_01/web.1211/e24446/jdbc.htm#INTRO217> : Web logic

Jenkins (Project ) : <https://jenkins.io/doc/tutorials/build-a-java-app-with-maven/>