

Actual Linux Foundation CKAD Tests | CKAD Certification Dumps

Achieve success by using our corrected Linux Foundation CKAD exam questions 2024. We offer success guarantee with our updated CKAD dumps.

Linux Foundation CKAD Exam Questions [Rectified 2024] - Get Ready For The Exam

Are you taking the Certified Kubernetes Application Developer Exam and want to ensure perfect preparation for the CKAD Kubernetes Application Developer exam? CertsLink [Linux Foundation CKAD exam questions](#) preparation can help you get there with ease. CertsLink Linux Foundation CKAD exam questions is a comprehensive learning package that offers the CKAD Kubernetes Application Developer exam real questions and answers with key features so that you can prepare for the CKAD Certified Kubernetes Application Developer Exam smoothly.



Real Linux Foundation CKAD Exam Questions In The PDF Format

The Kubernetes Application Developer CKAD exam questions are available in pdf format, which makes it convenient for you to save the Linux Foundation CKAD pdf to any device such as desktop, mac, smartphone, laptop, and tablet. It also means that the Linux Foundation CKAD exam questions is easily accessible no matter where you are, so you can prepare for your CKAD Certified Kubernetes Application Developer Exam at any time anywhere.

P.S. Free 2025 Linux Foundation CKAD dumps are available on Google Drive shared by PDFTorrent:
<https://drive.google.com/open?id=1HGILm1DkiQziAeSDUAj47aSdAGYZ0jwh>

In order to meet the demand of all customers and protect your machines network security, our company can promise that our CKAD test training guide have adopted technological and other necessary measures to ensure the security of personal information they collect, and prevent information leaks, damage or loss. In addition, the CKAD exam dumps system from our company can help all customers ward off network intrusion and attacks prevent information leakage, protect user machines network security. If you choose our CKAD study questions as your study tool, we can promise that we will try our best to enhance the safety guarantees and keep your information from revealing, and your privacy will be protected well. You can rest assured to buy the CKAD exam dumps from our company.

So rest assured that with the PDFTorrent CKAD exam questions you will get everything that is necessary for CKAD exam preparation and success. Take a decision right now and just get registered in the Linux Foundation CKAD certification exam and start preparation with PDFTorrent CKAD Exam Questions. You do not need to get worried about it choose the right PDFTorrent Linux Foundation Certified Kubernetes Application Developer Exam exam questions formats and start this journey without wasting further time.

>> Actual Linux Foundation CKAD Tests <<

CKAD Certification Dumps - Valid CKAD Torrent

You can now get Linux Foundation CKAD exam certification our PDFTorrent have the full version of Linux Foundation CKAD

exam. You do not need to look around for the latest Linux Foundation CKAD training materials, because you have to find the best Linux Foundation CKAD Training Materials. Rest assured that our questions and answers, you will be completely ready for the Linux Foundation CKAD certification exam.

The CKAD Exam tests the candidate's ability to design, build, configure, and expose cloud-native applications for Kubernetes. CKAD exam is conducted in a hands-on, performance-based format, where candidates are required to solve real-world problems using Kubernetes. CKAD Exam Tests the candidate's proficiency in various tasks, such as deploying applications, configuring storage, networking, security, and troubleshooting.

Linux Foundation Certified Kubernetes Application Developer Exam Sample Questions (Q76-Q81):

NEW QUESTION # 76

You have a Deployment named 'wordpress-deployment' that runs 3 replicas of a Wordpress container with the image 'wordpress:latest'. You need to ensure that when a new image is pushed to the Docker Hub repository 'my-wordpress-repo/wordpress:latest', the Deployment automatically updates to use the new image. Additionally, you need to set up a rolling update strategy where only one pod is updated at a time. The maximum number of unavailable pods at any given time should be 1.

Answer:

Explanation:

See the solution below with Step by Step Explanation.

Explanation:

Solution (Step by Step) :

1. Update the Deployment YAML.

- Add 'imagePullPolicy: Always' to the container definition to ensure the deployment pulls the latest image from the Docker Hub repository even if a local copy exists.
- Set 'strategy-type: Rollingupdate' to enable a rolling update strategy.
- Configure 'strategy.rollingupdate.maxUnavailable: 1' to allow only one pod to be unavailable during the update process.
- Set 'strategy.rollingupdate.maxSurge: 0' to restrict the number of pods added during the update to zero.

NEW QUESTION # 77

Refer to Exhibit.



Context

You are tasked to create a secret and consume the secret in a pod using environment variables as follow:

Task

- * Create a secret named another-secret with a key/value pair; key1/value4
- * Start an nginx pod named nginx-secret using container image nginx, and add an environment variable exposing the value of the secret key key 1, using COOL_VARIABLE as the name for the environment variable inside the pod

Answer:

Explanation:

Solution:

```
student@node-1:~$ kubectl create secret generic some-secret --from-literal=key1=value4
secret/some-secret created
student@node-1:~$ kubectl get secret
NAME                                TYPE                                DATA  AGE
default-token-4kvr5                 kubernetes.io/service-account-token 3      2d11h
some-secret                         Opaque                             1      5s
student@node-1:~$ kubectl run nginx-secret --image=nginx --dry-run=client -o yaml > nginx_secret.yml
student@node-1:~$ vim nginx_secret.yml
```

```
apiVersion: v1
kind: Pod
metadata:
  creationTimestamp: null
  labels:
    run: nginx-secret
    name: nginx-secret
spec:
  containers:
  - image: nginx
    name: nginx-secret
    resources: {}
    dnsPolicy: ClusterFirst
    restartPolicy: Always
status: {}

"nginx_secret.yml" 15L, 253C
```



```
apiVersion: v1
kind: Pod
metadata:
  labels:
    run: nginx-secret
    name: nginx-secret
spec:
  containers:
  - image: nginx
    name: nginx-secret
    env:
    - name: COOL_VARIABLE
      valueFrom:
        secretKeyRef:
          name: some-secret
          key: key1
```



Readme Web Terminal

THE LINUX FOUNDATION

```
student@node-1:~$ kubectl get pods -n web
NAME      READY   STATUS    RESTARTS   AGE
cache     1/1     Running   0           9s
student@node-1:~$ kubectl create secret generic some-secret --from-literal=key1=value4
secret/some-secret created
student@node-1:~$ kubectl get secret
NAME                TYPE
default-token-4kvr5  kubernetes.io/service-account-token
some-secret          Opaque
student@node-1:~$ kubectl run nginx-secret --image=nginx --dry-run=client -o yaml > nginx_secret.yaml
student@node-1:~$ vim nginx_secret.yaml
student@node-1:~$ kubectl create -f nginx_secret.yaml
pod/nginx-secret created
student@node-1:~$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
liveness-http       1/1     Running   0           6h38m
nginx-101            1/1     Running   0           6h39m
nginx-secret         0/1     ContainerCreating   0           4s
poller               1/1     Running   0           6h39m
student@node-1:~$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
liveness-http       1/1     Running   0           6h38m
nginx-101            1/1     Running   0           6h39m
nginx-secret         1/1     Running   0           8s
poller               1/1     Running   0           6h39m
student@node-1:~$
```

NEW QUESTION # 78

You are managing a Kubernetes cluster with multiple teams working on different projects. Each team needs its own isolated environment within the cluster to deploy their applications and manage their resources without interfering with others. Describe how you would use Kubernetes namespaces to achieve this, and provide an example of how you might configure a namespace for a team working on a new e-commerce application.

Answer:

Explanation:

See the solution below with Step by Step Explanation.

Explanation:

Solution (Step by Step) :

1. Create Namespaces for Teams: use 'kubectl create namespace' command to create namespaces for each team. For example, 'kubectl create namespace ecom-team'.
2. Configure Resource Quotas: Set resource limits for each namespace using 'kubectl create -f' command. This prevents one team from consuming all the resources available on the cluster. Here's a sample resource quota file:

```
apiVersion: v1
kind: ResourceQuota
metadata:
  name: ecom-team-quota
  namespace: ecom-team
spec:
  requests:
    cpu: "2"
    memory: "2Gi"
  limits:
    cpu: "4"
    memory: "4Gi"
  pods: "10"
```

3. Apply Role-Based Access Control (RBAC): Use 'kubectl create -f' command to define role bindings for each team. This allows you to control the actions that each team can perform within their namespace. Here's a sample role binding file:

```

apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: ecom-team-binding
  namespace: ecom-team
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: ecom-team-role
subjects:
- kind: User
  name: ecom-user
  apiGroup: rbac.authorization.k8s.io

```

4. Create Resources within the Namespace: Deploy your applications and other resources within the dedicated namespace for the e-commerce team. For example, you can deploy a 'Deployment' With the following configuration:

```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: ecom-app-deployment
  namespace: ecom-team
spec:
  replicas: 3
  selector:
    matchLabels:
      app: ecom-app
  template:
    metadata:
      labels:
        app: ecom-app
    spec:
      containers:
      - name: ecom-app
        image: example/ecom-app:latest

```

5. Verify Namespace Configuration: Use 'kubectl get namespaces' to list all namespaces, and 'kubectl describe namespace' to view details of a specific namespace. 6. Manage Namespace Access: You can use tools like 'kubectl' or a graphical user interface (GUI) to manage the access rights and resources within each namespace. 7. Cleanup: When a team no longer needs a specific namespace, you can delete it using 'kubectl delete namespace'.

NEW QUESTION # 79

Context



Context

As a Kubernetes application developer you will often find yourself needing to update a running application.

Task

Please complete the following:

- * Update the app deployment in the kdpd00202 namespace with a maxSurge of 5% and a maxUnavailable of 2%
- * Perform a rolling update of the web1 deployment, changing the lfcncf/nginx image version to 1.13
- * Roll back the app deployment to the previous version

Answer:

Explanation:

Solution:

```

student@node-1:~$ kubectl edit deployment app -n kdpd00202
uid: 1dfa2527-5c61-46a9-8dd3-e24643d3ce14
spec:
  progressDeadlineSeconds: 600
  replicas: 10
  revisionHistoryLimit: 10
  selector:
    matchLabels:
      app: nginx
  strategy:
    rollingUpdate:
      maxSurge: 5%
      maxUnavailable: 2
    type: RollingUpdate
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: nginx
    spec:
      containers:
      - image: lfcncf/nginx:1.13
        imagePullPolicy: IfNotPresent
        name: nginx
        ports:
        - containerPort: 80
          protocol: TCP
:wg:

student@node-1:~$ kubectl edit deployment app -n kdpd00202
deployment.apps/app edited
student@node-1:~$ kubectl rollout status deployment app -n kdpd00202
Waiting for deployment "app" rollout to finish: 6 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 7 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 7 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 7 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 8 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 8 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 8 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 8 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 9 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 9 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 9 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "app" rollout to finish: 8 of 10 updated replicas are available...
Waiting for deployment "app" rollout to finish: 9 of 10 updated replicas are available...
deployment "app" successfully rolled out
student@node-1:~$ kubectl rollout undo deployment app -n kdpd00202
deployment.apps/app rolled back
student@node-1:~$ kubectl rollout status deployment app -n kdpd00202
Waiting for deployment "app" rollout to finish: 6 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 6 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 6 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 6 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 7 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 7 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 9 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 9 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 9 out of 10 new replicas have been updated...
Waiting for deployment "app" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "app" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "app" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "app" rollout to finish: 8 of 10 updated replicas are available...
Waiting for deployment "app" rollout to finish: 9 of 10 updated replicas are available...
deployment "app" successfully rolled out
student@node-1:~$

```

NEW QUESTION # 80

You have a Deployment named 'wordpress-deployment' that runs 3 replicas of a WordPress container. You need to implement a persistent volume claim (PVC) for each pod that stores the website data, and you want to ensure that the data persists even if the pod is deleted or restarted. The PVC should be created using a storage class named 'standard' with a capacity of 10Gi.

Answer:

Explanation:

See the solution below with Step by Step Explanation.

Explanation:

Solution (Step by Step) :

1). Create a Storage Class:

- Create a 'standard' storage class:

```
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: standard
provisioner: kubernetes.io/gce-pd
parameters:
  type: pd-standard
```

- Apply the YAML file: `bash kubectl apply -f standard-storage-class.yaml` 2. Create a Persistent Volume Claim: - Create a PVC named 'wordpress-pvc' with a request for 10Gi storage and using the 'standard' storage class:

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: wordpress-pvc
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 10Gi
  storageClassName: standard
```

- Apply the YAML file: `bash kubectl apply -f wordpress-pvc.yaml` 3. Update the Deployment - Update the 'wordpress-deployment' YAML file to mount the PVC to each pod:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: wordpress-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: wordpress
  template:
    metadata:
      labels:
        app: wordpress
    spec:
      containers:
        - name: wordpress
          image: wordpress:latest
          ports:
            - containerPort: 80
          volumeMounts:
            - name: wordpress-data
              mountPath: /var/www/html
      volumes:
        - name: wordpress-data
          persistentVolumeClaim:
            claimName: wordpress-pvc
```

- Apply the updated YAML file: `bash kubectl apply -f wordpress-deployment.yaml` 4. Verify the Deployment - Check the status of the deployment using `'kubectl get deployments wordpress-deployment'` to confirm the rollout and updated replica count. - Use

'kubect describe pods -l app=wordpress' to confirm that each pod is using the 'wordpress-pvc' and the website data is stored in the persistent volume. - You can now access the WordPress website through the service that is associated with the Deployment. 5. Test Data Persistence: - Delete or restan one of the pods in the deployment. - Observe that the website data remains intact because the PVC is persistent and the data is stored in the underlying volume.,

NEW QUESTION # 81

.....

For the buyers who want to buy CKAD Study Materials, some may have the concern of the security of website. We can tell you that if you buy the CKAD exam dumps of us, and we ensure the safety of yours. We have the specialized technicians to maintain the website at times, therefore the safety of website is guaranteed, and if you indeed encounter some problem, just contact with our service stuff, they will help you to solve the problem.

CKAD Certification Dumps: <https://www.pdf torrent.com/CKAD-exam-prep-dumps.html>

- Pass Guaranteed Quiz Linux Foundation - CKAD Accurate Actual Tests ☐ The page for free download of 《 CKAD 》 on { www.exams4collection.com } will open immediately ☐ Exam CKAD Review
- Pass Guaranteed Linux Foundation CKAD - Linux Foundation Certified Kubernetes Application Developer Exam Fantastic Actual Tests ➔ Search for { CKAD } on 《 www.pdfvce.com 》 immediately to obtain a free download ☐ CKAD Reliable Braindumps Files
- Pass Guaranteed Quiz Linux Foundation - CKAD Accurate Actual Tests ☐ Copy URL ➤ www.prep4pass.com ◀ open and search for ➤ CKAD ☐ to download for free ☐ Authentic CKAD Exam Hub
- New CKAD Dumps Ebook ☐ CKAD Well Prep ☐ Reliable CKAD Braindumps Free ☐ Copy URL 【 www.pdfvce.com 】 open and search for “CKAD” to download for free ☐ Exam CKAD Review
- New CKAD Dumps Ebook ☐ CKAD Exam Tests ☐ CKAD Exam Topic ☐ Search on ✓ www.itcerttest.com ☐ ✓ ☐ for ✓ CKAD ☐ ✓ ☐ to obtain exam materials for free download ☐ New CKAD Dumps Ebook
- CKAD Guaranteed Passing ☐ New CKAD Dumps Ebook ☐ Valid CKAD Exam Topics ☐ Copy URL { www.pdfvce.com } open and search for ✓ CKAD ☐ ✓ ☐ to download for free ☐ CKAD New Practice Materials
- CKAD Latest Material ☐ CKAD Free Vce Dumps ☐ New CKAD Dumps Ebook ☐ Search on ➤ www.actual4labs.com ☐ for 《 CKAD 》 to obtain exam materials for free download ☐ CKAD Latest Material
- Mock CKAD Exam ↘ Valid CKAD Vce ☐ CKAD Exam Topic ☐ Open website (www.pdfvce.com) and search for ➤ CKAD ◀ for free download ☐ Visual CKAD Cert Test
- Authentic CKAD Exam Hub ☐ Reliable CKAD Braindumps Free ☐ Reliable CKAD Braindumps Free ☎ Search on ☐ www.free4dump.com ☐ for ➡ CKAD ☐ to obtain exam materials for free download ☐ CKAD New Study Questions
- Pass Guaranteed Quiz Linux Foundation - CKAD Accurate Actual Tests ☐ Simply search for ➡ CKAD ☐ ☐ ☐ for free download on ➡ www.pdfvce.com ☐ ☐ CKAD Guaranteed Passing
- CKAD Vce File ☐ New CKAD Dumps Ebook ☐ Exam CKAD Review ☐ Copy URL “ www.real4dumps.com ” open and search for 「 CKAD 」 to download for free ☐ CKAD New Study Questions
- kuhenan.com, worldsuccesses.com, pct.edu.pk, bijie.cnrxw.cn, raywalk191.bloguetechno.com, www.stes.tyc.edu.tw, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, Disposable vapes

BTW, DOWNLOAD part of PDFTorrent CKAD dumps from Cloud Storage: <https://drive.google.com/open?id=1HGILm1DkiQziAeSDUAj47aSdAGYZ0jwh>