Exam Questions CKAD

Certified Kubernetes Application Developer (CKAD) Program

https://www.2passeasy.com/dumps/CKAD/

NEW QUESTION 1
Exhibit:

> You must switch to the correct cluster/configuration context. Failure to do so may result in a zero score.
>
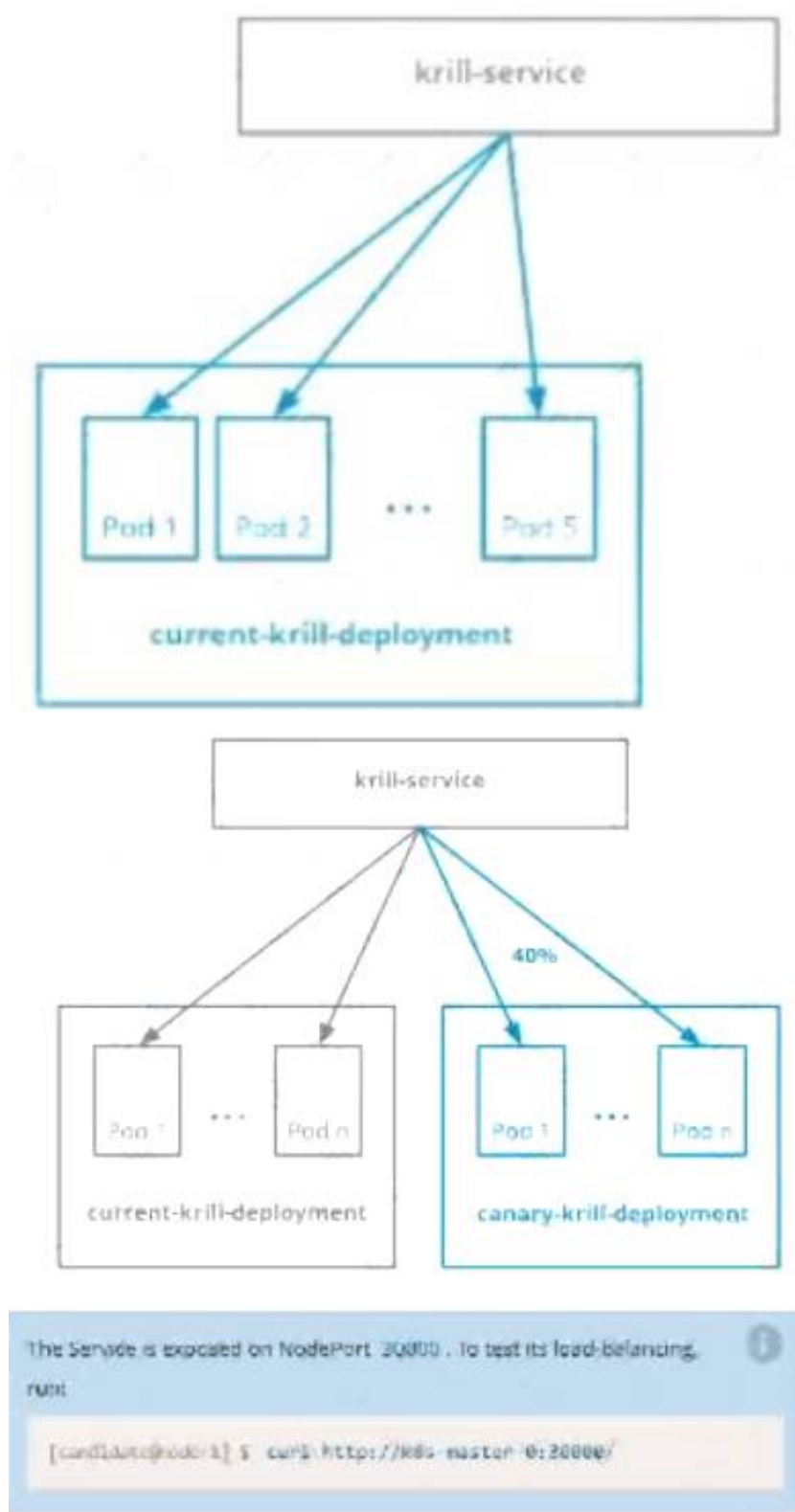> [candidate@node-1] $ kubectl config use-c ontext sk8s
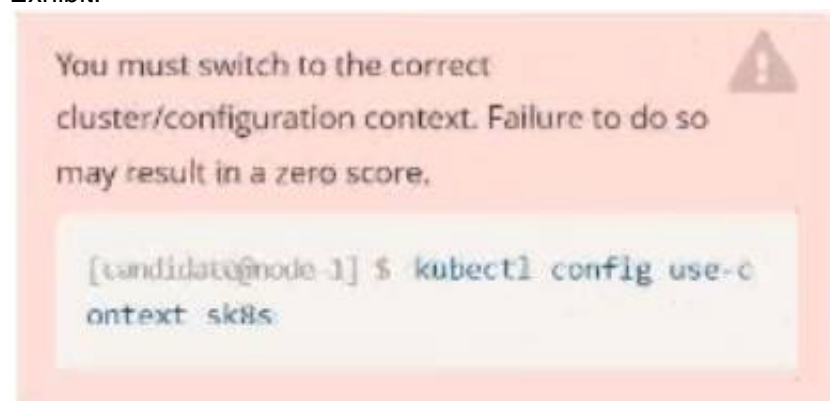
Context
You are asked to prepare a Canary deployment for testing a new application release.
Task:
A Service named krill-Service in the goshark namespace points to 5 pod created by the Deployment named current-krill-deployment

> The Service is exposed on NodePort 30800 . To test its load-balancing, run:
>
> [candidate@node-1] $ curl http://k8s-master-0:30800/

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution:

```
candidate@node-1:~/humane-storks$ kubectl  scale  deploy canary-krill-deployment --replicas 4 -n goshawk
deployment.apps/canary-krill-deployment scaled
candidate@node-1:~/humane-storks$ kubectl get deploy -n goshawk
NAME                       READY   UP-TO-DATE   AVAILABLE   AGE
canary-krill-deployment    4/4     4            4           46s
current-krill-deployment   5/5     5            5           7h22m
candidate@node-1:~/humane-storks$ wget https://k8s.io/examples/
```

Text Description automatically generated

```
File Edit View Terminal Tabs Help
candidate@node-1:~/humane-stork$ wget https://k8s.io/examples/admin/resource/quota-pod.yaml
--2022-09-24 11:43:51--  https://k8s.io/examples/admin/resource/quota-pod.yaml
Resolving k8s.io (k8s.io)... 34.107.204.206, 2600:1901:0:26f3::
Connecting to k8s.io (k8s.io)|34.107.204.206|:443... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://kubernetes.io/examples/admin/resource/quota-pod.yaml [following]
--2022-09-24 11:43:52--  https://kubernetes.io/examples/admin/resource/quota-pod.yaml
Resolving kubernetes.io (kubernetes.io)... 147.75.40.148
Connecting to kubernetes.io (kubernetes.io)|147.75.40.148|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 90 [application/x-yaml]
Saving to: 'quota-pod.yaml'

quota-pod.yaml              100%[===================================================>]      90  --.-KB/s    in 0s

2022-09-24 11:43:52 (15.0 MB/s) - 'quota-pod.yaml' saved [90/90]

candidate@node-1:~/humane-stork$ vim quota-pod.yaml
```

```
File Edit View Terminal Tabs Help
2022-09-24 11:43:52 (15.0 MB/s) - 'quota-pod.yaml' saved [90/90]

candidate@node-1:~/humane-stork$ vim quota-pod.yaml
candidate@node-1:~/humane-stork$ kubectl  create  -f quota-pod.yaml
resourcequota/pod-demo created
candidate@node-1:~/humane-stork$ kubectl get quota -n go
No resources found in go namespace.
candidate@node-1:~/humane-stork$ kubectl get quota -n goshawk
NAME        AGE    REQUEST       LIMIT
pod-demo    19s    pods: 9/10
candidate@node-1:~/humane-stork$ curl http://k8s-master-0:30000/
current-krill-deployment-fb7c7995c-kvtjr
app.kubernetes.io/name="current"
app.kubernetes.io/part-of="krill"
pod-template-hash="fb7c7995c"candidate@node-1:~/humane-stork$ curl http://k8s-master-0:30000/
current-krill-deployment-fb7c7995c-4whfm
app.kubernetes.io/name="current"
app.kubernetes.io/part-of="krill"
pod-template-hash="fb7c7995c"candidate@node-1:~/humane-stork$ curl http://k8s-master-0:30000/
canary-krill-deployment-5f78fd4786-dfk7l
app.kubernetes.io/name="canary"
app.kubernetes.io/part-of="krill"
pod-template-hash="5f78fd4786"candidate@node-1:~/humane-stork$ curl http://k8s-master-0:30000/
canary-krill-deployment-5f78fd4786-z5zrt
app.kubernetes.io/name="canary"
app.kubernetes.io/part-of="krill"
pod-template-hash="5f78fd4786"candidate@node-1:~/humane-stork$ curl http://k8s-master-0:30000/
canary-krill-deployment-5f78fd4786-2774b
app.kubernetes.io/name="canary"
app.kubernetes.io/part-of="krill"
pod-template-hash="5f78fd4786"candidate@node-1:~/humane-stork$
```

NEW QUESTION 2
Exhibit:

You must switch to the correct
cluster/configuration context. Failure to do so
may result in a zero score.

```
[candidate@node-1] $ kubectl config use-c
ontext sk8s
```

Task:
Create a Deployment named expose in the existing ckad00014 namespace running 6 replicas of a Pod. Specify a single container using the ifccncf/nginx: 1.13.7 image
Add an environment variable named NGINX_PORT with the value 8001 to the container then expose port 8001

A. Mastered
B. Not Mastered

Answer: A

Explanation:

Solution:

```
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl  create  deploy expose -n  ckad00014  --image lfccncf/nginx:1.13.7 --dry-run=client -o yaml> d
ep.yaml
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$

candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
```

Text Description automatically generated

```
File Edit View Terminal Tabs Help
apiVersion: apps/v1
kind: Deployment
metadata:
  creationTimestamp: null
  labels:
    app: expose
  name: expose
  namespace: ckad00014
spec:
  replicas: 6
  selector:
    matchLabels:
      app: expose
  strategy: {}
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: expose
    spec:
      containers:
      - image: lfccncf/nginx:1.13.7
        name: nginx
        ports:
          - containerPort: 8001
        env:
          - name: NGINX_PORT
            value: "8001"


:wq
```

Text Description automatically generated

```
File Edit View Terminal Tabs Help
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl  create  deploy expose -n  ckad00014  --image lfccncf/nginx:1.13.7 --dry-run=client -o yaml> d
ep.yaml
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$

candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$ vim dep.yaml
candidate@node-1:~$ kubectl  create  -f dep.yaml
deployment.apps/expose created
candidate@node-1:~$ kubectl get pods -n ckad00014
NAME                         READY   STATUS            RESTARTS   AGE
expose-85dd99d4d9-25675      0/1     ContainerCreating  0          6s
expose-85dd99d4d9-4fhcc      0/1     ContainerCreating  0          6s
expose-85dd99d4d9-fld7j      0/1     ContainerCreating  0          6s
expose-85dd99d4d9-tt6rm      0/1     ContainerCreating  0          6s
expose-85dd99d4d9-vjd8b      0/1     ContainerCreating  0          6s
expose-85dd99d4d9-vtzpq      0/1     ContainerCreating  0          6s
candidate@node-1:~$ kubectl  get deploy -n ckad00014
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
expose    6/6     6            6           15s
candidate@node-1:~$
```

NEW QUESTION 3
Exhibit:

Set configuration context:

```
[student@node-1] $ | kubectl config
use-context  k8s
```

Context
A web application requires a specific version of redis to be used as a cache. Task
Create a pod with the following characteristics, and leave it running when complete:
• The pod must run in the web namespace. The namespace has already been created
• The name of the pod should be cache
• Use the lfccncf/redis image with the 3.2 tag
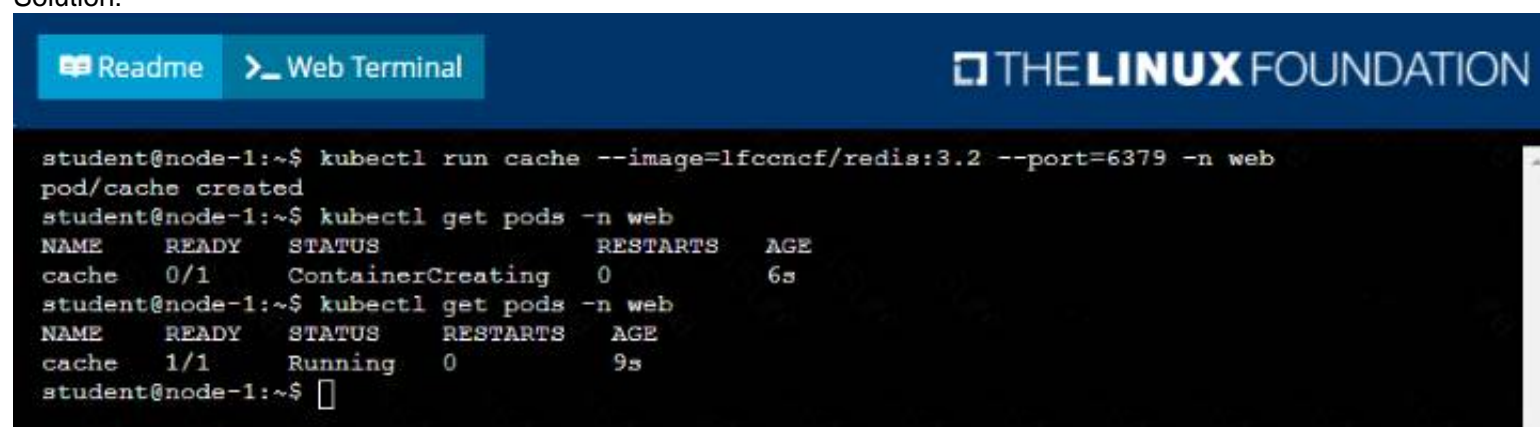• Expose port 6379
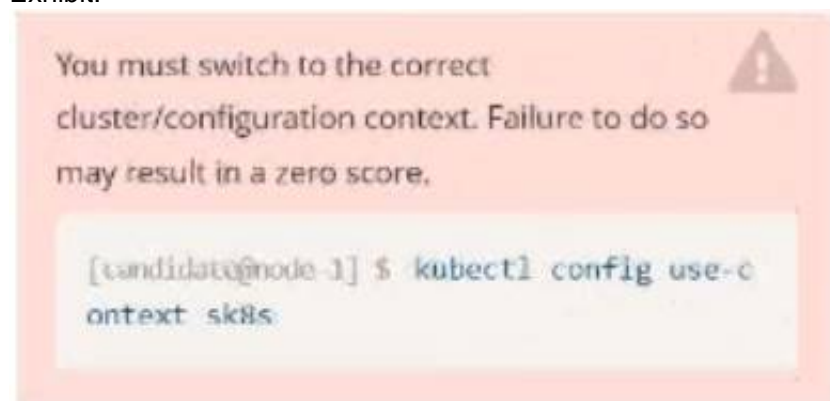
A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution:

Readme > **Web Terminal**   **THE LINUX FOUNDATION**

```
student@node-1:~$ kubectl run cache --image=lfccncf/redis:3.2 --port=6379 -n web
pod/cache created
student@node-1:~$ kubectl get pods -n web
NAME    READY    STATUS             RESTARTS    AGE
cache    0/1     ContainerCreating    0           6s
student@node-1:~$ kubectl get pods -n web
NAME    READY    STATUS    RESTARTS    AGE
cache    1/1     Running    0           9s
student@node-1:~$ ▯
```

NEW QUESTION 4
Exhibit:

You must switch to the correct
cluster/configuration context. Failure to do so
may result in a zero score.

```
[candidate@node-1] $ kubectl config use-c
ontext sk8s
```

Task:
1- Update the Propertunel scaling configuration of the Deployment web1 in the ckad00015 namespace setting maxSurge to 2 and maxUnavailable to 59
2- Update the web1 Deployment to use version tag 1.13.7 for the lfconf/nginx container image. 3- Perform a rollback of the web1 Deployment to its previous version
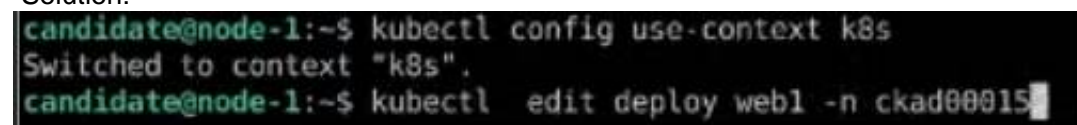
A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution:

```
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl  edit deploy web1 -n ckad00015▮
```

Text Description automatically generated

```
File Edit View Terminal Tabs Help
      app: nginx
  strategy:
    rollingUpdate:
      maxSurge: 2%
      maxUnavailable: 5%
    type: RollingUpdate
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: nginx
    spec:
      containers:
      - image: lfccncf/nginx:1.13.7
        imagePullPolicy: IfNotPresent
        name: nginx
        ports:
        - containerPort: 80
          protocol: TCP
        resources: {}
        terminationMessagePath: /dev/termination-log
        terminationMessagePolicy: File
      dnsPolicy: ClusterFirst
      restartPolicy: Always
      schedulerName: default-scheduler
      securityContext: {}
      terminationGracePeriodSeconds: 30
status:
  availableReplicas: 2
  conditions:
  - lastTransitionTime: "2022-09-24T04:26:41Z"
```

```
File Edit View Terminal Tabs Help
Switched to context "k8s".
candidate@node-1:~$ kubectl  create  secret  generic  app-secret -n default  --from-literal=key3=value1
secret/app-secret created
candidate@node-1:~$ kubectl get secrets
NAME          TYPE       DATA   AGE
app-secret    Opaque     1      4s
candidate@node-1:~$ kubectl  run nginx-secret -n default  --image=nginx:stable --dry-run=client -o yaml> sec.yaml
candidate@node-1:~$ vim sec.yaml
candidate@node-1:~$ kubectl  create  -f sec.yaml
pod/nginx-secret created
candidate@node-1:~$ kubectl  get pods
NAME            READY    STATUS     RESTARTS   AGE
nginx-secret    1/1      Running    0          7s
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl  edit deploy web1 -n ckad00015
deployment.apps/web1 edited
candidate@node-1:~$ kubectl  rollout  status  deploy web1 -n ckad00015
deployment "web1" successfully rolled out
candidate@node-1:~$ kubectl  rollout undo  deploy web1 -n ckad00015
deployment.apps/web1 rolled back
candidate@node-1:~$ kubectl  rollout  history  deploy web1 -n ckad00015
deployment.apps/web1
REVISION  CHANGE-CAUSE
2         <none>
3         <none>

candidate@node-1:~$ kubectl get rs -n ckad00015
NAME              DESIRED   CURRENT   READY   AGE
web1-56f98bcb79   0         0         0       63s
web1-85775b6b79   2         2         2       6h53m
candidate@node-1:~$
```

NEW QUESTION 5
Exhibit:

Set configuration context:

```
[student@node-1] $ │ kubectl config
use-context  k8s
```

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

A. Mastered
B. Not Mastered

Answer: A

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

```

Readme **Web Terminal**   **THE LINUX FOUNDATION**

```
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                            1,1           All
```

Readme **Web Terminal**   **THE LINUX FOUNDATION**

```
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
~
~
~
~
~
~
~
~
~
~
-- INSERT --                                           16,20         All
```

```
📖 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                               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
student@node-1:~$ kubectl create -f nginx_secret.yml
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:~$
```
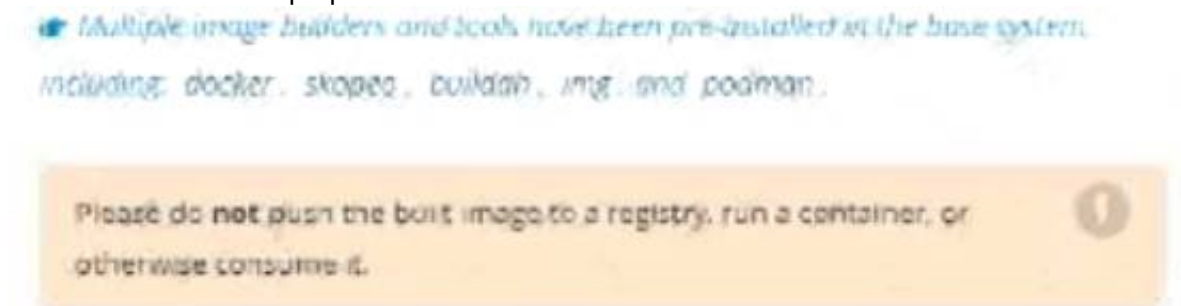
NEW QUESTION 6
Exhibit:

No configuration context change is required for this task.

Task:
A Dockerfile has been prepared at -/human-stork/build/Dockerfile

☛ Multiple image builders and tools have been pre-installed in the base system including docker, skopeo, buildah, img and podman.

Please do **not** push the built image to a registry, run a container, or otherwise consume it.

A. Mastered
B. Not Mastered

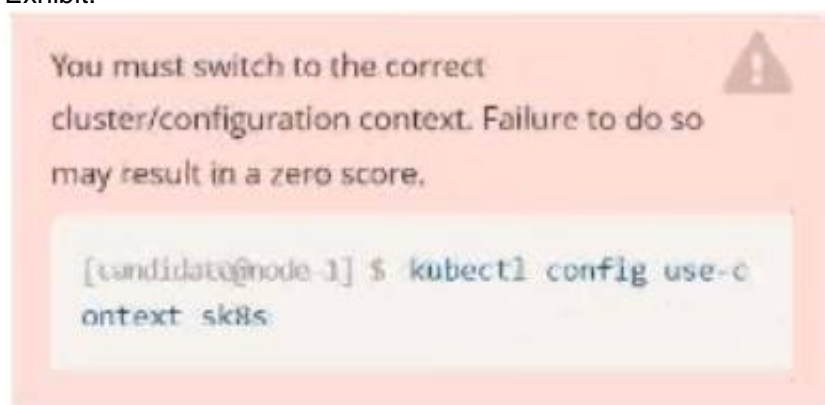**Answer:** A

Explanation:
 Solution:

```
candidate@node-1:~$ cd humane-stork/build/
candidate@node-1:~/humane-stork/build$ ls -l
total 16
-rw-r--r-- 1 candidate candidate 201 Sep 24 04:21 Dockerfile
-rw-r--r-- 1 candidate candidate 644 Sep 24 04:21 text1.html
-rw-r--r-- 1 candidate candidate 813 Sep 24 04:21 text2.html
-rw-r--r-- 1 candidate candidate 383 Sep 24 04:21 text3.html
candidate@node-1:~/humane-stork/build$ sudo docker build -t macaque:3.0 .
Sending build context to Docker daemon  6.144kB
Step 1/5 : FROM docker.io/lfccncf/nginx:mainline
 ---> ea335eea17ab
Step 2/5 : ADD text1.html /usr/share/nginx/html/
 ---> 8967ee9ee5d0
Step 3/5 : ADD text2.html /usr/share/nginx/html/
 ---> cb0554422f26
Step 4/5 : ADD text3.html /usr/share/nginx/html/
 ---> 62e879ab821e
Step 5/5 : COPY text2.html /usr/share/nginx/html/index.html
 ---> 331c8a94372c
Successfully built 331c8a94372c
Successfully tagged macaque:3.0
candidate@node-1:~/humane-stork/build$ sudo docker save macaque:3.0 > ~/humane-stork/macaque-3.0.tar
candidate@node-1:~/humane-stork/build$ cd ..
candidate@node-1:~/humane-stork$ ls -l
total 142532
drwxr-xr-x 2 candidate candidate      4096 Sep 24 04:21 build
-rw-rw-r-- 1 candidate candidate 145948672 Sep 24 11:39 macaque-3.0.tar
candidate@node-1:~/humane-stork$
```
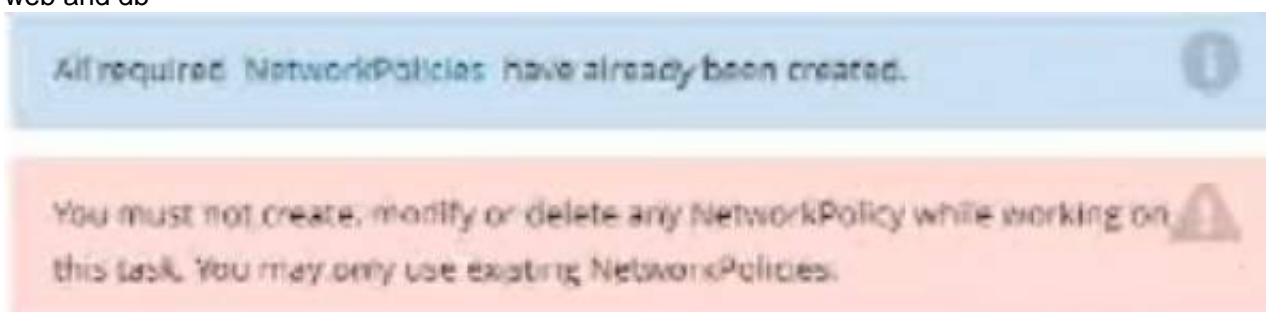
NEW QUESTION 7
Exhibit:

You must switch to the correct cluster/configuration context. Failure to do so may result in a zero score.

[candidate@node-1] $ kubectl config use-c ontext sk8s

Task:
Update the Pod ckad00018-newpod in the ckad00018 namespace to use a NetworkPolicy allowing the Pod to send and receive traffic only to and from the pods web and db

All required NetworkPolicies have already been created.

You must not create, modify or delete any NetworkPolicy while working on this task. You may only use existing NetworkPolicies.

A. Mastered
B. Not Mastered

Answer: A

Explanation:

Solution:

```
candidate@node-1:~$ kubectl config use-context nk8s
Switched to context "nk8s".
candidate@node-1:~$ kubectl  describe  netpol -n ckad00018
```

```
File Edit View Terminal Tabs Help
Name:          all-access
Namespace:     ckad00018
Created on:    2022-09-24 04:27:37 +0000 UTC
Labels:        <none>
Annotations:   <none>
Spec:
  PodSelector:     all-access=true
  Allowing ingress traffic:
    To Port: <any> (traffic allowed to all ports)
    From: <any> (traffic not restricted by source)
  Allowing egress traffic:
    To Port: <any> (traffic allowed to all ports)
    To: <any> (traffic not restricted by destination)
  Policy Types: Ingress, Egress


Name:          default-deny
Namespace:     ckad00018
Created on:    2022-09-24 04:27:37 +0000 UTC
Labels:        <none>
Annotations:   <none>
Spec:
  PodSelector:     <none> (Allowing the specific traffic to all pods in this namespace)
  Allowing ingress traffic:
    <none> (Selected pods are isolated for ingress connectivity)
  Not affecting egress traffic
  Policy Types: Ingress
candidate@node-1:~$ kubectl  label  pod ckad00018-newpod -n ckad00018   web-access=true
pod/ckad00018-newpod labeled
candidate@node-1:~$ kubectl  label  pod ckad00018-newpod -n ckad00018   db-access=true
pod/ckad00018-newpod labeled
candidate@node-1:~$
```

NEW QUESTION 8
Exhibit:

Set configuration context:

```
[student@node-1] $ | kubectl config
use-context k8s
```

Context
A pod is running on the cluster but it is not responding. Task
The desired behavior is to have Kubemetes restart the pod when an endpoint returns an HTTP 500 on the
/healthz endpoint. The service, probe-pod, should never send traffic to the pod while it is failing. Please complete the following:
• The application has an endpoint, /started, that will indicate if it can accept traffic by returning an HTTP 200. If the endpoint returns an HTTP 500, the application has not yet finished initialization.
• The application has another endpoint /healthz that will indicate if the application is still working as expected by returning an HTTP 200. If the endpoint returns an HTTP 500 the application is no longer responsive.
• Configure the probe-pod pod provided to use these endpoints
• The probes should use port 8080

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution:
apiVersion: v1 kind: Pod metadata: labels:
test: liveness
name: liveness-exec
spec: containers:
- name: liveness
image: k8s.gcr.io/busybox
args:
- /bin/sh
- -c
- touch /tmp/healthy; sleep 30; rm -rf /tmp/healthy; sleep 600
livenessProbe: exec: command:
- cat
- /tmp/healthy
initialDelaySeconds: 5
periodSeconds: 5
In the configuration file, you can see that the Pod has a single Container. The periodSeconds field specifies that the kubelet should perform a liveness probe every 5 seconds. The initialDelaySeconds field tells the kubelet that it should wait 5 seconds before performing the first probe. To perform a probe, the kubelet executes the command cat /tmp/healthy in the target container. If the command succeeds, it returns 0, and the kubelet considers the container to be alive and healthy. If the command returns a non-zero value, the kubelet kills the container and restarts it.
When the container starts, it executes this command:
/bin/sh -c "touch /tmp/healthy; sleep 30; rm -rf /tmp/healthy; sleep 600"
For the first 30 seconds of the container's life, there is a /tmp/healthy file. So during the first 30 seconds, the command cat /tmp/healthy returns a success code. After 30 seconds, cat /tmp/healthy returns a failure co
Create the Pod:
kubectl apply -f https://k8s.io/examples/pods/probe/exec-liveness.yaml Within 30 seconds, view the Pod events:
kubectl describe pod liveness-exec
The output indicates that no liveness probes have failed yet:
FirstSeen LastSeen Count From SubobjectPath Type Reason Message
--------- -------- ----- ---- ------------- -------- ------ ------
24s 24s 1 {default-scheduler } Normal Scheduled Successfully assigned liveness-exec to worker0
23s 23s 1 {kubelet worker0} spec.containers{liveness} Normal Pulling pulling image "k8s.gcr.io/busybox" 23s 23s 1 {kubelet worker0} spec.containers{liveness}
Normal Pulled Successfully pulled image
"k8s.gcr.io/busybox"
23s 23s 1 {kubelet worker0} spec.containers{liveness} Normal Created Created container with docker id 86849c15382e; Security:[seccomp=unconfined]
23s 23s 1 {kubelet worker0} spec.containers{liveness} Normal Started Started container with docker id 86849c15382e
After 35 seconds, view the Pod events again: kubectl describe pod liveness-exec
At the bottom of the output, there are messages indicating that the liveness probes have failed, and the containers have been killed and recreated.
FirstSeen LastSeen Count From SubobjectPath Type Reason Message
--------- -------- ----- ---- ------------- -------- ------ ------
37s 37s 1 {default-scheduler } Normal Scheduled Successfully assigned liveness-exec to worker0
36s 36s 1 {kubelet worker0} spec.containers{liveness} Normal Pulling pulling image "k8s.gcr.io/busybox" 36s 36s 1 {kubelet worker0} spec.containers{liveness}
Normal Pulled Successfully pulled image
"k8s.gcr.io/busybox"
36s 36s 1 {kubelet worker0} spec.containers{liveness} Normal Created Created container with docker id 86849c15382e; Security:[seccomp=unconfined]
36s 36s 1 {kubelet worker0} spec.containers{liveness} Normal Started Started container with docker id 86849c15382e
2s 2s 1 {kubelet worker0} spec.containers{liveness} Warning Unhealthy Liveness probe failed: cat: can't open '/tmp/healthy': No such file or directory
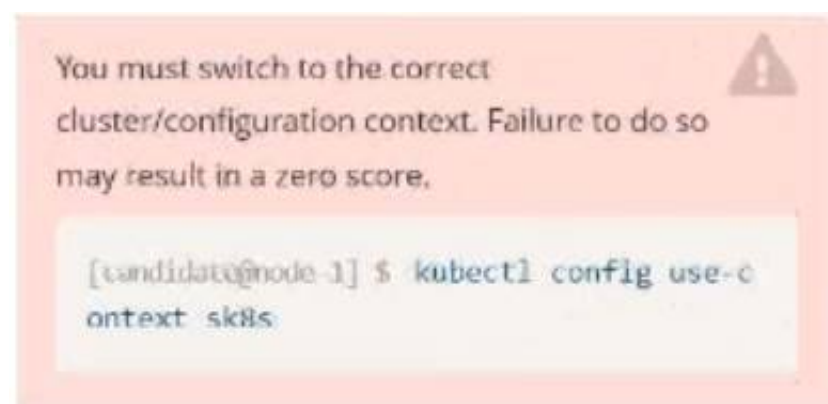Wait another 30 seconds, and verify that the container has been restarted: kubectl get pod liveness-exec
The output shows that RESTARTS has been incremented: NAME READY STATUS RESTARTS AGE
liveness-exec 1/1 Running 1 1m

NEW QUESTION 9
Exhibit:

> You must switch to the correct cluster/configuration context. Failure to do so may result in a zero score.
>
> ```
> [candidate@node-1] $ kubectl config use-c
> ontext sk8s
> ```

Task

A Deployment named backend-deployment in namespace staging runs a web application on port 8081.

☛ The Deployment's manifest files can be found at

~/spicy-pikachu/backend-deployment.yaml .

Modify the Deployment specifying a readiness probe

using path /healthz .

Set initialDelaySeconds to 8 and periodSeconds to 5 .

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution:

```
File Edit View Terminal Tabs Help
Warning: Permanently added '172.31.17.21' (ECDSA) to the list of known hosts.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

candidate@node-1:~$ vi ~/spicy-pikachu/backend-deployment.yaml
candidate@node-1:~$ kubectl config use-context sk8s
Switched to context "sk8s".
candidate@node-1:~$ vim .vimrc
candidate@node-1:~$ vim ~/spicy-pikachu/backend-deployment.yaml
```
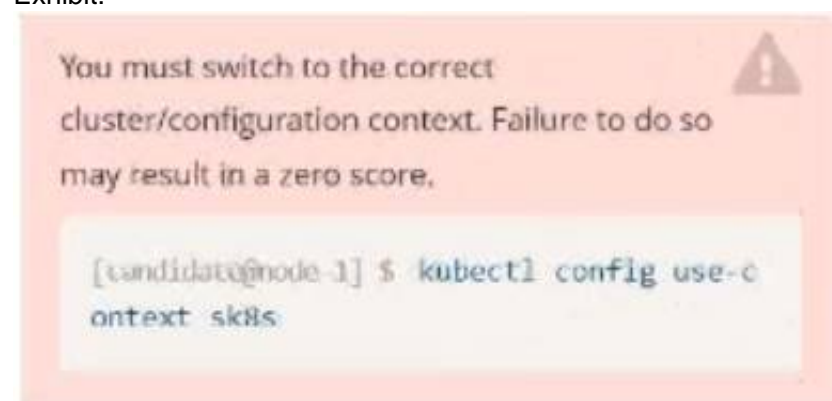
Text Description automatically generated

```
File Edit View Terminal Tabs Help
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend-deployment
  namespace: staging
spec:
  selector:
    matchLabels:
      app: nginx
  replicas: 3
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
          ports:
            - containerPort: 8081
          readinessProbe:
            initialDelaySeconds: 8
            periodSeconds: 5
            httpGet:
              path: /healthz
              port: 8081
          volumeMounts:
            - mountPath: /etc/nginx/conf.d/
              name: config
            - mountPath: /usr/share/nginx/html/
              name: www
-- INSERT --                                                      26,28        Top
```

```
File Edit View Terminal Tabs Help
Warning: Permanently added '172.31.17.21' (ECDSA) to the list of known hosts.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

candidate@node-1:~$ vi ~/spicy-pikachu/backend-deployment.yaml
candidate@node-1:~$ kubectl config use-context sk8s
Switched to context "sk8s".
candidate@node-1:~$ vim .vimrc
candidate@node-1:~$ vim ~/spicy-pikachu/backend-deployment.yaml
candidate@node-1:~$ kubectl  apply  -f ~/spicy-pikachu/backend-deployment.yaml
deployment.apps/backend-deployment configured
candidate@node-1:~$ kubectl get  pods -n staging
NAME                                 READY   STATUS    RESTARTS   AGE
backend-deployment-59d449b99d-cxct6  1/1     Running   0          20s
backend-deployment-59d449b99d-h2zjq  0/1     Running   0          9s
backend-deployment-78976f74f5-b8c85  1/1     Running   0          6h40m
backend-deployment-78976f74f5-flfsj  1/1     Running   0          6h40m
candidate@node-1:~$ kubectl get deploy -n staging
NAME                 READY   UP-TO-DATE   AVAILABLE   AGE
backend-deployment   3/3     3            3           6h40m
candidate@node-1:~$ kubectl get deploy -n staging
NAME                 READY   UP-TO-DATE   AVAILABLE   AGE
backend-deployment   3/3     3            3           6h41m
candidate@node-1:~$ vim ~/spicy-pikachu/backend-deployment.yaml
```

NEW QUESTION 10
Exhibit:

You must switch to the correct
cluster/configuration context. Failure to do so
may result in a zero score.

```
[candidate@node-1] $ kubectl config use-c
ontext sk8s
```

Task:
Create a Pod named nginx resources in the existing pod resources namespace. Specify a single container using nginx:stable image.
Specify a resource request of 300m cpus and 1G1 of memory for the Pod's container.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution:

```
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl  run nginx-resources -n pod-resources  --image=nginx:stable --dry-run=client -o yaml > hw.yaml
candidate@node-1:~$ vim hw.yaml
```

Text Description automatically generated with medium confidence

```
File Edit View Terminal Tabs Help
apiVersion: v1
kind: Pod
metadata:
  creationTimestamp: null
  labels:
    run: nginx-resources
  name: nginx-resources
  namespace: pod-resources
spec:
  containers:
  - image: nginx:stable
    name: nginx-resources
    resources:
      requests:
        cpu: 300m
        memory: "1Gi"


:wq
```

Text Description automatically generated

```
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl  run nginx-resources -n pod-resources  --image=nginx:stable --dry-run=client -o yaml > hw.yaml
candidate@node-1:~$ vim hw.yaml
candidate@node-1:~$ kubectl  create  -f hw.yaml
pod/nginx-resources created
candidate@node-1:~$ kubectl get pods -n pod-resources
NAME              READY    STATUS     RESTARTS    AGE
nginx-resources   1/1      Running    0           13s
candidate@node-1:~$ kubectl  describe  pods -n pod-resources
```

Text Description automatically generated

```
File Edit View Terminal Tabs Help
        memory:        1Gi
    Environment:   <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-dmx9j (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  kube-api-access-dmx9j:
    Type:                   Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName:          kube-root-ca.crt
    ConfigMapOptional:      <nil>
    DownwardAPI:            true
QoS Class:                  Burstable
Node-Selectors:             <none>
Tolerations:                node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                            node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type    Reason     Age   From               Message
  ....    ......     ....  ....               .......
  Normal  Scheduled  20s   default-scheduler  Successfully assigned pod-resources/nginx-resources to k8s-node-0
  Normal  Pulling    19s   kubelet            Pulling image "nginx:stable"
  Normal  Pulled     13s   kubelet            Successfully pulled image "nginx:stable" in 6.55664052s
  Normal  Created    13s   kubelet            Created container nginx-resources
  Normal  Started    12s   kubelet            Started container nginx-resources
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl  create  deploy expose -n  ckad00014  --image lfccncf/nginx:1.13.7 --dry-run=client -o yaml>
```

NEW QUESTION 10
Exhibit:

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

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution:

NEW QUESTION 13
......

THANKS FOR TRYING THE DEMO OF OUR PRODUCT

Visit Our Site to Purchase the Full Set of Actual CKAD Exam Questions With Answers.

We Also Provide Practice Exam Software That Simulates Real Exam Environment And Has Many Self-Assessment Features. Order the CKAD Product From:

https://www.2passeasy.com/dumps/CKAD/

Money Back Guarantee

CKAD Practice Exam Features:

* CKAD Questions and Answers Updated Frequently

* CKAD Practice Questions Verified by Expert Senior Certified Staff

* CKAD Most Realistic Questions that Guarantee you a Pass on Your FirstTry

* CKAD Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year