CKAD日本語復習赤本、CKAD日本語

2026年Fast2testの最新CKAD PDFダンプおよびCKAD試験エンジンの無料共有：https://drive.google.com/open?id=1YHGdZkd7WuWAlMl6oRk5eRZxcJ0_vo_K

Linux FoundationのCKAD試験に準備するために、たくさんの本と塾なしで、我々Fast2testのソフトを使用すればリラクスで目標を達成できます。弊社の商品はあなたの圧力を減少できます。それだけでなく、お金を無駄にする心配なあなたに保障を提供いたします。あなたは弊社の商品を利用して、一回でLinux FoundationのCKAD試験に合格できなかったら、弊社は全額で返金することを承諾いたします。

Linux Foundation Certified Kubernetes Application Developer（CKAD）試験は、Kubernetesで作業する開発者の知識とスキルをテストする人気のある認定プログラムです。この試験は、Kubernetesクラスターでクラウドネイティブアプリケーションの設計、構築、構成、展開における個人の習熟度を評価するように設計されています。 CKAD試験に合格した候補者は、Linux Foundationによって認定Kubernetesアプリケーション開発者として公式に認められています。

>> CKAD日本語復習赤本 <<

CKAD日本語 & CKAD日本語受験教科書

お客様が問題を解決できるように、当社は常に問題を最優先し、価値あるサービスを提供することを強く求めています。 CKAD質問トレントは、短時間で試験に合格し、認定資格を取得するのに役立つと確信しています。 CKADガイドの質問を理解するのが待ち遠しいかもしれません。他の教材と比較した場合、当社の製品の品質がより高いことをお約束します。現時点では、CKADガイドトレントのデモを無料でダウンロードできます。CKAD試験問題をご存知の場合は、ぜひお試しください。

CKAD認定試験は、Kubernetesアプリケーション開発の専門知識を持っていることを証明するために有用な資格です。これは、開発者がキャリアを進め、収入を増やすのに役立ちます。さらに、組織がKubernetesのフルポテンシャルをアプリケーションに活用することができる熟練したKubernetes開発者を特定し、採用するための方法でもあります。

Linux Foundation Certified Kubernetes Application Developer Exam 認定 CKAD 試験問題 (Q190-Q195):

質問 # 190
You are deploying a microservice application that requires secure access to an external database. The database credentials are stored as environment variables Within the application container. You want to create a Kubernetes secret that securely stores these credentials and can be mounted as a file in the container.

正解：

解説：
See the solution below with Step by Step Explanation.
Explanation:
Solution (Step by Step) :
1. Create a Kubernetes Secret:
- Create a YAML file, for example, 'database-secret.yamr , with the following content:
- Replace ". ". ". and with the actual values, Base64 encoded. You can use the 'base64' command to encode the values: bash echo "your _ username" | base64 echo 'Your_password" | base64 echo "your _ host" | base64 echo "your _ port" | base64 2. Apply the Secret: - Apply the secret to your Kubernetes cluster: bash kubectl apply -f database-secret.yaml 3. Modify the Deployment: - Modify your Deployment YAML file to mount the secret as a file:
4. Apply the Updated Deployment: - Apply the updated Deployment YAML file using: bash kubectl apply -f my-microservice-deployment.yaml 5. Accessing Credentials: - The application container can now access the environment variables from the secret using 'process-env-DATABASE USER , 'process.env.DATABASE_PASSWORD', etc. Additionally, the secret data is mounted as a file at '/var/secrets/database'.

質問 # 191
You are developing a service that uses a custom configuration file called 'service.properties'. You want to use ConfigMaps to store and manage this file in a secure and efficient manner. The 'service-properties' file contains sensitive information such as database credentials and API keys.
How would you create a ConfigMap that securely stores the 'service-properties' file, ensuring that the file is accessible only to the service's container?

正解：

解説：
See the solution below with Step by Step Explanation.
Explanation:
Solution (Step by Step) :
1. Create a Secret for Sensitive Data:
- Create a Secret
- Encode the 'service-properties' file: bash echo "your-database-username=your-database-username" > service-properties echo "your-database-password=your-database-password" >> service-properties echo 'Your-api-key=your-api-key" >> service.properties base64 -w 0 service.properties - Replace with the output from the base64 command. 2. Create the ConfigMap for the File:
3. Apply the Secret and ConfigMap: bash kubectl apply -f service-secrets-yaml kubectl apply -f service-config.yaml 4. Update the Deployment to use the ConfigMap and Secret
5. Apply the updated Deployment: bash kubectl apply -f my-service-deployment-yaml 6. Access the File in the Container. - Mount the ConfigMap and Secret: - The ConfigMap mounts the 'service.properties' file as a placeholder. - The Secret mounts the actual 'service.properties' file securely. - Access the File: - The container should access the 'service.properties' file from '/var/secrets/service/service.properties' This approach uses a Secret to store sensitive data and a ConfigMap to mount the file securely within the container. The container will have access to the 'service-properties' file, but the actual data is stored in the Secret, ensuring its confidentiality'.

質問 # 192
Context
Context
You have been tasked with scaling an existing deployment for availability, and creating a service to expose the deployment within your infrastructure.
Task
Start with the deployment named kdsn00101-deployment which has already been deployed to the namespace kdsn00101 . Edit it

to:
* Add the func=webFrontEnd key/value label to the pod template metadata to identify the pod for the service definition
* Have 4 replicas
Next, create ana deploy in namespace kdsn00l01 a service that accomplishes the following:
* Exposes the service on TCP port 8080
* is mapped to me pods defined by the specification of kdsn00l01-deployment
* Is of type NodePort
* Has a name of cherry

正解：

解説：
Solution:

質問 # 193
You have a Deployment named 'redis-deployment that runs 3 replicas of a Redis container. You need to implement a rolling update strategy that allows tor a maximum ot one pod to be unavailable at any given time during tne update process, With the new pod becoming available before the old pod is terminated. Additionally, you want to ensure that the update process is triggered automatically whenever a new image is pushed to the Docker Hub repository 'redislabs/redis:latest.

正解：

解説：
See the solution below with Step by Step Explanation.
Explanation:
Solution (Step by Step) :
1. Update the Deployment YAML:
- Update the 'replicas to 2_
- Define 'maxunavailable: 1 ' and 'maxSurge: 1 ' in the 'strategy.rollingupdate' section to control the rolling update process.
- Configure a 'strategy-type' to 'Rollingupdate' to trigger a rolling update when the deployment is updated.
- Add a 'spec-template.spec.imagePullPolicy: Always' to ensure that the new image is pulled even if it exists in the pod's local cache.
2. Create the Deployment: - Apply the updated YAML file using 'kubectl apply -f redis-deployment.yaml' 3. Verify the Deployment - Check the status of the deployment using 'kubectl get deployments redis-deployment' to confirm the rollout and updated replica count. 4. Trigger the Automatic Update. - Push a new image to the Docker Hub repository 5. Monitor the Deployment: - Use 'kubectl get pods -l app=rediS to monitor the pod updates during the rolling update process. You will observe that one new pod with the updated image is created, and then one old pod is terminated- This ensures that there is no downtime during the update process. 6. Check for Successful Update: - Once the deployment is complete, use 'kubectl describe deployment redis-deployment' to see that the 'updatedReplicaS field matches the 'replicas' field, indicating a successful update.

質問 # 194
You have a Deployment named 'my-app-deployment' running an application that requires a specific version of a database. This version is available in a private Docker registry with access credentials stored in a Secret. How would you configure the Deployment to pull the database image from the private registry using the Secret's credentials?

正解：

解説：
See the solution below with Step by Step Explanation.
Explanation:
Solution (Step by Step) :
1. Create a Secret:
- Create a secret containing the username and password required to access the private registry.
- Replace 'your-registry-username' and 'your-registry-password' with your actual credentials.
2. Update the Deployment - Modify the Deployment configuration to include the 'imagePullSecrets' field. - Add the name oftne secret you created in the previous step. - Replace 'your-private-registry-domain/your-database-image:your-version' with the actual image name and version.
3. Apply the Changes: - Apply the updated Deployment configuration using 'kubectl apply -f my-app-deployment.yamr. 4. Verify

the Pull: - Check the logs of the Pods in the Deployment. You should see messages indicating that the database image is pulled from the private registry using the provided credentials.

質問 # 195

......

CKAD日本語: https://jp.fast2test.com/CKAD-premium-file.html

無料でクラウドストレージから最新のFast2test CKAD PDFダンプをダウンロードする：https://drive.google.com/open?id=1YHGdZkd7WuWAlMl6oRk5eRZxcJ0_vo_K