1**. What is Kubernetes?**

[Kubernetes](https://www.simplilearn.com/tutorials/kubernetes-tutorial/what-is-kubernetes) is an open-source container orchestration tool that is used to automate tasks such as the management, monitoring, scaling, and deployment of containerized applications.

**2. What is Container Orchestration?**

Consider a scenario where you have 5-6 microservices for an application. Now, these microservices are put in individual containers, but won’t be able to communicate without container orchestration.

**3. What is Kubelet?**

This is an agent service which runs on each node and enables the slave to communicate with the master.

**4. Explain node port**

The node port service is a fundamental way to get external traffic to your service. It opens a particular port on all nodes and forwards network traffic sent to this port.

**5. What is Kubectl used for?**

Kubectl is a software for controlling Kubernetes clusters. Ctl stands for control, which is a command-line interface to pass the command to the cluster and manage the Kubernetes

component.

**6. What is minikube?**

Minikube is a software that helps the user to run Kubernetes. It runs on the single nodes that are inside VM on your computer. This tool is also used by programmers who are developing an application using Kubernetes.

**7. What do you know about clusters in Kubernetes?**

A Kubernetes cluster is a set of node machines for running containerized applications. At a minimum, a cluster contains a worker node and a master node. The master node is responsible for maintaining the desired state of the cluster, such as which applications are running and which container images they use. Worker nodes actually run the applications and workloads.

**8. What is the role of kube-apiserver and kube-scheduler?**

The kube — apiserver is the front-end of the master node control panel. It exposes all the APIs of the Kubernetes Master node components and establishes communication between the Kubernetes Node and the Kubernetes master components.

The kube-scheduler distributes and manages workload on the worker nodes. It selects the most suitable node to run the unscheduled pod based on resource requirement and keeps a track of resource utilization. It ensures that the workload is not scheduled on nodes that are already full.

**9. What is a node in Kubernetes?**

A node is the smallest fundamental unit of computing hardware. It represents a single machine in a cluster, which could be a physical machine in a data center or a virtual machine from a cloud provider. Each machine can substitute any other machine in a Kubernetes cluster. The master in Kubernetes controls the nodes that have containers

**10. What is a pod in Kubernetes?**

Pods are high-level structures that wrap one or more containers. This is because containers are not run directly in Kubernetes. Containers in the same pod share a local network and the same resources, allowing them to easily communicate with other containers in the same pod as if they were on the same machine while at the same time maintaining a degree of isolation.

**11. What are Daemon sets?**

A Daemon set is a set of pods that runs only once on a host. They are used for host layer attributes like a network or for monitoring a network, which you may not need to run on a host more than once.

**12. What is a Namespace in Kubernetes?**

Namespaces are used for dividing cluster resources between multiple users. They are meant for environments where there are many users spread across projects or teams and provide a scope of resources.

**13. Kubernetes controller manager?**

The controller manager is a daemon that is used for embedding core control loops, garbage collection, and Namespace creation. It enables the running of multiple processes on the master node even though they are compiled to run as a single process.

The primary controller managers that can run on the master node are the endpoints controller, service accounts controller, namespace controller, node controller, token controller, and replication controller.

**14. What is etcd?**

Kubernetes uses etcd as a distributed key-value store for all of its data, including metadata and configuration data, and allows nodes in Kubernetes clusters to read and write data. Etcd represents the state of a cluster at a specific moment in time and is a canonical hub for state management and cluster coordination of a Kubernetes cluster.

**15. What is ClusterIP?**

The ClusterIP is the default Kubernetes service that provides a service inside a cluster (with no external access) that other apps inside your cluster can access.

**16. What is the difference between a replica set and a replication controller?**

A replication controller monitors the pods and automatically restarts them if they fail. If the node fails, this controller will respawn all the pods of that node on another node. If the pods die, they won't be spawned again unless wrapped around a replica set.

Replica Set as the next-generation replication controller. This kind of support has some selector types and supports the equality-based and the set-based selectors.

It allows filtering by label values and keys. To match the object, they have to satisfy all the specified label constraints.

**17. What is a headless service?**

A headless service is used to interface with service discovery mechanisms without being tied to a ClusterIP, therefore allowing you to directly reach pods without having to access them through a proxy. It is useful when neither load balancing nor a single Service IP is required.

**18. What is Helm?**

HELM is an open-source package manager for Kubernetes, a powerful container orchestration platform for modern applications. It is officially owned by Kubernetes and managed by the Cloud Native Computing Foundation (CNCF).

It can be time consuming and tedious to write and maintain Kubernetes YAML manifests for all of the essential Kubernetes objects. Helm streamlines the process by generating a single package that can be distributed across your cluster.

Concepts used by Helm are:  
Chart – It is a package consists of pre configured Kubernetes Resources.  
Release – It is an instance that can be deployed to the Cluster with the help of Helm.  
Repository – It is a group of charts that are available for others.

**19. What are Helm Charts?**

Helm Charts is used for combining all the Kubernetes YAML Manifests in a single package which can also be advertised to our Kubernetes Cluster. Once we install Helm CHart in our cluster it is easy as running one single helm, that simplifies the deployment of the containerized Applications.