Q: 9:

Configure application data

Deploy an application using the openshift/hello-openshift or image that meets

the following requirements;

The application is part of a project named: acid

The application is named: phosphoric

The application uses a key named RESPONSE in a configuration map named sedicen

The application is running and available at

http://phosphoricacid.apps.domain20.example.com and display

Soda pop won't stop can't stop

Prerequisite:

$ podman pull docker.io/openshift/hello-openshift

$ podman login registry.ocp4.example.com:8443

Username: Developer

Password: developer

$ podman tag docker.io/openshift/hello-openshift

registry.ocp4.example.com:8443/openshift/hello-openshift

$ podman push registry.ocp4.example.com:8443/openshift/hello-openshift

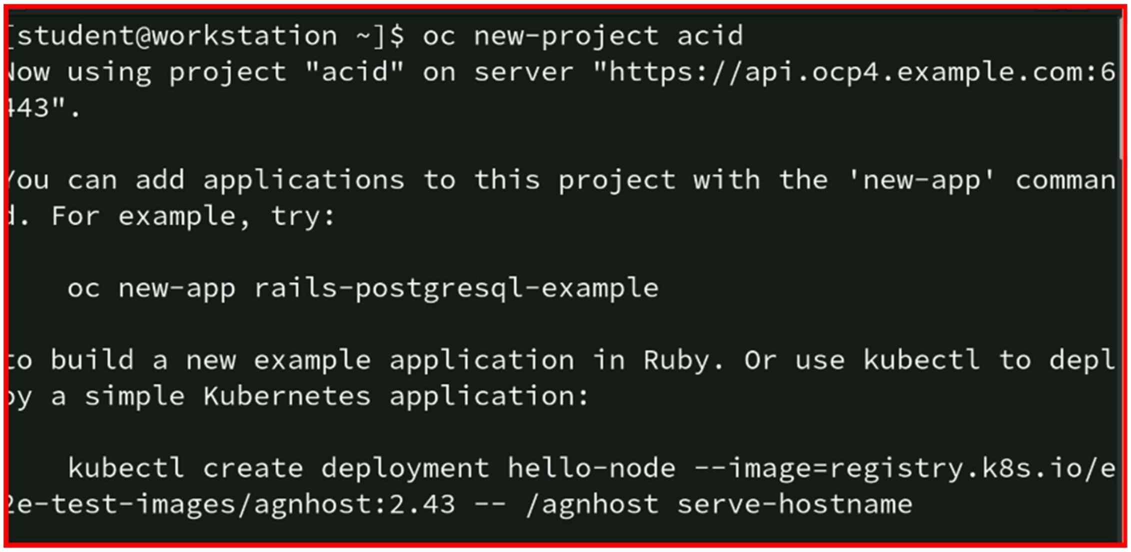
Answer:

1. Search for the image:

$ podman search openshift/hello-openshift

2. Create the project

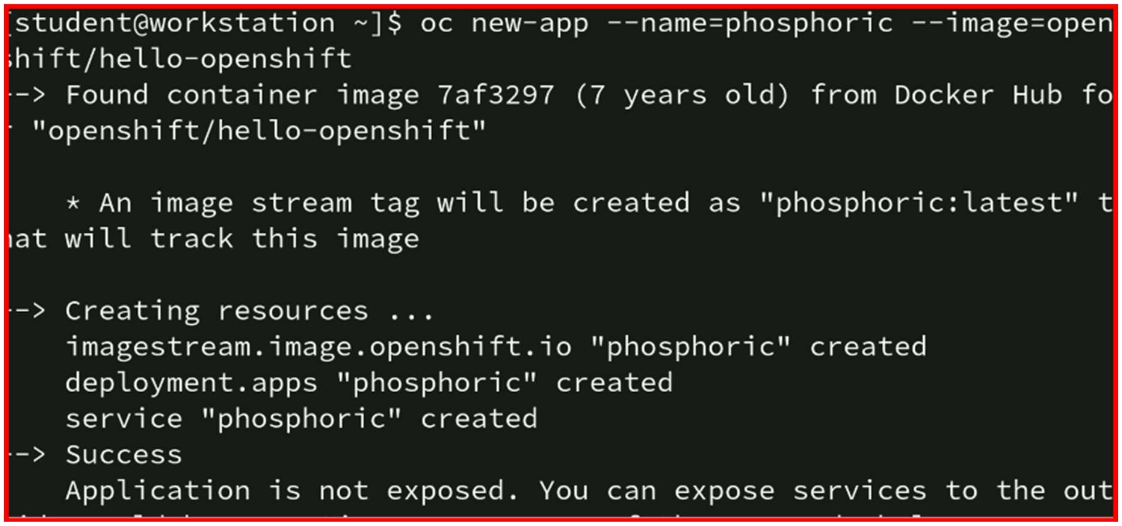
$ oc new-project acid



```
[student@workstation ~]$ oc new-project acid
Now using project "acid" on server "https://api.ocp4.example.com:6
443".

You can add applications to this project with the 'new-app' comman
d. For example, try:

    oc new-app rails-postgresql-example

to build a new example application in Ruby. Or use kubectl to depl
oy a simple Kubernetes application:

    kubectl create deployment hello-node --image=registry.k8s.io/e
2e-test-images/agnhost:2.43 -- /agnhost serve-hostname
```

3. Runthe application

$ oc new-app--name=phosphoric--image=openshift/hello-openshift

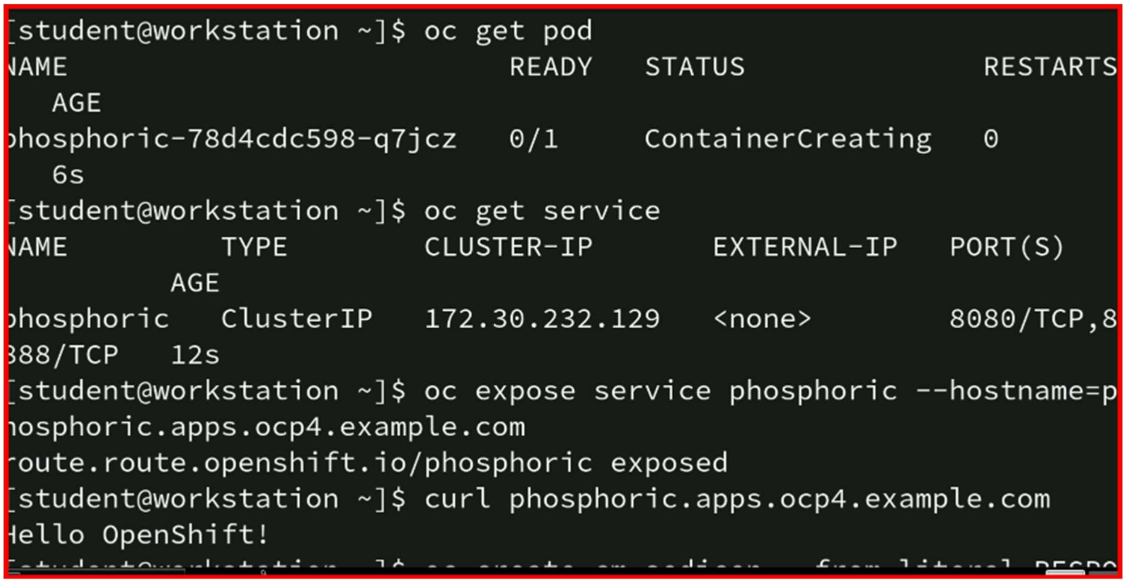


```
[student@workstation ~]$ oc new-app --name=phosphoric --image=open
shift/hello-openshift
--> Found container image 7af3297 (7 years old) from Docker Hub fo
r "openshift/hello-openshift"

    * An image stream tag will be created as "phosphoric:latest" t
hat will track this image

--> Creating resources ...
    imagestream.image.openshift.io "phosphoric" created
    deployment.apps "phosphoric" created
    service "phosphoric" created
--> Success
    Application is not exposed. You can expose services to the out
```

4. Verify

$ oc get pod

$ oc get service



```
[student@workstation ~]$ oc get pod
NAME                           READY   STATUS              RESTARTS
   AGE
phosphoric-78d4cdc598-q7jcz    0/1     ContainerCreating   0
   6s
[student@workstation ~]$ oc get service
NAME          TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)
          AGE
phosphoric    ClusterIP   172.30.232.129   <none>        8080/TCP,8
888/TCP   12s
[student@workstation ~]$ oc expose service phosphoric --hostname=p
hosphoric.apps.ocp4.example.com
route.route.openshift.io/phosphoric exposed
[student@workstation ~]$ curl phosphoric.apps.ocp4.example.com
Hello OpenShift!
```

5. Configure the route

$ oc expose servicephosphoric--hostname=phosphoric.apps.ocp4.example.com

6. Verify

$ curl phosphoric.apps.ocp4.example.com

7. Create config map

```
$ oc create cm sedicen--from-literal RESPONSE=”Soda pop won't

stop can't stop”
```

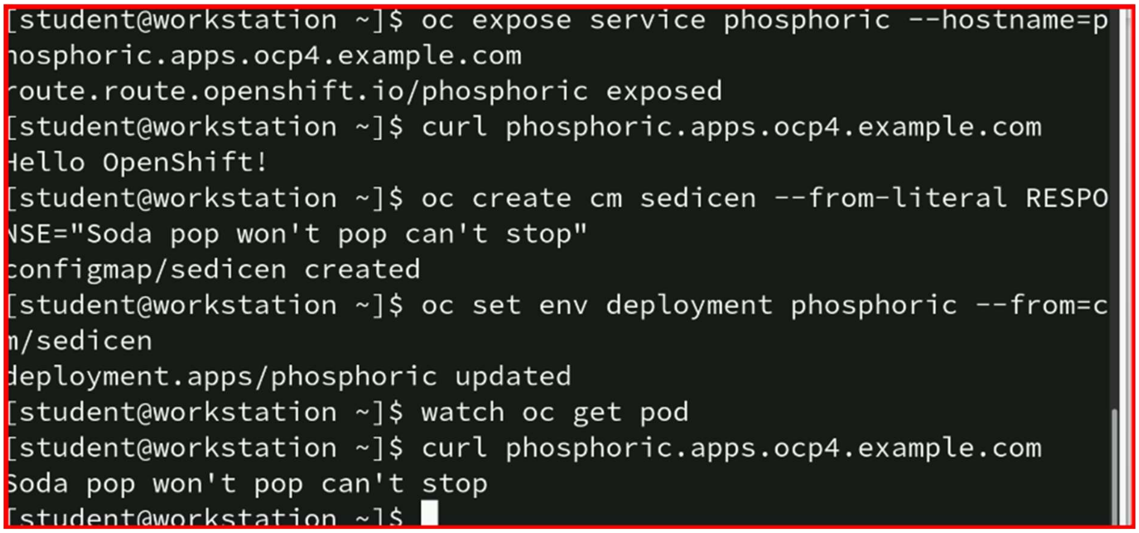
8. MapEnvironment variable with the Config map

```
$ oc set env deploymentphosphoric--from=cm/sedicen
```

9. Verify

```
$ watch oc get pod
```

```
$ curl phosphoric.apps.ocp4.example.com
```



```
[student@workstation ~]$ oc expose service phosphoric --hostname=p
hosphoric.apps.ocp4.example.com
route.route.openshift.io/phosphoric exposed
[student@workstation ~]$ curl phosphoric.apps.ocp4.example.com
Hello OpenShift!
[student@workstation ~]$ oc create cm sedicen --from-literal RESPO
NSE="Soda pop won't pop can't stop"
configmap/sedicen created
[student@workstation ~]$ oc set env deployment phosphoric --from=c
m/sedicen
deployment.apps/phosphoric updated
[student@workstation ~]$ watch oc get pod
[student@workstation ~]$ curl phosphoric.apps.ocp4.example.com
Soda pop won't pop can't stop
[student@workstation ~]$
```