Exam Session - Knowledge Check: Kubernetes Basics

cloudacademy.com/quiz/exam/3761619/results

#1

_____ provide load balancing across a set of pods.

✗

Clusters

✗

Controllers

✓

Services

✗

Endpoints

Explanation

Services provide load balancing across a set of pods. For every Service object, the apiserver makes an endpoints resource available. You can view this resource with:

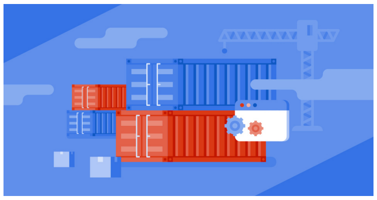
```
$ kubectl get endpoints ${SERVICE_NAME}
```

https://kubernetes.io/docs/tasks/debug-application-cluster/debug-application/

Covered in this lecture

Deployments

Course:Introduction to Kubernetes



11m

#2

You have deployed an application in Kubernetes. The application container exposes port 80. You need to be able to access the application from outside of the cluster. What Kubernetes resource should you use to meet this requirement?

✓

A service

✕
A binding

✕
A deployment

✕
An endpoint

Explanation

A service provides a mechanism for accessing a logical set of pods. You can use a service of type NodePort or LoadBalancer to allow external access to an application running in Kubernetes.

A binding is used for associating a role with a user to authorize actions the user is allowed to perform.

A deployment can deploy an application in the cluster, but it can't grant external access without a service.

An endpoint is a resource that a service automatically manages to keep track of the pods that are accessible via the service.

🔗/lab/deploy-a-stateless-application-in-a-kubernetes-cluster/deploying-a-stateless-application-in-the-kubernetes-cluster/
Covered in this lecture
Deploying a Microservices Application into EKS
Course:Introduction to AWS EKS

22m

#3

In Kubernetes, a(n) _____ is a group of one or more containers (such as Docker containers), the shared storage for those containers, and options about how to run the containers.

✕
namespace

✕
deployment

✓

pod

✗

replica set

Explanation

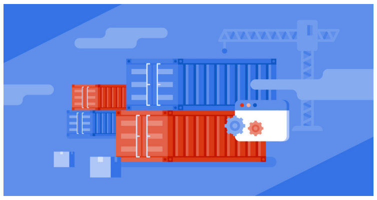
A *pod* (as in a pod of whales or pea pod) is a group of one or more containers (such as Docker containers), the shared namespaces for those containers, and options about how to run the containers. Pods are always co-located and co-scheduled, and run in a shared context. A pod models an application-specific “logical host” - it contains one or more application containers which are relatively tightly coupled — in a pre-container world, they would have executed on the same physical or virtual machine.

https://kubernetes.io/docs/user-guide/pods/

Covered in this lecture

Pods

Course:Introduction to Kubernetes



15m

#4

What is the recommended Kubernetes resource for running applications?

✗

Stand-alone Pod

✗

Container

✗

ReplicaSet

✓

Deployment

Explanation

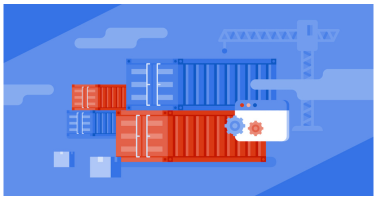
Deployments are a high-level concept for managing applications in Kubernetes. Pods are where applications actually run, but deployments make managing applications much simpler than working with pods directly. For example, you can easily scale and perform a rolling update to a new version using deployments.

Deployments automatically maintain a ReplicaSet for replicating pods, and are easier to work with than a ReplicaSet.

Containers aren't a type of Kubernetes resources.

/course/introduction-to-kubernetes/deployments/

Covered in this lecture
Probes
Course:Introduction to Kubernetes



11m

#5

Which two kubectl commands are useful for collecting information about any type of resource that is active in a Kubernetes cluster? (Choose 2 answers)

✓

get

✓

describe

✗

list

✗

watch

Explanation

The get and describe commands are useful for reporting information about active resources in a cluster.

list and watch are not kubectl commands. They are two examples of read verbs in the Kubernetes API. For example, kubectl sends list requests to the Kubernetes API to create the output of kubectl get commands.

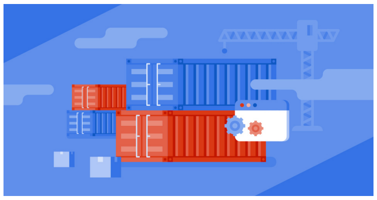
The explain, expose, and logs commands are kubectl commands but are not useful for gathering information about Kubernetes resources. The explain command provides information for understanding the fields of resources, but doesn't get information about resources running in the cluster. The expose command creates a service resource. The logs command get container logs. The logs may be useful for understanding the status of a Pod resources, but logs is not useful for all types of resources.

/lab/deploy-a-stateless-application-in-a-kubernetes-cluster/deploying-a-stateless-application-in-the-kubernetes-cluster/

Covered in this lecture

Multi-Container Pods

Course:Introduction to Kubernetes



10m

#6

What are some reasons you would prefer to use Deployments rather than "naked" Pods (Pods that are not managed by a higher-level resource, such as a Deployment) for managing applications in Kubernetes? (Select all that apply)

✓

Deployments are compatible with Horizontal Pod Autoscalers

✓

Deployments can reschedule pods that fail

✓

Deployments support rolling updates and rollbacks

✕

Deployments support a variety of restart policies

Explanation

Deployments can be autoscaled using Horizontal Pod Autoscalers, reschedule pods that fail, and perform rolling updates and rollbacks. Naked pods (pods without any higher-level resource managing them) cannot do any of those.

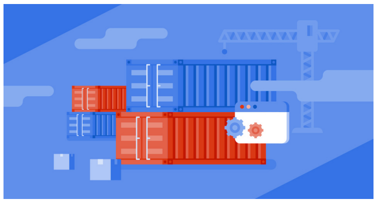
Pods can run multiple containers and support various restart policies, so those are not reasons to prefer Deployments over naked Pods.

/course/introduction-to-kubernetes/deployments/

Covered in this lecture

Deployments

Course:Introduction to Kubernetes



11m

#7

A _____ is a higher-level controller that automates rolling updates of applications declaratively.

✗

Scheduler

✗

FailoverController

✓

Deployment

✗

HAmaster

Explanation

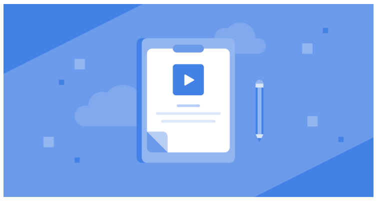
If you deploy applications with Replication Controllers, you should consider switching them to Deployments. A Deployment is a higher-level controller that automates rolling updates of applications declaratively, and therefore is recommended.

https://kubernetes.io/docs/concepts/workloads/controllers/deployment/

Covered in this lecture

Create GitHub Webhook for Automated Build & Deploys

Course:OpenShift 101 - Quick Start



10m

#8

You need to utilize a PersistentVolume for application storage in a Kubernetes cluster. What field of a PersistentVolume can you use to control the number of nodes that can mount the PersistentVolume for reading and writing?

✓
accessMode

✗
mountOptions

✗
sharingOptions

✗
nodeSelector

Explanation

A PersistentVolumes accessMode field controls how many nodes can mount it for reading and writing. The supported values are ReadWriteOnce, ReadOnlyMany, and ReadWriteMany.

🔗/lab/deploy-a-stateful-application-in-a-kubernetes-cluster/deploying-stateful-application-kubernetes-cluster/
#9

Which of the following statements related to Kubernetes storage are true? (Select all that apply)

✓
A Volume's lifetime is connected to the lifetime of a pod

✗
A PersistentVolume's lifetime is connected to the lifetime of a pod

✗
Volumes must be explicitly claimed by pods

✓
PersistentVolumes must be explicitly claimed by pods

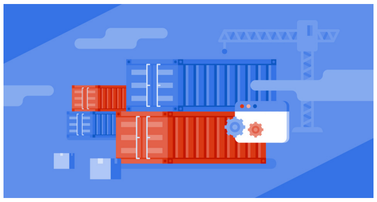
Explanation

A Volume's lifetime is the same as the lifetime of the pod that encloses it. PersistentVolumes have a lifetime independent of any pod allowing the data on a PersistentVolume to be reused by other pods.

PersistentVolumes must be claimed by pods using PersistentVolumeClaims. Volumes do not require the use of claims. Simply including a volume in a Pod spec is enough to create and access the volume in the pod.

Both Volumes and PersistentVolumes can be accessed by all of the containers in the pod enclosing them.

🔗/course/introduction-to-kubernetes/volumes/
Covered in this lecture
Volumes
Course:Introduction to Kubernetes



13m
🔖
#10

To update a service without an outage, kubectl supports what is called _____.

✗
high availability (HA) deployment

✓
rolling update

✗
automatic-hot failover

✗
automatic-warm failover

Explanation

To update a service without an outage, kubectl supports what is called ‘rolling update’, which updates one pod at a time, rather than taking down the entire service at the same time. Note that kubectl rolling-update only supports Replication Controllers.

🔗https://kubernetes.io/docs/tutorials/kubernetes-basics/update/update-intro/
Covered in this lecture
Rolling Updates and Rollbacks
Course:Introduction to Kubernetes

8m