1. What is the importance of Load Balance in Kubernetes?

A load balancer is needed because it gives a standard way to distribute network traffic among different services, which runs in the backend.

In Kubernetes, all the incoming traffic lands to a single IP address on the load balancer which is a way to expose your service to outside the internet which routes the incoming traffic to a particular pod (via service) using an algorithm known as round-robin. Even if any pod goes down load balances are notified so that the traffic is not routed to that particular unavailable node. Thus load balancers in Kubernetes are responsible for distributing a set of tasks (incoming traffic) to the pods,

1. What is the relationship between Kubernetes and Docker?

Kubernetes and Docker are both comprehensive de-facto solutions to intelligently manage containerized applications and provide powerful capabilities, and from this, some confusion has emerged. “Kubernetes” is now sometimes used as a shorthand for an entire container environment based on Kubernetes. In reality, they are not directly comparable, have different roots, and solve for different things.

Docker is a platform and tool for building, distributing and running Docker containers. It offers its own native clustering tool that can be used to orchestrate and schedule containers on machine clusters. Kubernetes is a container orchestration system for Docker containers that is more extensive than Docker Swarm and is meant to coordinate clusters of nodes at scale in production in an efficient manner. It works around the concept of pods, which are scheduling units (and can contain one or more containers) in the Kubernetes ecosystem, and they are distributed among nodes to provide high availability. One can easily run a Docker build on a Kubernetes cluster, but Kubernetes itself is not a complete solution and is meant to include custom plugins.

Kubernetes and Docker are both fundamentally different technologies but they work very well together, and both facilitate the management and deployment of containers in a distributed architecture.

1. What distinguishes Kubernetes from other containers?

* Kubernetes gives the user control over which server will host the container. It will be in charge of deciding how to launch. As a result, Kubernetes automates a variety of manual tasks.  
  Kubernetes is a container orchestration system that manages multiple clusters at the same time.
* It also offers additional services such as container management, security, networking, and storage.
* Kubernetes keeps track of nodes and containers’ health.
* Users can easily and quickly scale resources not only vertically but also horizontally with Kubernetes.

1. What exactly do you mean when you say heapster?

A Heapster is a metrics collection and performance monitoring system for data that are collected by the Kublet.

1. What exactly is a kubelet?

The kubelet is a service agent which controls and maintains group pf pods by checking pod specification using Kubernetes. The kubelet runs on each node and allows to communicate between a master node and a slave node.