



Linux-Foundation

Exam Questions KCNA

Kubernetes and Cloud Native Associate (KCNA)

NEW QUESTION 1

kubeadm is an administrative dashboard for kubernetes

- A. False
- B. True

Answer: A

Explanation:

<https://kubernetes.io/docs/reference/setup-tools/kubeadm/>

Graphical user interface, text, application Description automatically generated

Kubeadm

Kubeadm is a tool built to provide `kubeadm init` and `kubeadm join` as best-practice "fast paths" for creating Kubernetes clusters.

kubeadm performs the actions necessary to get a minimum viable cluster up and running. By design, it cares only about bootstrapping, not about provisioning machines. Likewise, installing various nice-to-have addons, like the Kubernetes Dashboard, monitoring solutions, and cloud-specific addons, is not in scope.

Instead, we expect higher-level and more tailored tooling to be built on top of kubeadm, and ideally, using kubeadm as the basis of all deployments will make it easier to create conformant clusters.



NEW QUESTION 2

How to create deployment name app-dep, image=nginx, and replicas 5 using imperative command?

- A. `kubectrl create app-dep deployment --image=nginx --replicas=5`
- B. `kubectrl create deployment app-dep --image=nginx --replicas=5`
- C. `kubectrl create app-dep deployment --replicas=5 --image=nginx`

Answer: B

Explanation:

<https://kubernetes.io/docs/reference/generated/kubectrl/kubectrl-commands#-em-deployment-em-> Text Description automatically generated with medium confidence

Create a deployment named my-dep that runs the nginx image with 3 replicas

```
kubectrl create deployment my-dep --image=nginx --replicas=3
```

NEW QUESTION 3

What does CNCF stand for?

- A. Cloud Native Computing Foundation
- B. Cloud Native Cloud Foundation
- C. Cloud Native Container Foundation

Answer: A

Explanation:

<https://www.cncf.io/about/who-we-are/>

Graphical user interface, text, application Description automatically generated

The Cloud Native Computing Foundation (CNCf) hosts critical components of the global technology infrastructure. CNCf brings together the world's top developers, end users, and vendors and runs the largest open source developer conferences. CNCf is part of the nonprofit Linux Foundation.

NEW QUESTION 4

What is the main difference between Argo vs. Flux CD?

- A. Argo is pull-based, and Flux is push-based
- B. No difference; both are pull-based
- C. Argo is push-based, and Flux is pull-based
- D. No difference; both are push-based

Answer: C

Explanation:

ArgoCD:

<https://argo-cd.readthedocs.io/en/stable/developer-guide/ci/#can-i-retrigger-the-checks-without-pushing-a-new-c>

FluxCD: <https://fluxcd.io/>

NEW QUESTION 5

What is the primary interface for Kubernetes cluster?

- A. Kubernetes Api
- B. Kubelet
- C. YAML
- D. Control Plane
- E. JSON

Answer: A

Explanation:

<https://kubernetes.io/docs/concepts/overview/components/#kube-apiserver>

A screenshot of a computer Description automatically generated with medium confidence

kube-apiserver

The API server is a component of the Kubernetes control plane that exposes the Kubernetes API. The API server is the front end for the Kubernetes control plane.

The main implementation of a Kubernetes API server is [kube-apiserver](#). kube-apiserver is designed to scale horizontally—that is, it scales by deploying more instances. You can run several instances of kube-apiserver and balance traffic between those instances.

NEW QUESTION 6

Which kubernetes resource type allows defining which pods are isolated when it comes to network-ing?

- A. Network policy
- B. Domain Name System 'DNS'
- C. Role Binding
- D. Service

Answer: A

Explanation:

<https://kubernetes.io/docs/concepts/services-networking/network-policies/#the-two-sorts-of-pod-isolation> Text, letter Description automatically generated

The Two Sorts of Pod Isolation

There are two sorts of isolation for a pod: isolation for egress, and isolation for ingress. They concern what connections may be established. "Isolation" here is not absolute, rather it means "some restrictions apply". The alternative, "non-isolated for \$direction", means that no restrictions apply in the stated direction. The two sorts of isolation (or not) are declared independently, and are both relevant for a connection from one pod to another.

By default, a pod is non-isolated for egress; all outbound connections are allowed. A pod is isolated for egress if there is any NetworkPolicy that both selects the pod and has "Egress" in its `policyTypes`; we say that such a policy applies to the pod for egress. When a pod is isolated for egress, the only allowed connections from the pod are those allowed by the `egress` list of some NetworkPolicy that applies to the pod for egress. The effects of those `egress` lists combine additively.

By default, a pod is non-isolated for ingress; all inbound connections are allowed. A pod is isolated for ingress if there is any NetworkPolicy that both selects the pod and has "Ingress" in its `policyTypes`; we say that such a policy applies to the pod for ingress. When a pod is isolated for ingress, the only allowed connections into the pod are those from the pod's node and those allowed by the `ingress` list of some NetworkPolicy that applies to the pod for ingress. The effects of those `ingress` lists combine additively.

NEW QUESTION 7

What can you use to add new resource types to your cluster?

- A. start container
- B. CustomResourceDefinitions
- C. init container
- D. Flux
- E. CRI-O

Answer: B

Explanation:

<https://kubernetes.io/docs/concepts/extend-kubernetes/api-extension/custom-resources/> Graphical user interface, text, application Description automatically generated

CustomResourceDefinitions

The [CustomResourceDefinition](#) API resource allows you to define custom resources. Defining a CRD object creates a new custom resource with a name and schema that you specify. The Kubernetes API serves and handles the storage of your custom resource. The name of a CRD object must be a valid [DNS subdomain name](#).

This frees you from writing your own API server to handle the custom resource, but the generic nature of the implementation means you have less flexibility than with [API server aggregation](#).

Refer to the [custom controller example](#) for an example of how to register a new custom resource, work with instances of your new resource type, and use a controller to handle events.

NEW QUESTION 8

The Kubernetes rolling update is used for _____ .

- A. Updating a service
- B. Scaling an application
- C. Updating a deployment

Answer: C

Explanation:

<https://kubernetes.io/docs/tutorials/kubernetes-basics/update/update-intro/>

Graphical user interface, text Description automatically generated with medium confidence

Performing a Rolling Update

Objectives

- Perform a rolling update using kubectl.

Updating an application

Users expect applications to be available all the time and developers are expected to deploy new versions of them several times a day. In Kubernetes this is done with rolling updates. **Rolling updates** allow Deployments' update to take place with zero downtime by incrementally updating Pods instances with new ones. The new Pods will be scheduled on Nodes with available resources.

In the previous module we scaled our application to run multiple instances. This is a requirement for performing updates without affecting application availability. By default, the maximum number of Pods that can be unavailable during the update and the maximum number of new Pods that can be created, is one. Both options can be configured to either numbers or percentages (of Pods). In Kubernetes, updates are versioned and any Deployment update can be reverted to a previous (stable) version.

Summary:

- Updating an app

Rolling updates allow Deployments' update to take place with zero downtime by incrementally updating Pods instances with new ones.

NEW QUESTION 9

What is container orchestration?

- A. Packaging code and all of its dependencies into a single executable
- B. Adding code to a container image so it can run as a container
- C. Using automation to manage containers
- D. Spinning a new containers to replace old ones

Answer: C

Explanation:

<https://www.redhat.com/en/topics/containers/what-is-container-orchestration> Text Description automatically generated

Container orchestration automates the deployment, management, scaling, and networking of containers. Enterprises that need to deploy and manage hundreds or thousands of **Linux® containers** and hosts can benefit from container orchestration.

Container orchestration can be used in any environment where you use containers. It can help you to deploy the same application across different environments without needing to redesign it. And **microservices** in containers make it easier to orchestrate services, including storage, networking, and security.

NEW QUESTION 10

A new Pod is created. Then, the Pod is assigned to a Node. Which Kubernetes component was re-sponsible for determining which Node to assign the Pod to?

- A. kubelet
- B. Scheduler
- C. API Server
- D. Controller manager

Answer: B

Explanation:

<https://kubernetes.io/docs/reference/command-line-tools-reference/kube-scheduler/> Graphical user interface, text, application Description automatically generated

The Kubernetes scheduler is a control plane process which assigns Pods to Nodes. The scheduler determines which Nodes are valid placements for each Pod in the scheduling queue according to constraints and available resources. The scheduler then ranks each valid Node and binds the Pod to a suitable Node. Multiple different schedulers may be used within a cluster; kube-scheduler is the reference implementation. See [scheduling](#) for more information about scheduling and the kube-scheduler component.

```
kube-scheduler [flags]
```

NEW QUESTION 10

A is a ready-to-run software package, containing everything needed to run an application.

- A. Container Repository
- B. Container Runtime
- C. Docker
- D. Container Image

Answer: D

Explanation:

<https://kubernetes.io/docs/concepts/containers/#container-images> Text, letter Description automatically generated

Container images

A [container image](#) is a ready-to-run software package, containing everything needed to run an application: the code and any runtime it requires, application and system libraries, and default values for any essential settings.

By design, a container is immutable: you cannot change the code of a container that is already running. If you have a containerized application and want to make changes, you need to build a new image that includes the change, then recreate the container to start from the updated image.

NEW QUESTION 12

What is the command used to scale the application?

- A. kubectl run
- B. kubectl explain
- C. kubectl scale

Answer: C

Explanation:

<https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#scale> Graphical user interface, text, application, email Description automatically generated

scale

Set a new size for a deployment, replica set, replication controller, or stateful set.

Scale also allows users to specify one or more preconditions for the scale action.

If `--current-replicas` or `--resource-version` is specified, it is validated before the scale is attempted, and it is guaranteed that the precondition holds true when the scale is sent to the server.

Usage

```
$ kubectl scale [--resource-version=version] [--current-replicas=count] --replicas=COUNT (-f
FILENAME | TYPE NAME)
```



NEW QUESTION 14

Which of the following best describes a cloud-native app?

- A. An application where all logic is coded into a single large binary.
- B. An application that publishes an HTTPS web front-end.
- C. An application that takes advantages of cloud computing frameworks and their loosely coupled cloud services.
- D. An application that leverages services that are native to public cloud platforms such as Azure, GCP, and/or AWS.

Answer: C

Explanation:

Cloud-native apps leverage cloud computing frameworks and tend to be microservices based, where individual components of the app are coded as individual.

NEW QUESTION 17

To run a startup task before a Pod's container starts up. What Kubernetes feature can help you accomplish this?

- A. Init container
- B. Sidecar container
- C. Startup probe
- D. DaemonSet

Answer: A

Explanation:

<https://kubernetes.io/docs/concepts/workloads/pods/init-containers/> Text, letter Description automatically generated

A Pod can have multiple containers running apps within it, but it can also have one or more init containers, which are run before the app containers are started.

Init containers are exactly like regular containers, except:

- Init containers always run to completion.
- Each init container must complete successfully before the next one starts.

If a Pod's init container fails, the kubelet repeatedly restarts that init container until it succeeds. However, if the Pod has a `restartPolicy` of `Never`, and an init container fails during startup of that Pod, Kubernetes treats the overall Pod as failed.

To specify an init container for a Pod, add the `initContainers` field into the Pod specification, as an array of `container` items (similar to the app `containers` field and its contents). See [Container](#) in the API reference for more details.

NEW QUESTION 20

Which Kubernetes resource creates Kubernetes Jobs?

- A. JobFactory
- B. CronJob
- C. Task
- D. JobDeployment

Answer: B

Explanation:<https://kubernetes.io/docs/concepts/workloads/controllers/cron-jobs/>

Graphical user interface, text, application, email Description automatically generated

CronJob

FEATURE STATE: Kubernetes v1.21 [stable]*A CronJob creates Jobs on a repeating schedule.*

One CronJob object is like one line of a *crontab* (cron table) file. It runs a job periodically on a given schedule, written in **Cron** format.

NEW QUESTION 21

What is the command to list all the available objects in your Kubernetes cluster?

- A. kubectl get all
- B. kubectl get api-resources
- C. kubectl api-resources
- D. kubectl get pods

Answer: C**Explanation:**<https://kubernetes.io/docs/reference/kubectl/cheatsheet/>

Graphical user interface, text, application, email Description automatically generated

Resource types

List all supported resource types along with their shortnames, **API group**, whether they are **namespaced**, and **Kind**:

```
kubectl api-resources
```

NEW QUESTION 25

What command can you use to get documentation about a resource type from the command line?

- A. kubectl api-resources
- B. kubectl explain
- C. kubectl get
- D. kubectl get-resource

Answer: B**Explanation:**<https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#explain> Graphical user interface, text, application, email Description automatically generated

explain

List the fields for supported resources.

This command describes the fields associated with each supported API resource. Fields are identified via a simple JSONPath Identifier:

```
<type>.<fieldName>[.<fieldName>]
```

Add the `--recursive` flag to display all of the fields at once without descriptions. Information about each field is retrieved from the server in OpenAPI format.

Use "kubectl api-resources" for a complete list of supported resources.

Usage

```
$ kubectl explain RESOURCE
```



NEW QUESTION 29

How would you return all the pod data in the json format using kubectl command?

- A. kubectl get pods -o json
- B. kubectl get pods --all-namespaces
- C. kubectl get pods -o wide
- D. kubectl get pods -o jsonpath

Answer: A

Explanation:

<https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#get>

NEW QUESTION 34

How should folks new to the cloud native ecosystem, go about learning the different aspects of the ecosystem?

- A. by signing up the CNCF slack
- B. by reading the Kubernetes documentation
- C. by looking at the cloud native landscape
- D. by looking at the cloud native trail-map

Answer: D

Explanation:

<https://github.com/cncf/landscape#trail-map>

NEW QUESTION 36

Flux is built using which toolkit?

- A. CI/CD
- B. DevSecOps
- C. GitOps
- D. DevOps

Answer: C

Explanation:

<https://fluxcd.io/>
Graphical user interface, text, application Description automatically generated



NEW QUESTION 37

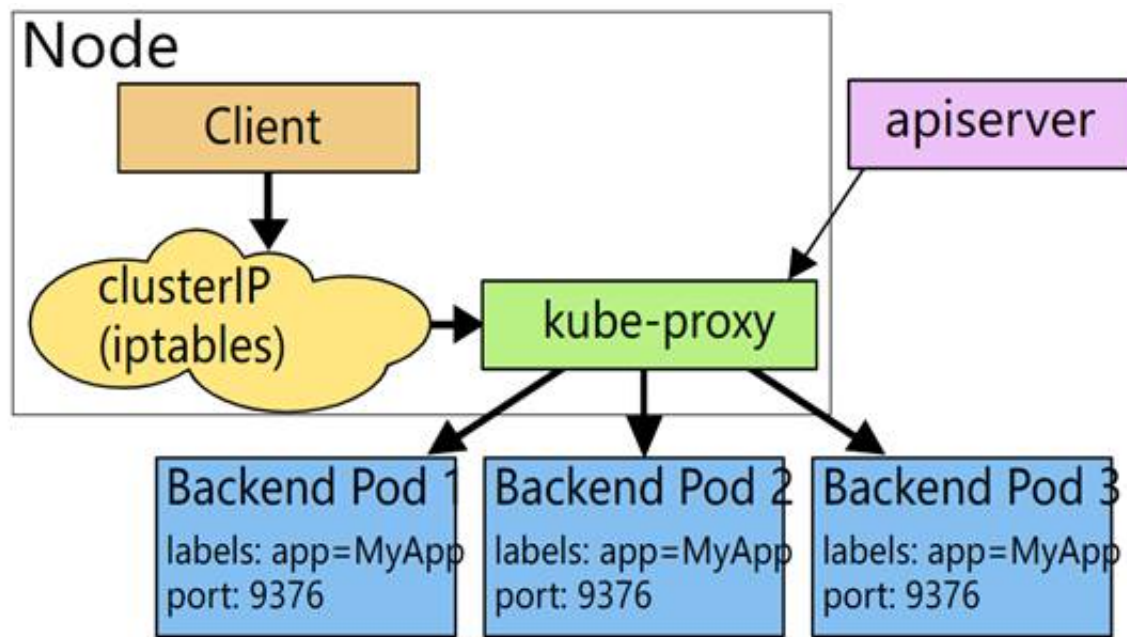
How does service logical group set of pods?

- A. Using hostname
- B. Using label and selectors
- C. Using IP address

Answer: B

Explanation:

<https://kubernetes.io/docs/concepts/services-networking/service/> Diagram Description automatically generated



NEW QUESTION 40

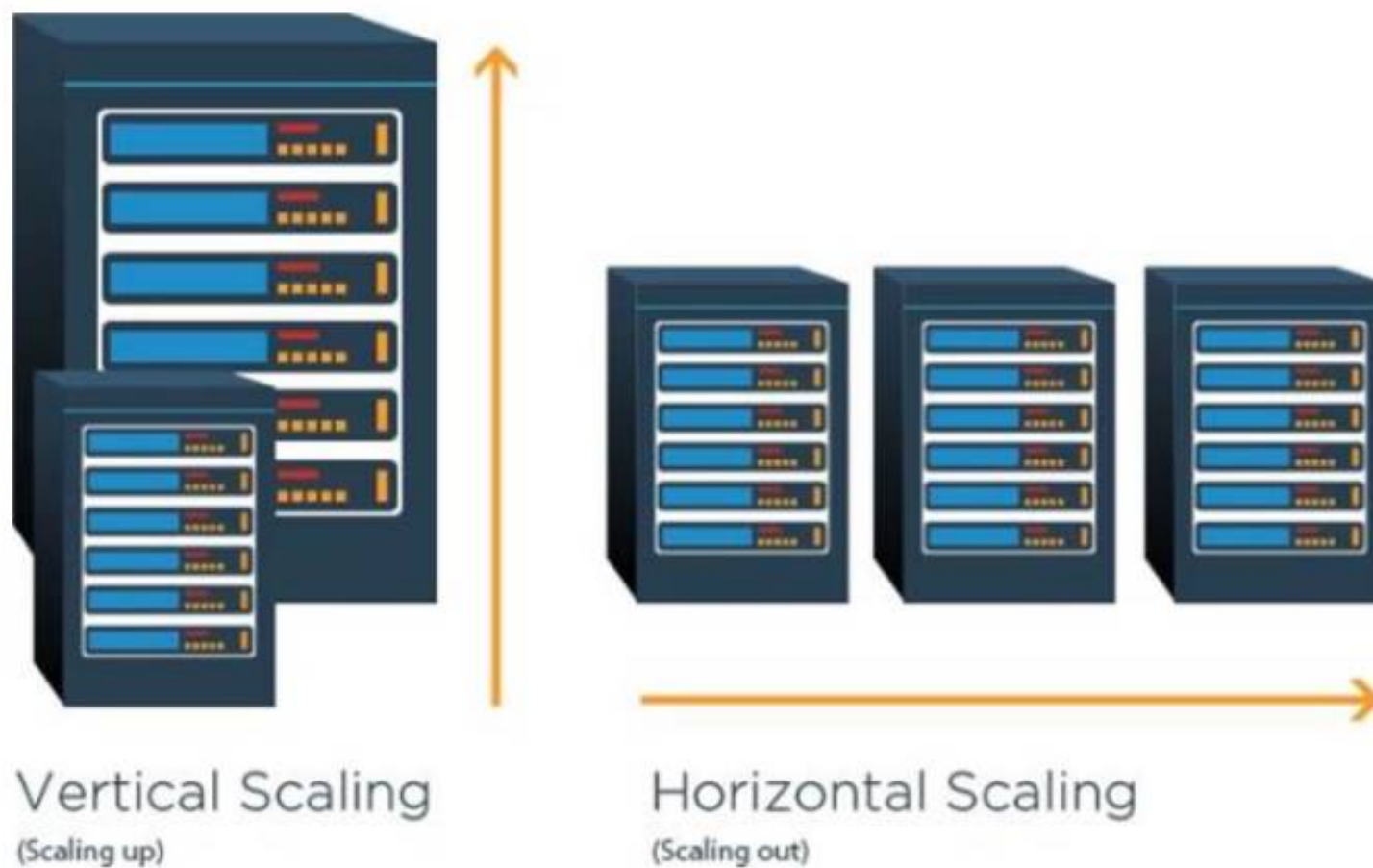
An application that is nearing its usage limit. To increase the amount of users it can handle, you allo-cate additional memory resources to each instance of the application. What type of scaling is this?

- A. Horizontal Scaling
- B. Cluster Autoscaling
- C. Recursive Scaling
- D. Vertical Scaling

Answer: D

Explanation:

Graphical user interface, diagram Description automatically generated



NEW QUESTION 41

Which part of a Kubernetes cluster is responsible for running container workloads?

- A. Worker Node
- B. kube-proxy
- C. Control plane
- D. etcd

Answer: A

Explanation:

Worker Nodes are responsible for executing containerized workloads.

NEW QUESTION 43

What makes cloud native technology so important?

- A. It makes data centric
- B. It strengthens team

- C. It removes roadblocks to innovation
- D. It helps gather software requirements
- E. It makes operational centric

Answer: C

Explanation:

<https://github.com/cncf/foundation/blob/main/charter.md>

Graphical user interface, text, application Description automatically generated

Cloud native technologies empower organizations to build and run scalable applications in modern, dynamic environments such as public, private, and hybrid clouds. Containers, service meshes, microservices, immutable infrastructure, and declarative APIs exemplify this approach.

These techniques enable loosely coupled systems that are resilient, manageable, and observable. Combined with robust automation, they allow engineers to make high-impact changes frequently and predictably with minimal toil.

NEW QUESTION 47

Which of the following is an advantage a cloud-native microservices application has over monolithic applications?

- A. Cloud-native microservices applications tend to be faster and more responsive than monolithic applications.
- B. Cloud-native microservice applications tend to be easier to troubleshoot.
- C. Cloud-native microservice applications tend to be easier to scale and perform updates on.

Answer: C

Explanation:

Cloud-native applications tend to be microservice base, they have individual services that can be independently scaled, updated and rolled back. This makes scaling and update operations simpler and less risky.

NEW QUESTION 52

What is the default service type in Kubernetes?

- A. ClusterIP
- B. NodePort
- C. serviceType
- D. loadBalancer

Answer: A

Explanation:

<https://kubernetes.io/docs/concepts/services-networking/service/#publishing-services-service-types> Graphical user interface, text, application, email Description automatically generated

Kubernetes ServiceTypes allow you to specify what kind of Service you want. The default is ClusterIP .

Type values and their behaviors are:

- **ClusterIP** : Exposes the Service on a cluster-internal IP. Choosing this value makes the Service only reachable from within the cluster. This is the default ServiceType .
- **NodePort**: Exposes the Service on each Node's IP at a static port (the NodePort). A ClusterIP Service, to which the NodePort Service routes, is automatically created. You'll be able to contact the NodePort Service, from outside the cluster, by requesting <NodeIP>:<NodePort> .
- **LoadBalancer**: Exposes the Service externally using a cloud provider's load balancer. NodePort and ClusterIP Services, to which the external load balancer routes, are automatically created.
- **ExternalName**: Maps the Service to the contents of the externalName field (e.g. foo.bar.example.com), by returning a CNAME record with its value. No proxying of any kind is set up.

NEW QUESTION 56

Which of the following is not a stop on the cloud native trailmap?

- A. Microservices
- B. CI/CD
- C. Containerization
- D. Software distribution

Answer: A

Explanation:

<https://github.com/cncf/landscape#trail-map>

NEW QUESTION 61

Which of the following is not the required field to describe Kubernetes objects?

- A. metadata
- B. apiVersion
- C. Kind
- D. Container
- E. spec

Answer: D

Explanation:

<https://kubernetes.io/docs/concepts/overview/working-with-objects/kubernetes-objects/> Graphical user interface, text, application Description automatically generated

Required Fields

In the `.yaml` file for the Kubernetes object you want to create, you'll need to set values for the following fields:

- `apiVersion` - Which version of the Kubernetes API you're using to create this object
- `kind` - What kind of object you want to create
- `metadata` - Data that helps uniquely identify the object, including a `name` string, `UID`, and optional `namespace`
- `spec` - What state you desire for the object

The precise format of the object `spec` is different for every Kubernetes object, and contains nested fields specific to that object. The [Kubernetes API Reference](#) can help you find the `spec` format for all of the objects you can create using Kubernetes.

NEW QUESTION 63

You might need to run a stateless application in kubernetes, and you want to be able to scale easily and perform rolling updates. What kubernetes resource type can you use to do this

- A. Dameon set
- B. Replica set
- C. Deployment
- D. pod
- E. service
- F. Stateful set

Answer: C

Explanation:

<https://kubernetes.io/docs/concepts/workloads/controllers/deployment/> Graphical user interface, text, application Description automatically generated

Deployments

A *Deployment* provides declarative updates for Pods and ReplicaSets.

You describe a *desired state* in a Deployment, and the Deployment Controller changes the actual state to the desired state at a controlled rate. You can define Deployments to create new ReplicaSets, or to remove existing Deployments and adopt all their resources with new Deployments.

Note: Do not manage ReplicaSets owned by a Deployment. Consider opening an issue in the main Kubernetes repository if your use case is not covered below.

NEW QUESTION 64

Observability and monitoring are not the same?

- A. True
- B. False

Answer: A

NEW QUESTION 65

What is autoscaling?

- A. Automatically measuring resource usage
- B. Automatically assigning workloads to nodes in a cluster
- C. Automatically repairing broken application instances
- D. Automatically adding or removing compute resources as needed

Answer: D

Explanation:

<https://kubernetes.io/blog/2016/07/autoscaling-in-kubernetes/>

Autoscaling means automatically scaling up or down in response to real-time usage data.

NEW QUESTION 66

Continuous delivery is .

- A. Manually deploying the code
- B. Coding, Building and Testing the code
- C. Automatically deploying code to [container or server] environment

Answer: C

NEW QUESTION 68

What is the name for a service that has no clusterIp address?

- A. Headless
- B. NodePort
- C. ClusterIP
- D. LoadBalancer

Answer: A

Explanation:

<https://kubernetes.io/docs/concepts/services-networking/service/#headless-services>

Text, letter Description automatically generated

Headless Services

Sometimes you don't need load-balancing and a single Service IP. In this case, you can create what are termed "headless" Services, by explicitly specifying "None" for the cluster IP (`.spec.clusterIP`).

You can use a headless Service to interface with other service discovery mechanisms, without being tied to Kubernetes' implementation.

For headless Services , a cluster IP is not allocated, kube-proxy does not handle these Services, and there is no load balancing or proxying done by the platform for them. How DNS is automatically configured depends on whether the Service has selectors defined:

NEW QUESTION 69

What kubectl command is used to edit a resource on the server?

- A. kubectl resource modify
- B. kubectl update resource
- C. kubectl edit
- D. kubectl resource edit

Answer: C

Explanation:

<https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#edit> Graphical user interface, text, application, email Description automatically generated

edit

Edit a resource from the default editor.

The edit command allows you to directly edit any API resource you can retrieve via the command-line tools. It will open the editor defined by your KUBE_EDITOR, or EDITOR environment variables, or fall back to 'vi' for Linux or 'notepad' for Windows. You can edit multiple objects, although changes are applied one at a time. The command accepts file names as well as command-line arguments, although the files you point to must be previously saved versions of resources.

Editing is done with the API version used to fetch the resource. To edit using a specific API version, fully-qualify the resource, version, and group.

The default format is YAML. To edit in JSON, specify "-o json".

The flag --windows-line-endings can be used to force Windows line endings, otherwise the default for your operating system will be used.

In the event an error occurs while updating, a temporary file will be created on disk that contains your unapplied changes. The most common error when updating a resource is another editor changing the resource on the server. When this occurs, you will have to apply your changes to the newer version of the resource, or update your temporary saved copy to include the latest resource version.

Edit the service named 'registry'

```
kubectl edit svc/registry
```

Use an alternative editor

```
KUBE_EDITOR="nano" kubectl edit svc/registry
```

Edit the job 'myjob' in JSON using the v1 API format

```
kubectl edit job.v1.batch/myjob -o json
```

Edit the deployment 'mydeployment' in YAML and save the modified config in its annotation

```
kubectl edit deployment/mydeployment -o yaml --save-config
```

Edit the deployment/mydeployment's status subresource

```
kubectl edit deployment mydeployment --subresource=status
```

NEW QUESTION 74

What cloud-native construct does a kubernetes pod wrap?

- A. Container
- B. Virtual Machine (VM)
- C. side car process
- D. Docker image

Answer: A

Explanation:

Kubernetes is an orchestrator of containerized apps. However, containers must be wrapped in pods before they can be deployed on kubernetes.

NEW QUESTION 79

What is horizontal scaling?

- A. Creating a Deployment
- B. Adding resources to existing apps and servers
- C. Moving workloads from one server to another
- D. Adding additional replicas of apps and servers

Answer: D

Explanation:

<https://kubernetes.io/docs/tasks/run-application/horizontal-pod-autoscale/> Text, letter Description automatically generated

In Kubernetes, a *HorizontalPodAutoscaler* automatically updates a workload resource (such as a Deployment or StatefulSet), with the aim of automatically scaling the workload to match demand.

Horizontal scaling means that the response to increased load is to deploy more Pods. This is different from *vertical* scaling, which for Kubernetes would mean assigning more resources (for example: memory or CPU) to the Pods that are already running for the workload.

If the load decreases, and the number of Pods is above the configured minimum, the HorizontalPodAutoscaler instructs the workload resource (the Deployment, StatefulSet, or other similar resource) to scale back down.

Horizontal pod autoscaling does not apply to objects that can't be scaled (for example: a DaemonSet.)

The HorizontalPodAutoscaler is implemented as a Kubernetes API resource and a controller. The resource determines the behavior of the controller. The horizontal pod autoscaling controller, running within the Kubernetes control plane, periodically adjusts the desired scale of its target (for example, a Deployment) to match observed metrics such as average CPU utilization, average memory utilization, or any other custom metric you specify.

NEW QUESTION 80

The three typical opentelemetry data is?

- A. Metrics
- B. Traces
- C. Logs
- D. All of the options

Answer: D

Explanation:

<https://opentelemetry.io/docs/concepts/data-sources/> Text Description automatically generated

What is OpenTelemetry?

OpenTelemetry is a set of APIs, SDKs, tooling and integrations that are designed for the creation and management of *telemetry data* such as traces, metrics, and logs. The project provides a vendor-agnostic implementation that can be configured to send telemetry data to the backend(s) of your choice. It supports a variety of popular open-source projects including Jaeger and Prometheus.

NEW QUESTION 81

Which of the following best describes the way kubernetes Role-based access control (RBAC) works?

- A. Kubernetes does not do RBAC
- B. Kubernetes RBAC states which users can perform which actions against which re-source
- C. Kubernetes RBAC lists which operations on which resources are denied to users
- D. Kubernetes RBAC is responsible for authenticating subjects such as users and groups

Answer: B

Explanation:

<https://kubernetes.io/docs/reference/access-authn-authz/rbac/>

Graphical user interface, text, application, email Description automatically generated

Using RBAC Authorization

Role-based access control (RBAC) is a method of regulating access to computer or network resources based on the roles of individual users within your organization.

RBAC authorization uses the `rbac.authorization.k8s.io` API group to drive authorization decisions, allowing you to dynamically configure policies through the Kubernetes API.

To enable RBAC, start the API server with the `--authorization-mode` flag set to a comma-separated list that includes `RBAC` ; for example:

```
kube-apiserver --authorization-mode=Example,RBAC --other-options --more-options
```

NEW QUESTION 84

Which kubernetes object do deployments use behind the scenes when they need to scale pods?

- A. Horizontal pod autoscaler
- B. ReplicaSets
- C. kubectl
- D. Replication controller

Answer: B

Explanation:

<https://kubernetes.io/docs/concepts/workloads/controllers/replicaset/>

Graphical user interface, text, application, email Description automatically generated

ReplicaSet

A ReplicaSet's purpose is to maintain a stable set of replica Pods running at any given time. As such, it is often used to guarantee the availability of a specified number of identical Pods.

NEW QUESTION 87

Which of the following is NOT a Kubernetes component?

- A. Scheduler
- B. Docker
- C. Cloud Controller manager
- D. Kube-proxy

Answer: B

Explanation:

Docker is not a Kubernetes component.

NEW QUESTION 91

What is the functionality of the daemon set?

- A. To run a copy of the pod in all the nodes of the cluster
- B. To initialize the pod before starting the main pod
- C. To run a copy of the pod in a single node of the cluster

Answer: A

Explanation:

<https://kubernetes.io/docs/concepts/workloads/controllers/daemonset/>

Graphical user interface, text, application Description automatically generated with medium confidence

DaemonSet

A *DaemonSet* ensures that all (or some) Nodes run a copy of a Pod. As nodes are added to the cluster, Pods are added to them. As nodes are removed from the cluster, those Pods are garbage collected. Deleting a DaemonSet will clean up the Pods it created.

Some typical uses of a DaemonSet are:

- running a cluster storage daemon on every node
- running a logs collection daemon on every node
- running a node monitoring daemon on every node

NEW QUESTION 93

'kubectl delete -n my-ns po,svc --all' will delete pods and services including uninitialized ones in the namespace 'my-ns'

- A. FALSE
- B. TRUE

Answer: B

Explanation:

<https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#delete> Graphical user interface, text, application, email Description automatically generated

IMPORTANT: Force deleting pods does not wait for confirmation that the pod's processes have been terminated, which can leave those processes running until the node detects the deletion and completes graceful deletion. If your processes use shared storage or talk to a remote API and depend on the name of the pod to identify themselves, force deleting those pods may result in multiple processes running on different machines using the same identification which may lead to data corruption or inconsistency. Only force delete pods when you are sure the pod is terminated, or if your application can tolerate multiple copies of the same pod running at once. Also, if you force delete pods, the scheduler may place new pods on those nodes before the node has released those resources and causing those pods to be evicted immediately.

Note that the delete command does NOT do resource version checks, so if someone submits an update to a resource right when you submit a delete, their update will be lost along with the rest of the resource.

After a CustomResourceDefinition is deleted, invalidation of discovery cache may take up to 10 minutes. If you don't want to wait, you might want to run "kubectl api-resources" to refresh the discovery cache.

Usage

```
$ kubectl delete [{-f FILENAME} | {-k DIRECTORY} | TYPE [(NAME | -l label | --all)]]
```

Delete a pod based on the type and name in the JSON passed into stdin

cat pod.json | kubectl delete -f -

Delete pods and services with same names "baz" and "foo"

kubectl delete pod,service baz foo

Delete pods and services with label name=myLabel

kubectl delete pods,services -l name=myLabel

Delete a pod with minimal delay

kubectl delete pod foo --now

Force delete a pod on a dead node

kubectl delete pod foo --force

Delete all pods

kubectl delete pods --all

NEW QUESTION 95

Which of the following computing model doesn't require you to provision infrastructure?

- A. None of the above
- B. Bare Metal
- C. Compute Engine
- D. Virtual Machines
- E. Serverless

Answer: E

NEW QUESTION 99

In Kubernetes, what is considered the primary cluster data source?

- A. etcd (pronounce: esty-d)
- B. api server
- C. kubelet
- D. scheduler

Answer: A

Explanation:

<https://kubernetes.io/docs/concepts/overview/components/#etcd>

Graphical user interface, text, application, email Description automatically generated

etcd

Consistent and highly-available key value store used as Kubernetes' backing store for all cluster data.

If your Kubernetes cluster uses etcd as its backing store, make sure you have a [back up](#) plan for those data.

You can find in-depth information about etcd in the official [documentation](#).

NEW QUESTION 104

Which command is used to expose Kubernetes service

- A. kubectl expose
- B. kubectl create
- C. kubectl run

Answer: A

Explanation:

<https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#expose> Text Description automatically generated

Create a service for a replicated nginx, which serves on port 80 and connects to the containers on port 8000

```
kubectl expose rc nginx --port=80 --target-port=8000
```

NEW QUESTION 109

.....

Thank You for Trying Our Product

We offer two products:

1st - We have Practice Tests Software with Actual Exam Questions

2nd - Questions and Answers in PDF Format

KCNA Practice Exam Features:

- * KCNA Questions and Answers Updated Frequently
- * KCNA Practice Questions Verified by Expert Senior Certified Staff
- * KCNA Most Realistic Questions that Guarantee you a Pass on Your FirstTry
- * KCNA Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year

100% Actual & Verified — Instant Download, Please Click
[Order The KCNA Practice Test Here](#)