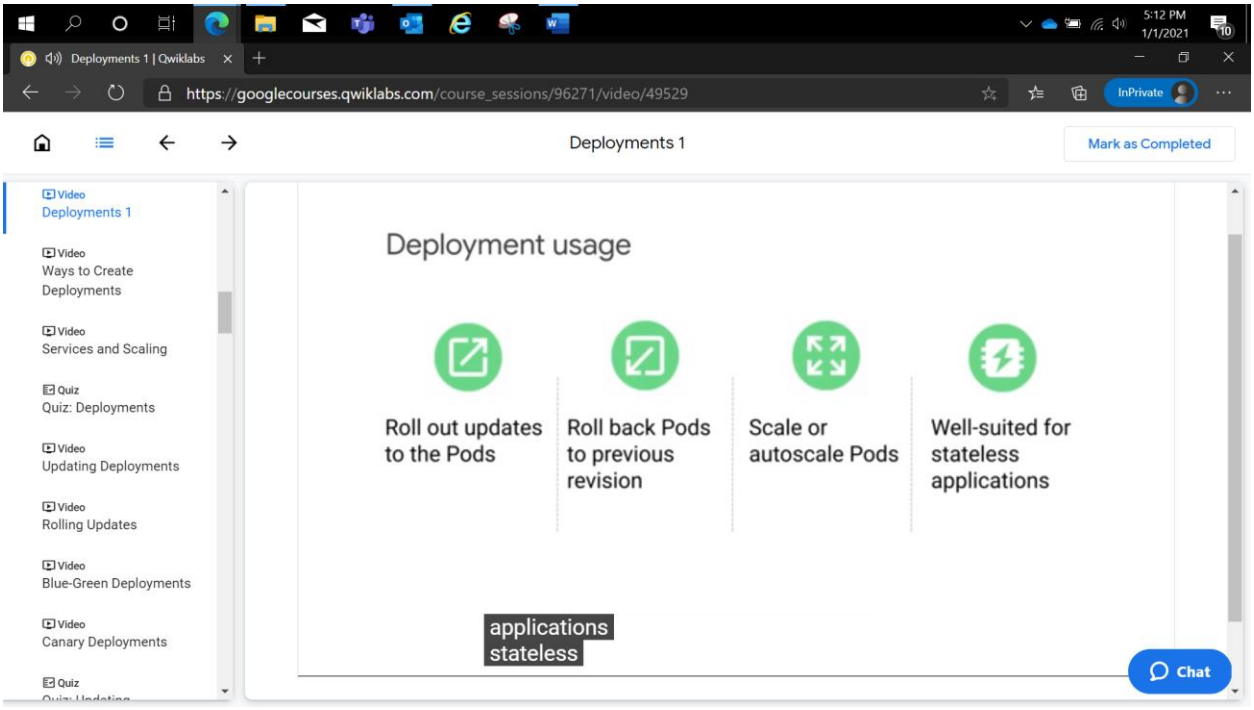
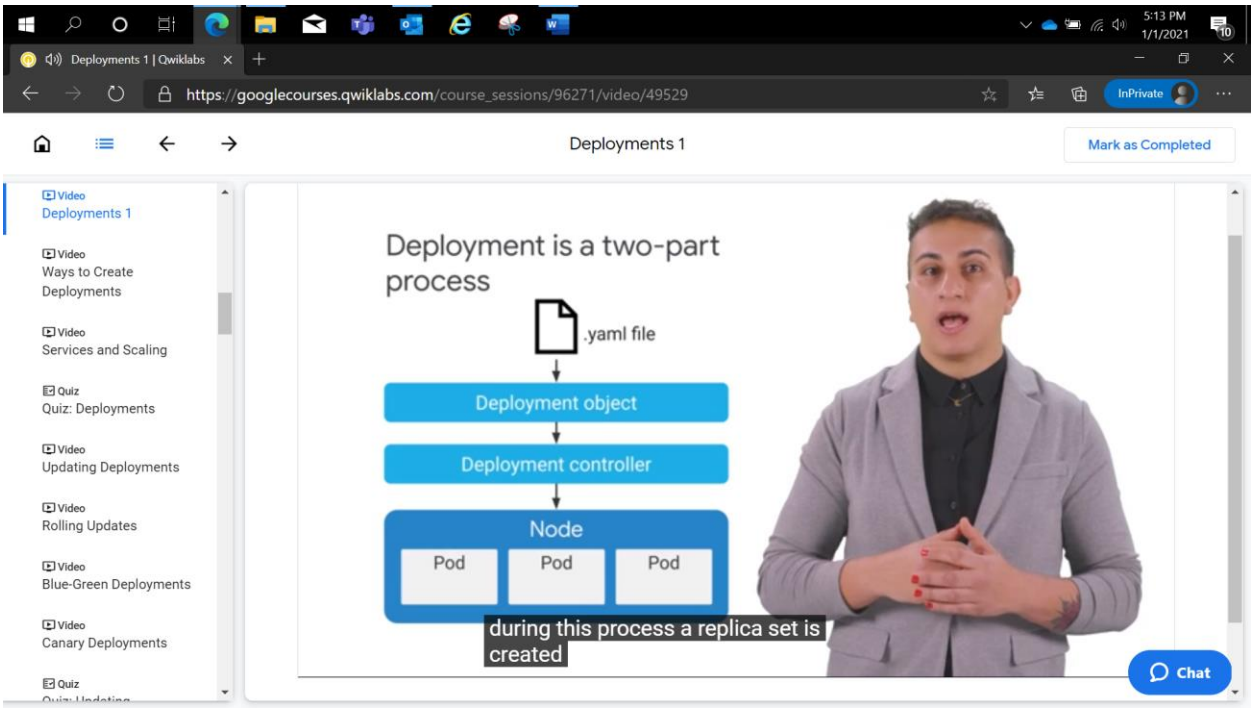
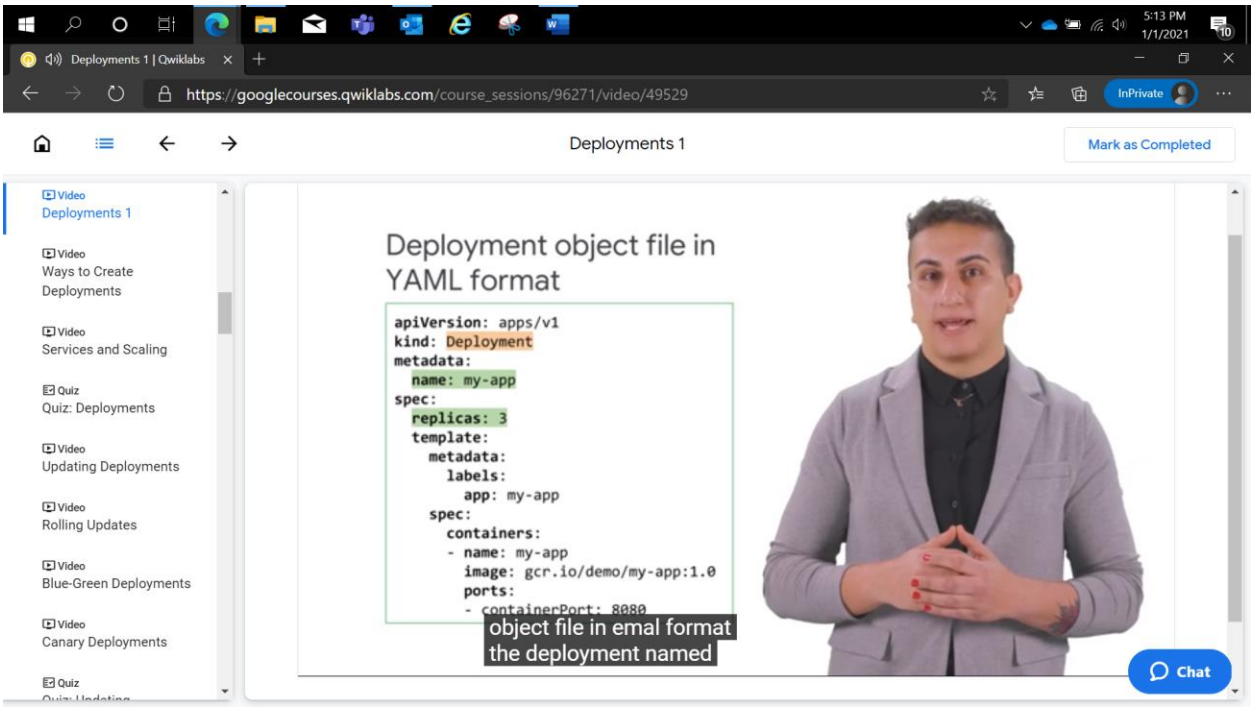
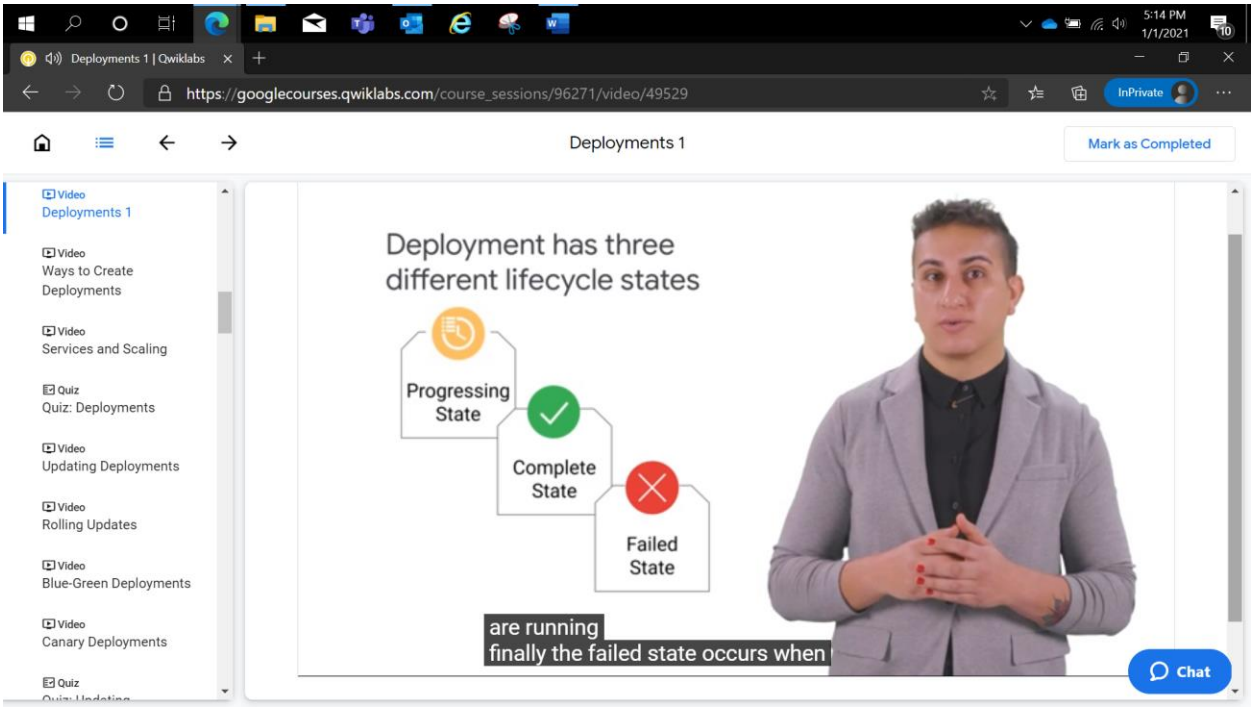
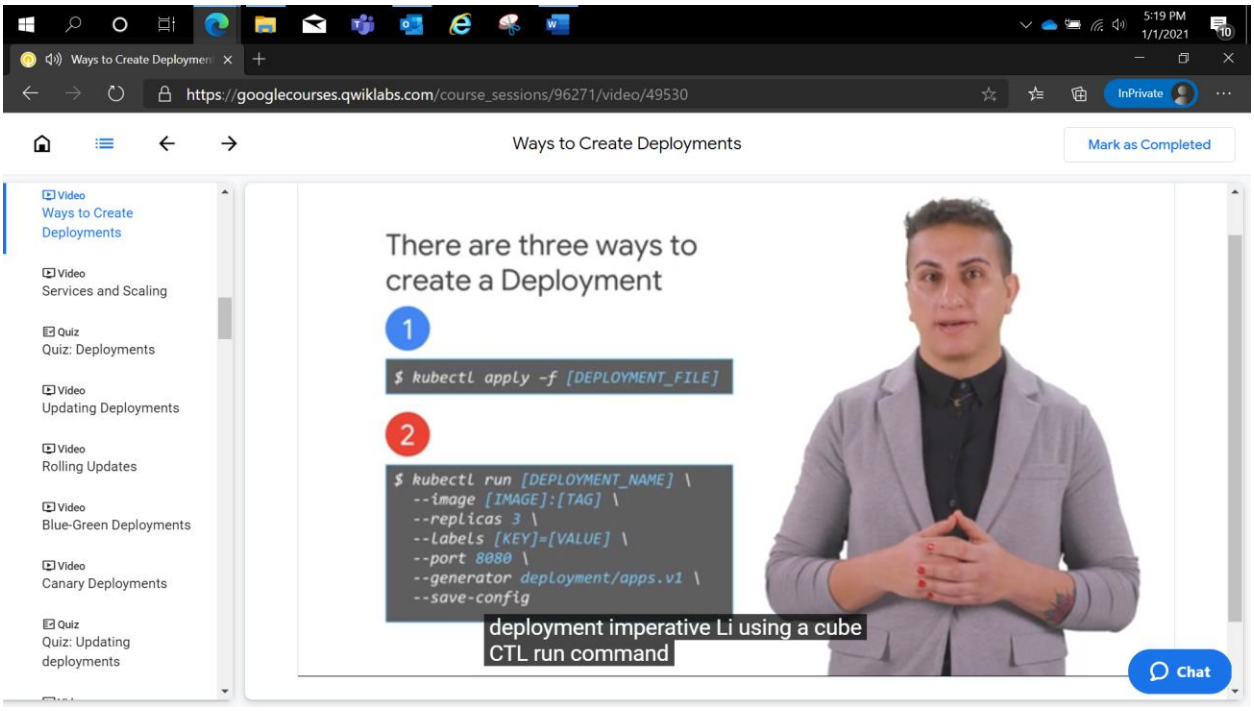
Chapter 2.2.2 – Deployment,Jobs and scaling

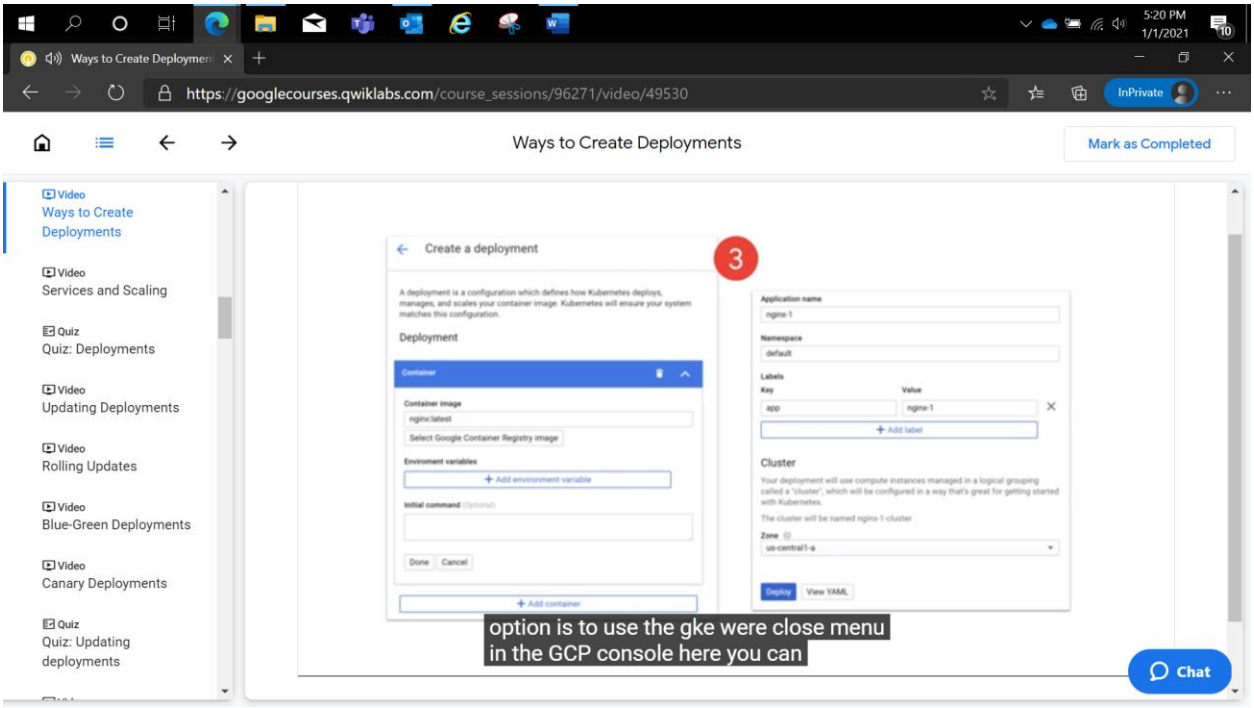


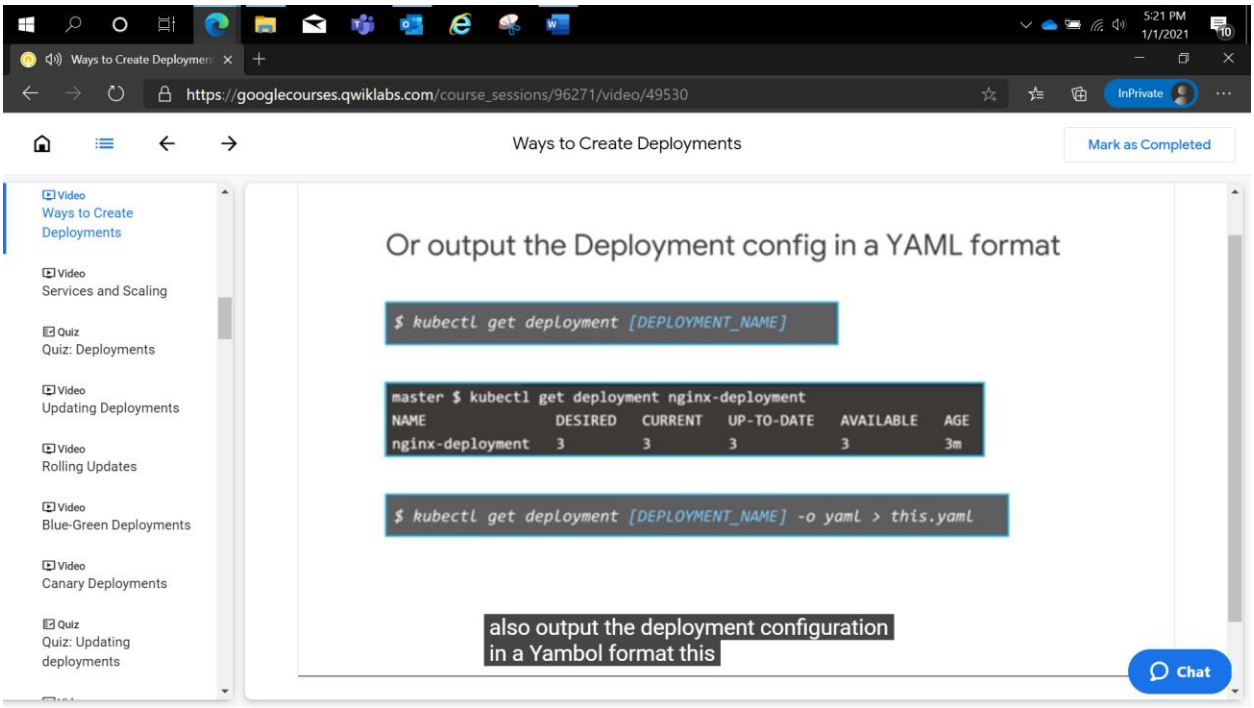


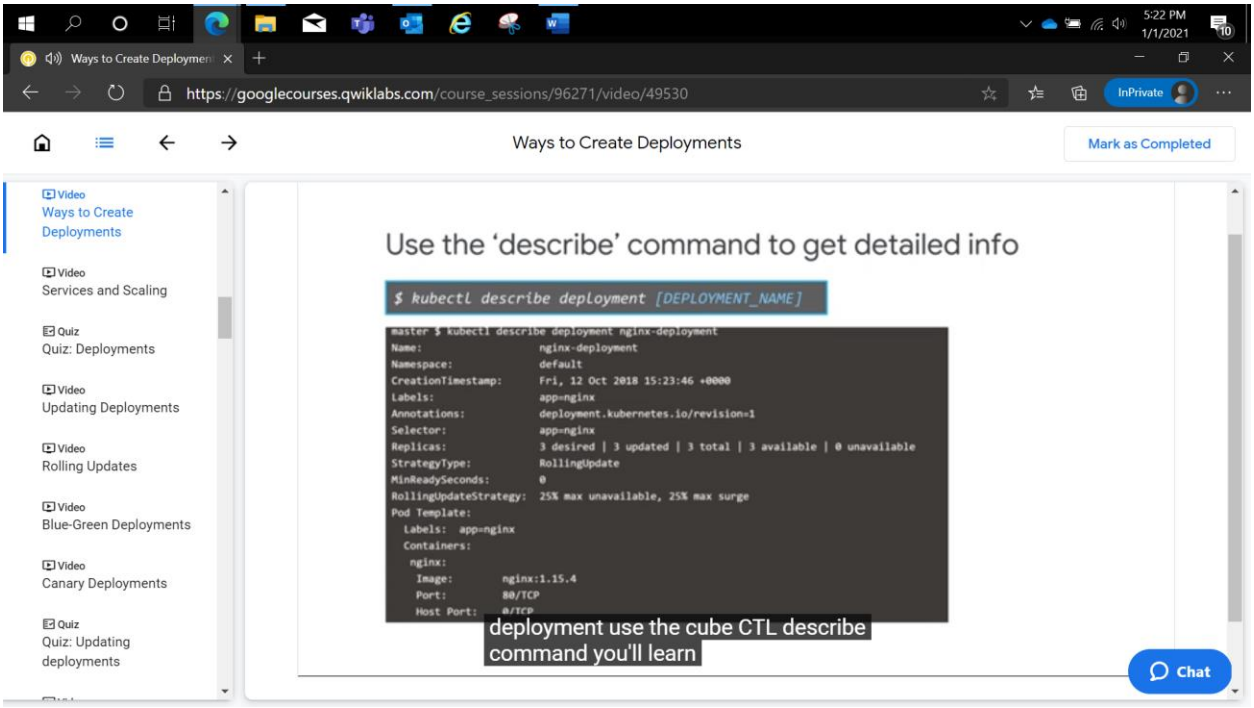

Deployments 1 | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49529
InPrivate
5:13 PM
1/1/2021
Deployments 1
Mark as Completed
Video
Deployments 1
Video
Ways to Create Deployments
Video
Services and Scaling
Quiz
Quiz: Deployments
Video
Updating Deployments
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Deployment object file in YAML format
apiVersion: apps/v1
kind: Deployment
metadata:
name: my-app
spec:
replicas: 3
template:
metadata:
labels:
app: my-app
spec:
containers:
- name: my-app
image: gcr.io/demo/my-app:1.0
ports:
- containerPort: 8080
object file in emal format
the deployment named
Chat

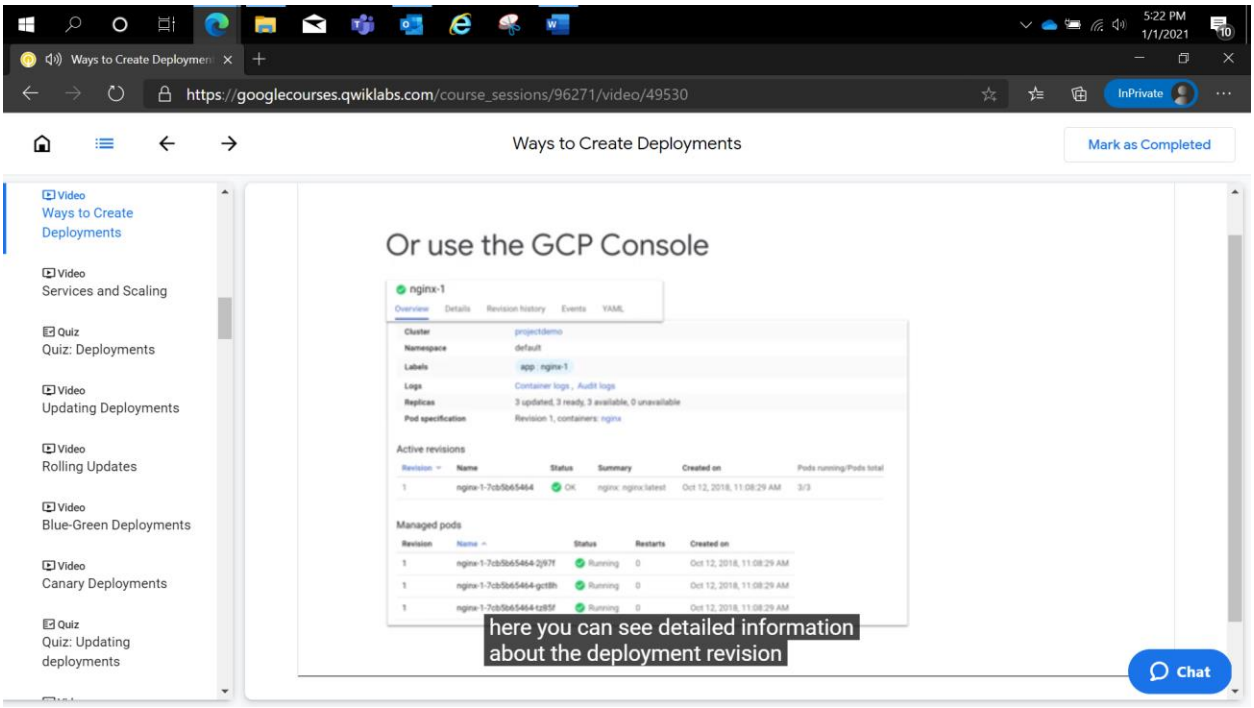

Deployments 1 | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49529
InPrivate
5:14 PM
1/1/2021
Deployments 1
Mark as Completed
Video
Deployments 1
Video
Ways to Create Deployments
Video
Services and Scaling
Quiz
Quiz: Deployments
Video
Updating Deployments
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Deployment has three different lifecycle states
Progressing State
Complete State
Failed State
are running
finally the failed state occurs when
Chat

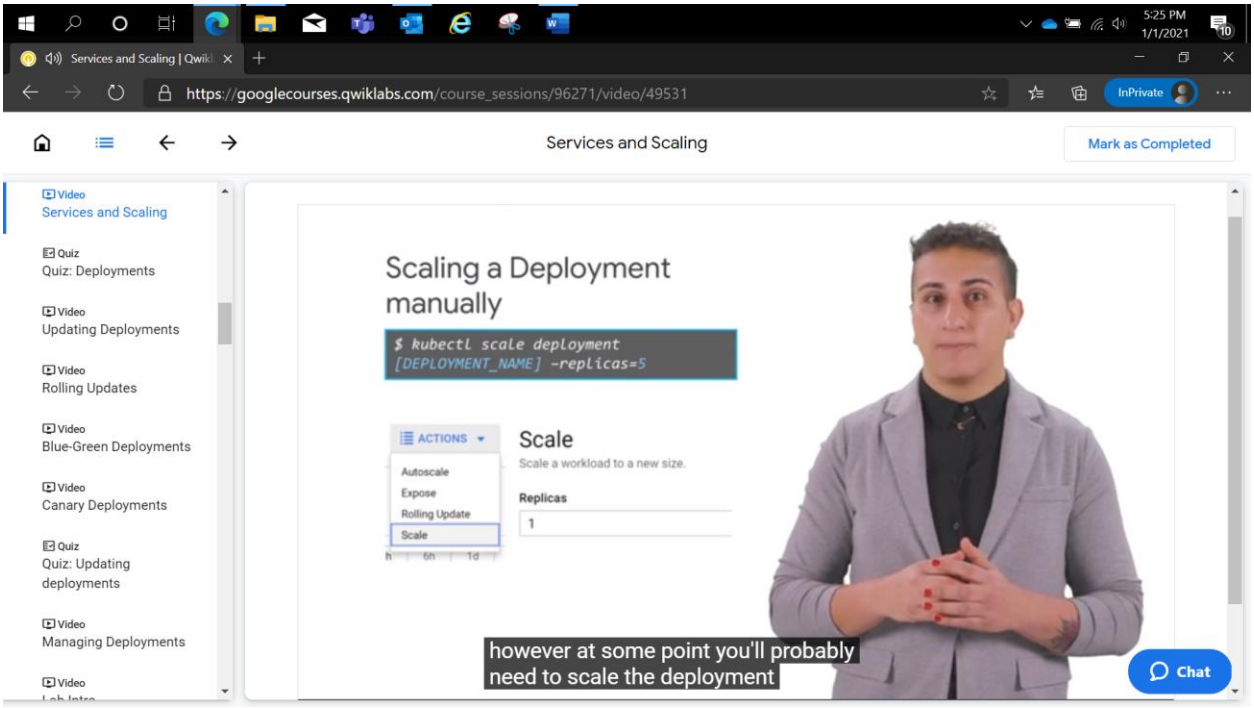

Ways to Create Deployments
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49530
InPrivate
5:19 PM
1/1/2021
Ways to Create Deployments
Mark as Completed
Video
Ways to Create Deployments
Video
Services and Scaling
Quiz
Quiz: Deployments
Video
Updating Deployments
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
There are three ways to create a Deployment
1
$ kubectl apply -f [DEPLOYMENT_FILE]
2
$ kubectl run [DEPLOYMENT_NAME] \
--image [IMAGE]:[TAG] \
--replicas 3 \
--labels [KEY]=[VALUE] \
--port 8080 \
--generator deployment/apps.v1 \
--save-config
deployment imperative Li using a cube
CTL run command
Chat

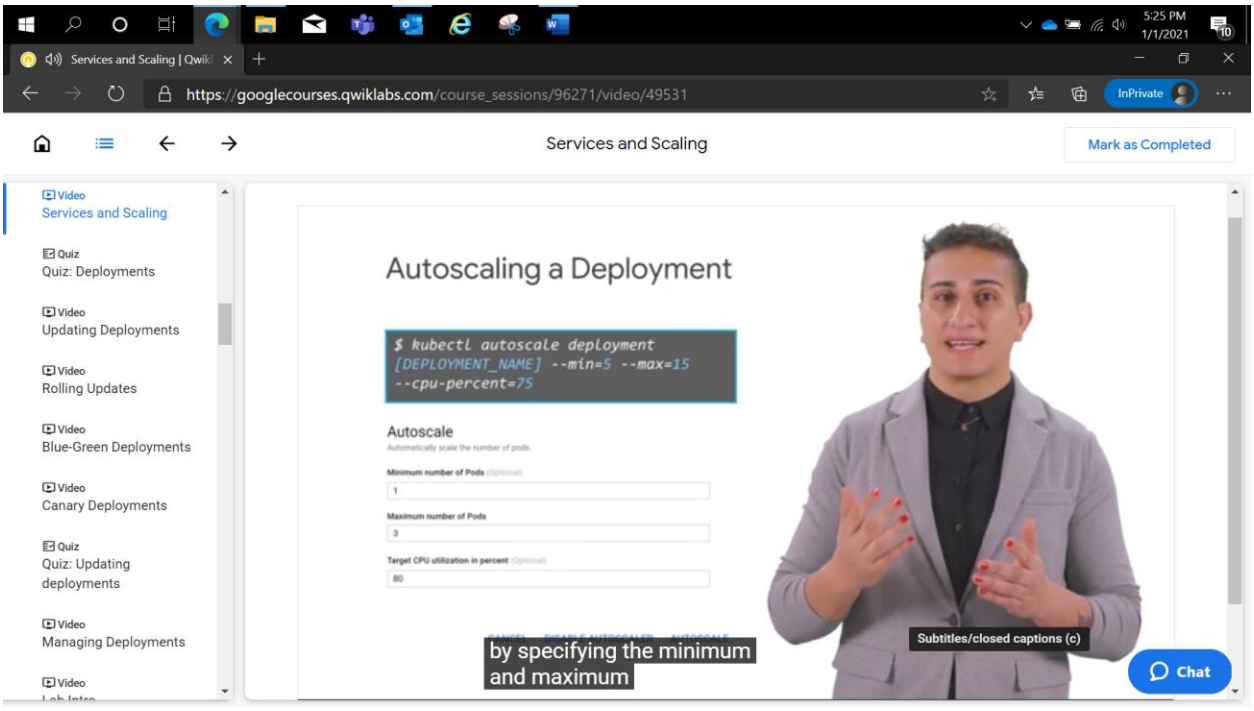

Ways to Create Deployments
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49530
InPrivate
5:20 PM
1/1/2021
Ways to Create Deployments
Mark as Completed
Video
Ways to Create Deployments
Video
Services and Scaling
Quiz
Quiz: Deployments
Video
Updating Deployments
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Create a deployment
3
Deployment
Container
Container image
Select Google Container Registry image
Environment variables
Add environment variable
Initial command
Done
Cancel
Add container
Application name
Namespace
default
Labels
Key
Value
app
Add label
Cluster
Zone
us-central1-a
Deploy
View YAML
option is to use the gke were close menu
in the GCP console here you can
Chat

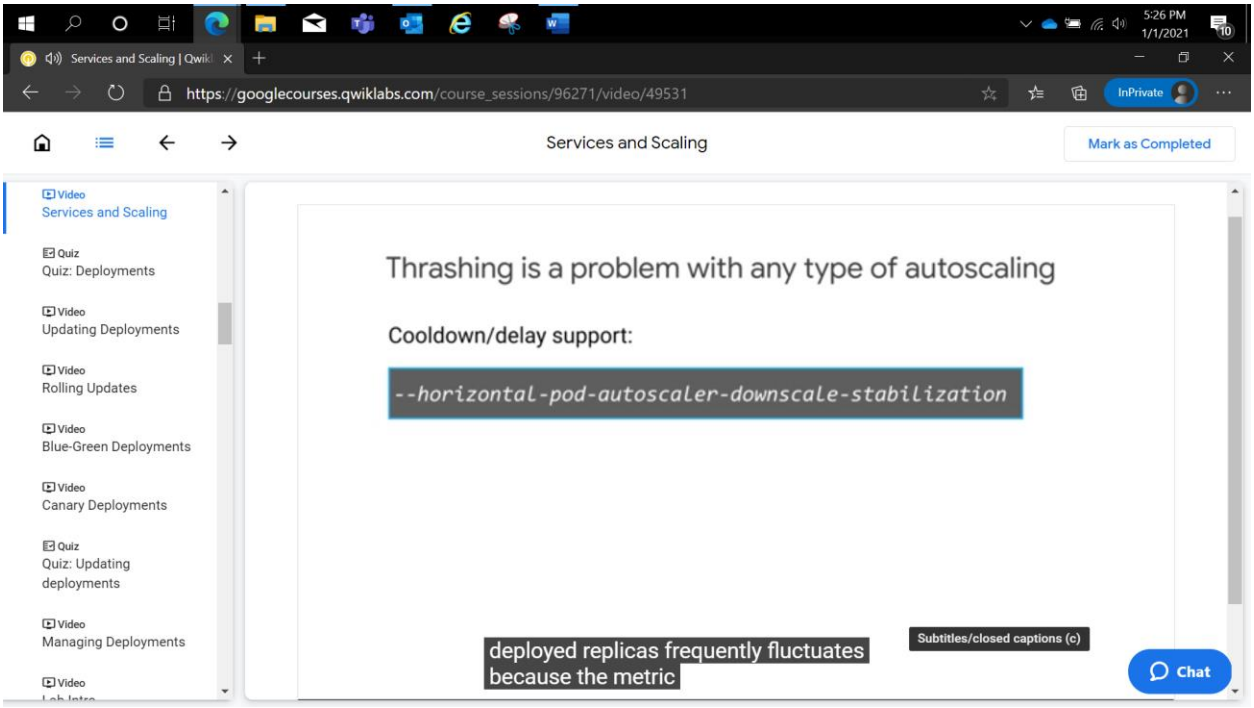

Ways to Create Deployments
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49530
Ways to Create Deployments
Mark as Completed
Video Ways to Create Deployments
Video Services and Scaling
Quiz Quiz: Deployments
Video Updating Deployments
Video Rolling Updates
Video Blue-Green Deployments
Video Canary Deployments
Quiz Quiz: Updating deployments
Or output the Deployment config in a YAML format
$ kubectl get deployment [DEPLOYMENT_NAME]
master $ kubectl get deployment nginx-deployment
NAME DESIRED CURRENT UP-TO-DATE AVAILABLE AGE
nginx-deployment 3 3 3 3 3m
$ kubectl get deployment [DEPLOYMENT_NAME] -o yaml > this.yaml
also output the deployment configuration in a Yambol format this
Chat

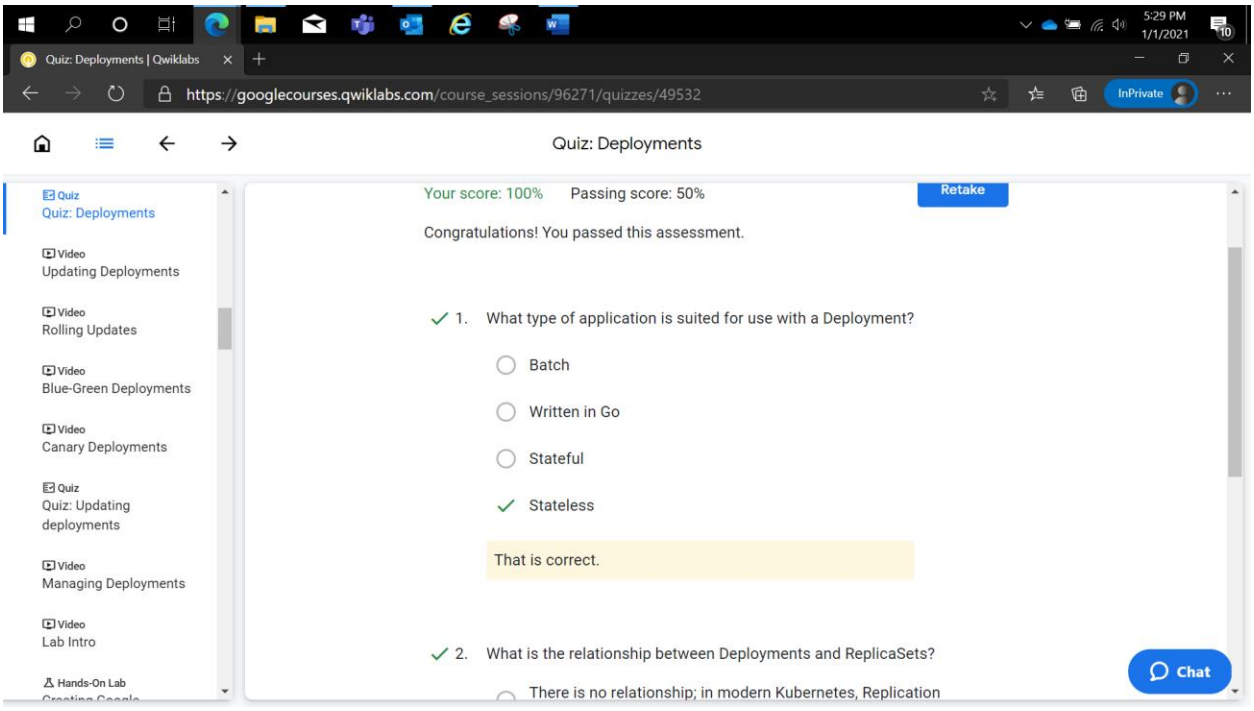

Ways to Create Deployments
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49530
Ways to Create Deployments
Mark as Completed
Video Ways to Create Deployments
Video Services and Scaling
Quiz Quiz: Deployments
Video Updating Deployments
Video Rolling Updates
Video Blue-Green Deployments
Video Canary Deployments
Quiz Quiz: Updating deployments
Use the 'describe' command to get detailed info
$ kubectl describe deployment [DEPLOYMENT_NAME]
master $ kubectl describe deployment nginx-deployment
Name: nginx-deployment
Namespace: default
CreationTimestamp: Fri, 12 Oct 2018 15:23:46 +0000
Labels: app=nginx
Annotations: deployment.kubernetes.io/revision=1
Selector: app=nginx
Replicas: 3 desired | 3 updated | 3 total | 3 available | 0 unavailable
StrategyType: RollingUpdate
MinReadySeconds: 0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
Labels: app=nginx
Containers:
nginx:
Image: nginx:1.15.4
Port: 80/TCP
Host Port: 0/TCP
deployment use the cube CTL describe command you'll learn
Chat

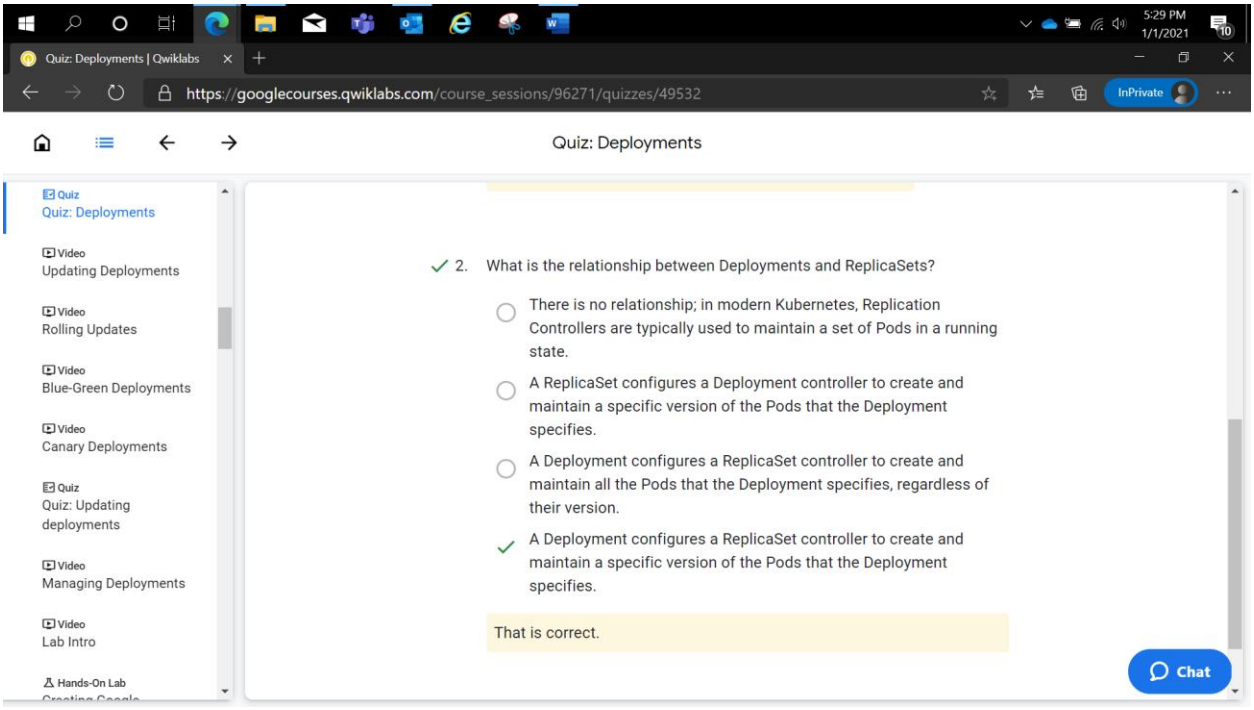

Ways to Create Deployments
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49530
Ways to Create Deployments
Mark as Completed
Video
Ways to Create Deployments
Video
Services and Scaling
Quiz
Quiz: Deployments
Video
Updating Deployments
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Or use the GCP Console
here you can see detailed information about the deployment revision
Chat

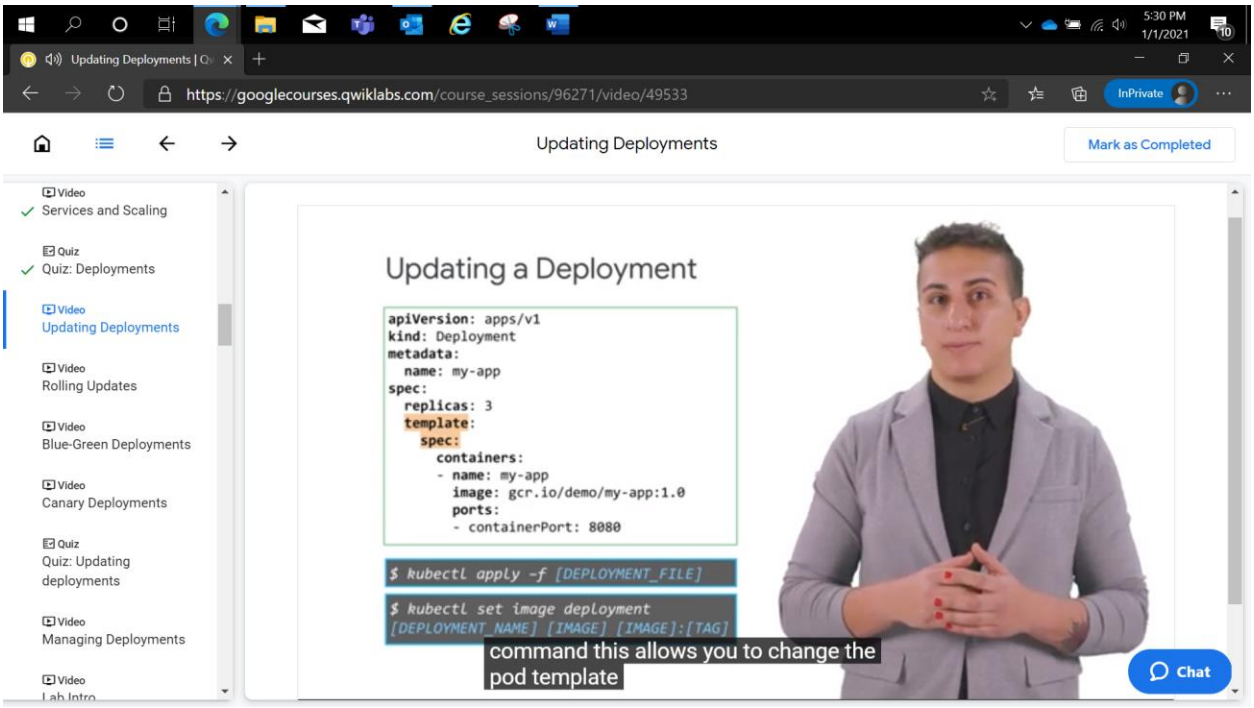

Services and Scaling
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49531
Services and Scaling
Mark as Completed
Video
Services and Scaling
Quiz
Quiz: Deployments
Video
Updating Deployments
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Scaling a Deployment manually
$ kubectl scale deployment [DEPLOYMENT_NAME] -replicas=5
ACTIONS
Autoscale
Expose
Rolling Update
Scale
Scale
Scale a workload to a new size.
Replicas
1
however at some point you'll probably need to scale the deployment
Chat

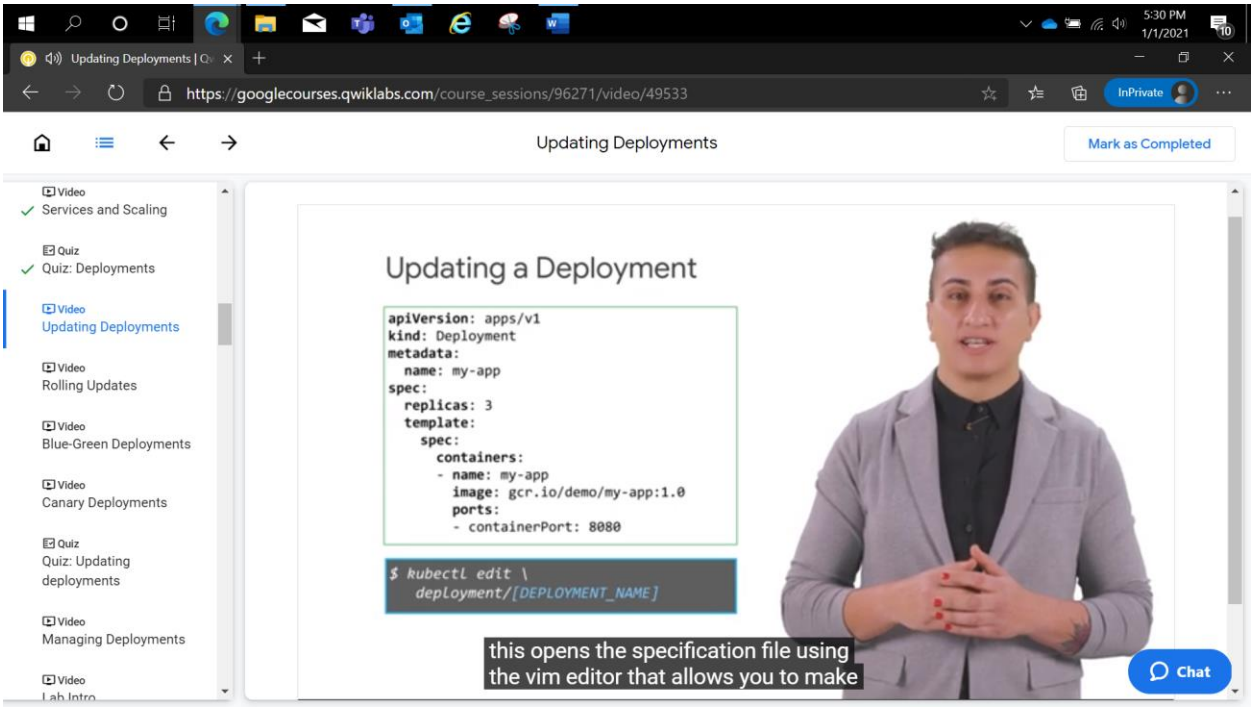

Services and Scaling | Qwikl
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49531
InPrivate
5:25 PM
1/1/2021
Services and Scaling
Mark as Completed
Video
Services and Scaling
Quiz
Quiz: Deployments
Video
Updating Deployments
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Autoscaling a Deployment
$ kubectl autoscale deployment [DEPLOYMENT_NAME] --min=5 --max=15 --cpu-percent=75
Autoscale
Minimum number of Pods
1
Maximum number of Pods
3
Target CPU utilization in percent
80
by specifying the minimum and maximum
Subtitles/closed captions (c)
Chat

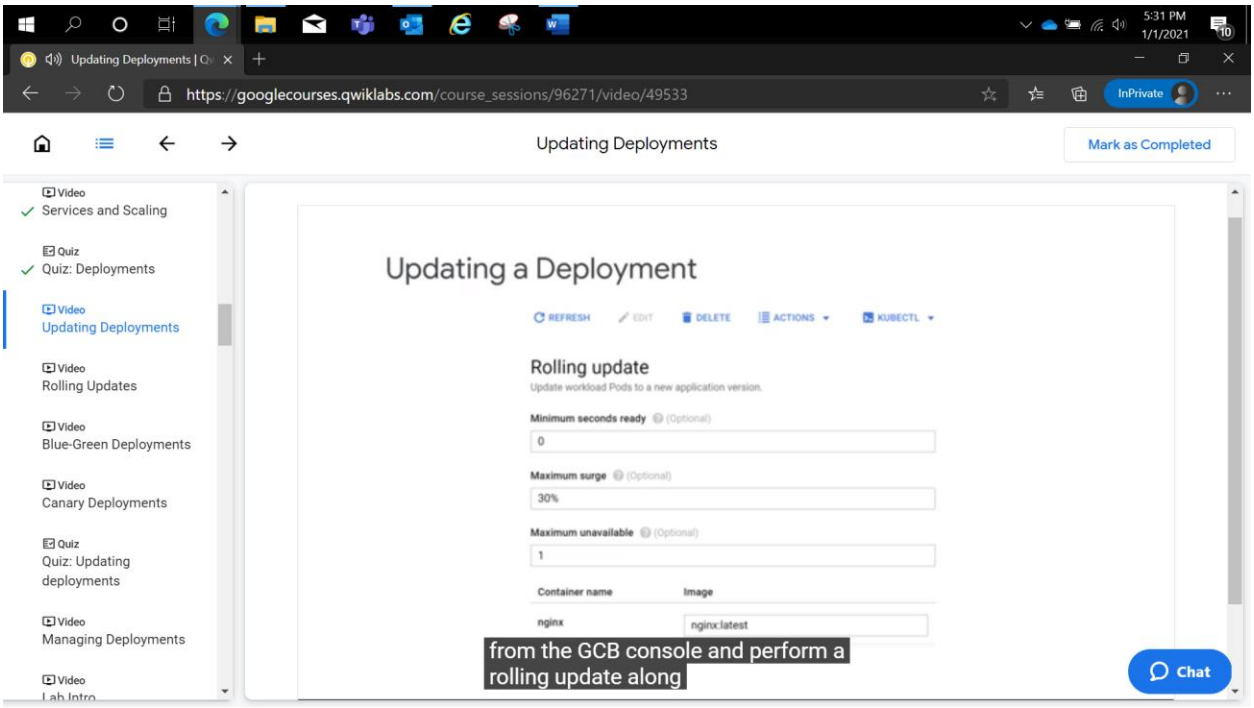

Services and Scaling | Qwikl
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49531
InPrivate
5:26 PM
1/1/2021
Services and Scaling
Mark as Completed
Video
Services and Scaling
Quiz
Quiz: Deployments
Video
Updating Deployments
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Thrashing is a problem with any type of autoscaling
Cooldown/delay support:
--horizontal-pod-autoscaler-downscale-stabilization
deployed replicas frequently fluctuates because the metric
Subtitles/closed captions (c)
Chat

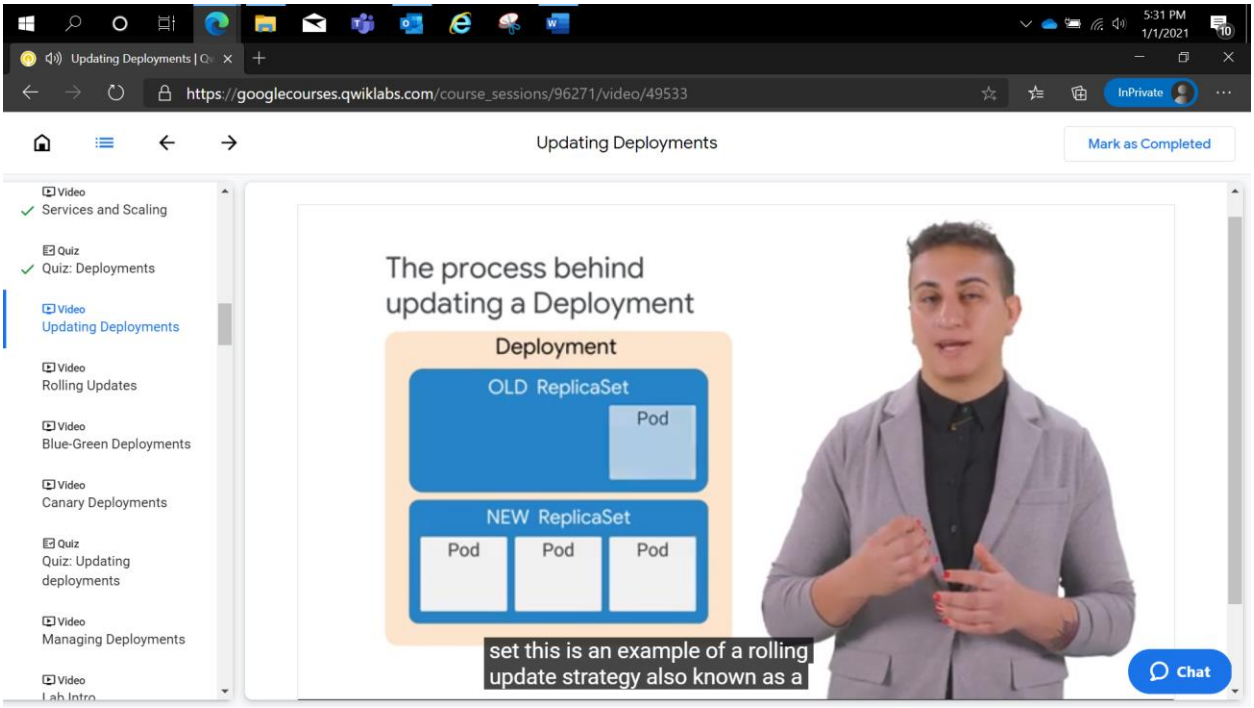

Quiz: Deployments | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49532
InPrivate
5:29 PM
1/1/2021
Quiz: Deployments
Quiz
Quiz: Deployments
Video
Updating Deployments
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Your score: 100%
Passing score: 50%
Retake
Congratulations! You passed this assessment.
1. What type of application is suited for use with a Deployment?
Batch
Written in Go
Stateful
Stateless
That is correct.
2. What is the relationship between Deployments and ReplicaSets?
There is no relationship; in modern Kubernetes, Replication
Chat

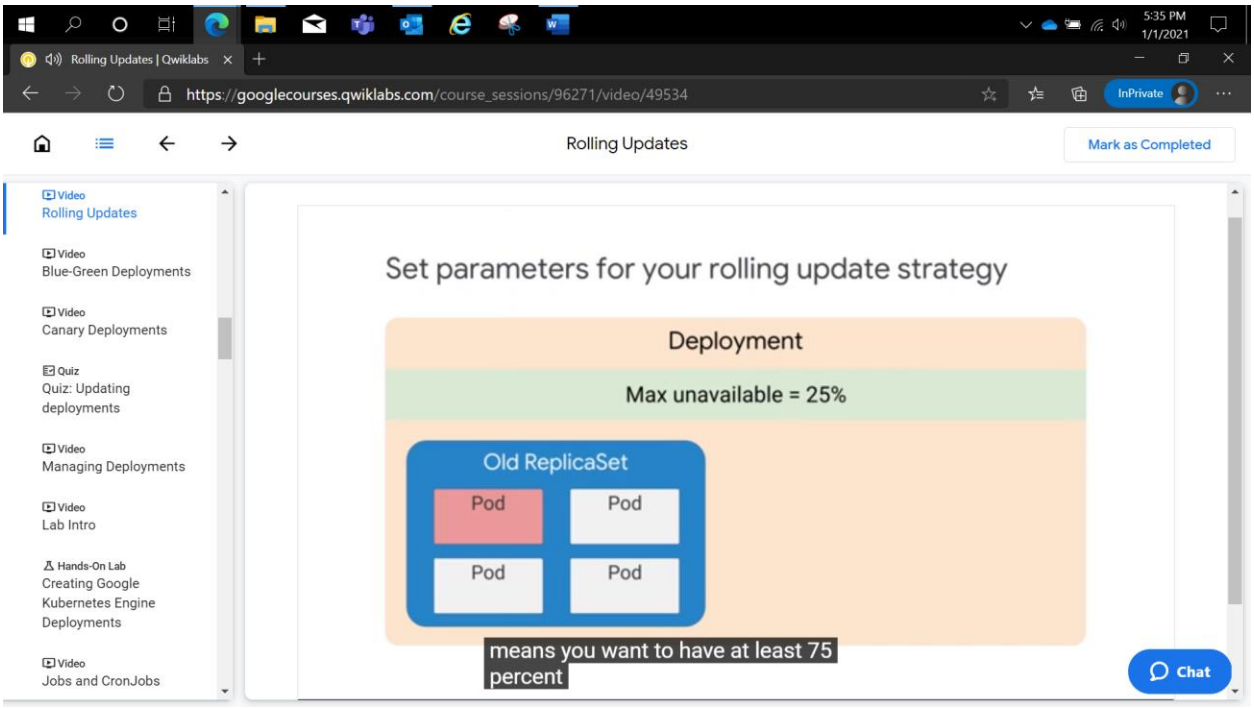

Quiz: Deployments | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49532
InPrivate
5:29 PM
1/1/2021
Quiz: Deployments
Quiz
Quiz: Deployments
Video
Updating Deployments
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
2. What is the relationship between Deployments and ReplicaSets?
There is no relationship; in modern Kubernetes, Replication Controllers are typically used to maintain a set of Pods in a running state.
A ReplicaSet configures a Deployment controller to create and maintain a specific version of the Pods that the Deployment specifies.
A Deployment configures a ReplicaSet controller to create and maintain all the Pods that the Deployment specifies, regardless of their version.
A Deployment configures a ReplicaSet controller to create and maintain a specific version of the Pods that the Deployment specifies.
That is correct.
Chat

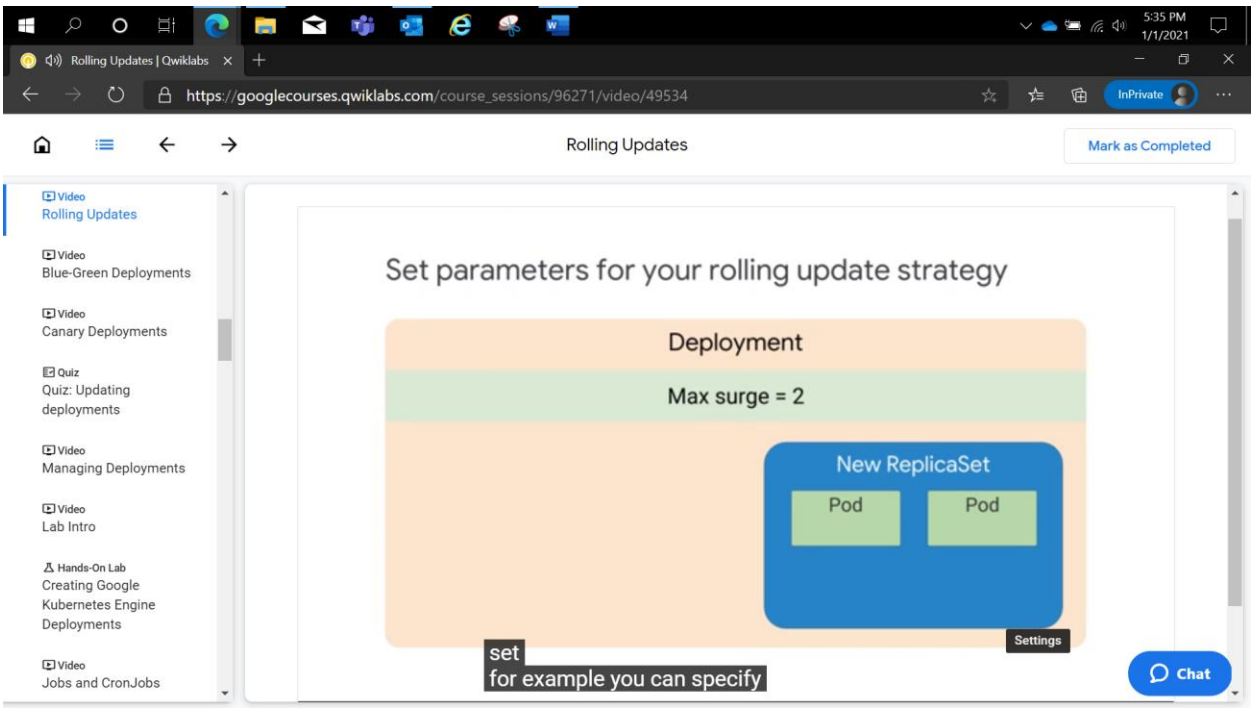

Updating Deployments | Q
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49533
InPrivate
5:30 PM
1/1/2021
Updating Deployments
Mark as Completed
Video
Services and Scaling
Quiz
Quiz: Deployments
Video
Updating Deployments
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Updating a Deployment
apiVersion: apps/v1
kind: Deployment
metadata:
name: my-app
spec:
replicas: 3
template:
spec:
containers:
- name: my-app
image: gcr.io/demo/my-app:1.0
ports:
- containerPort: 8080
$ kubectl apply -f [DEPLOYMENT_FILE]
$ kubectl set image deployment [DEPLOYMENT_NAME] [IMAGE] [IMAGE]:[TAG]
command this allows you to change the pod template
Chat

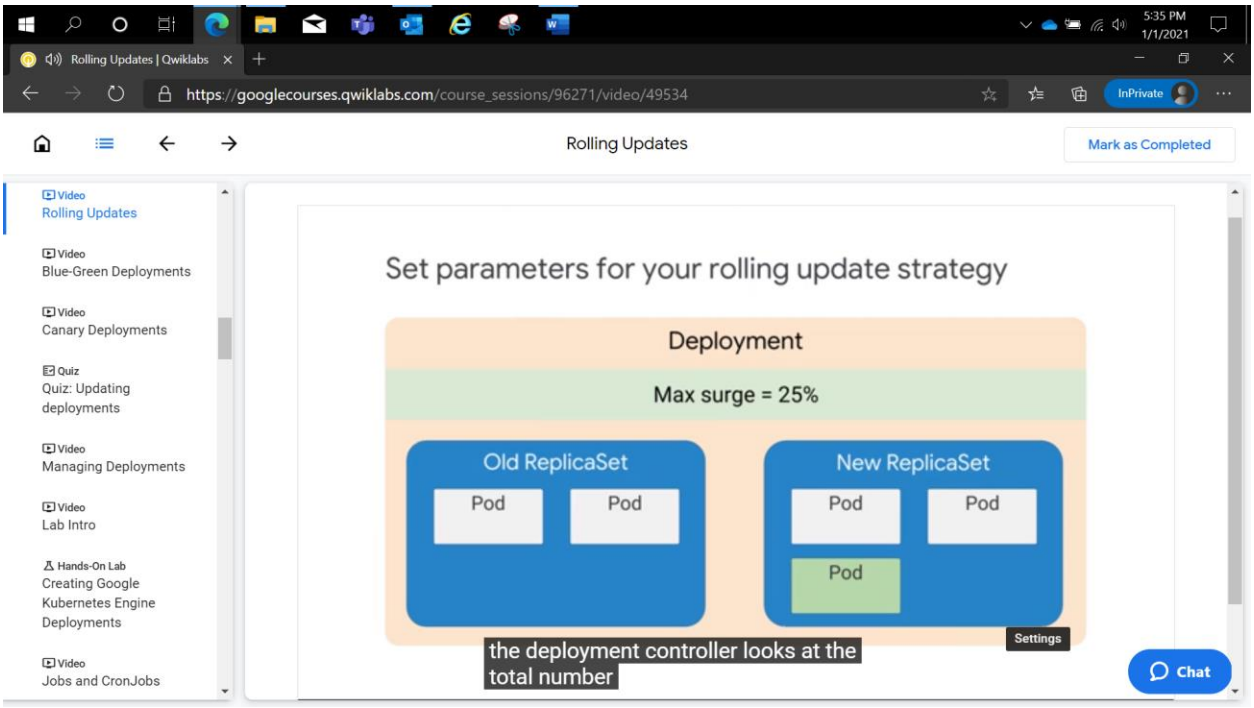

Updating Deployments | Q
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49533
InPrivate
5:30 PM
1/1/2021
Updating Deployments
Mark as Completed
Video
Services and Scaling
Quiz
Quiz: Deployments
Video
Updating Deployments
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Updating a Deployment
apiVersion: apps/v1
kind: Deployment
metadata:
name: my-app
spec:
replicas: 3
template:
spec:
containers:
- name: my-app
image: gcr.io/demo/my-app:1.0
ports:
- containerPort: 8080
$ kubectl edit \
deployment/[DEPLOYMENT_NAME]
this opens the specification file using the vim editor that allows you to make
Chat

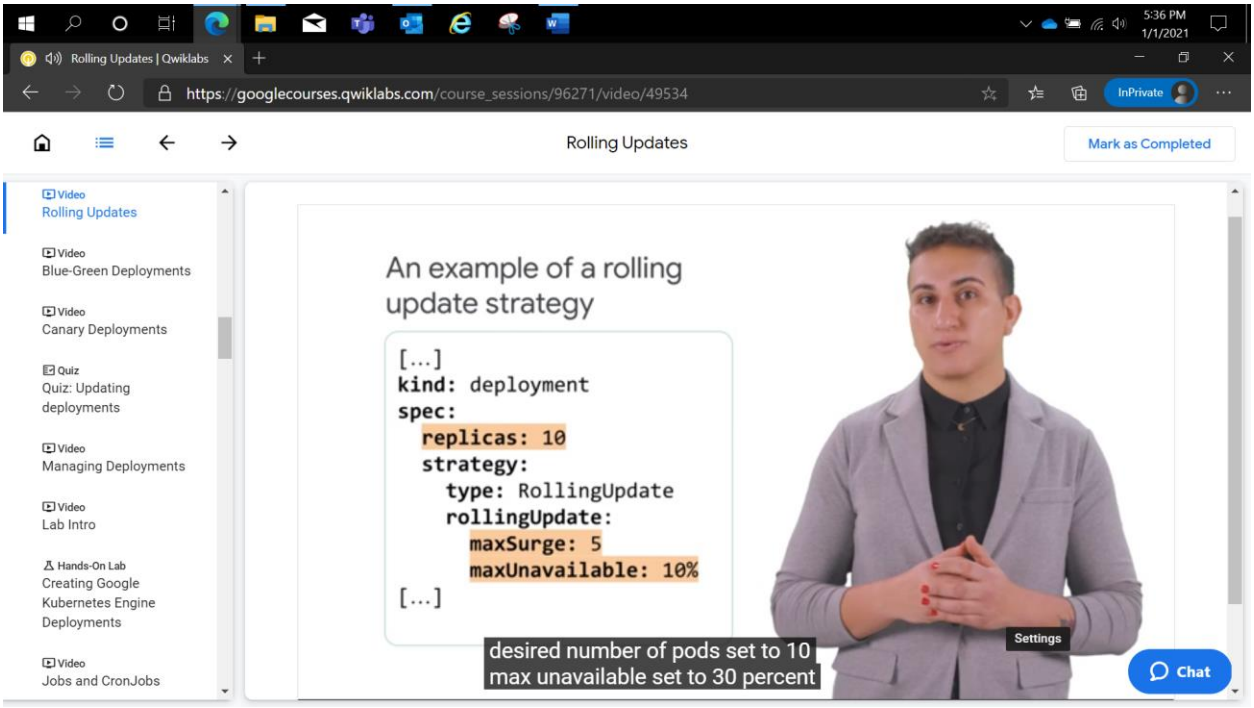

Updating Deployments | Q
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49533
InPrivate
5:31 PM
1/1/2021
Updating Deployments
Mark as Completed
Video
Services and Scaling
Quiz
Quiz: Deployments
Video
Updating Deployments
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Updating a Deployment
REFRESH
EDIT
DELETE
ACTIONS
KUBECTL
Rolling update
Update workload Pods to a new application version.
Minimum seconds ready (Optional)
0
Maximum surge (Optional)
30%
Maximum unavailable (Optional)
1
Container name
Image
nginx
nginx:latest
from the GCB console and perform a rolling update along
Chat

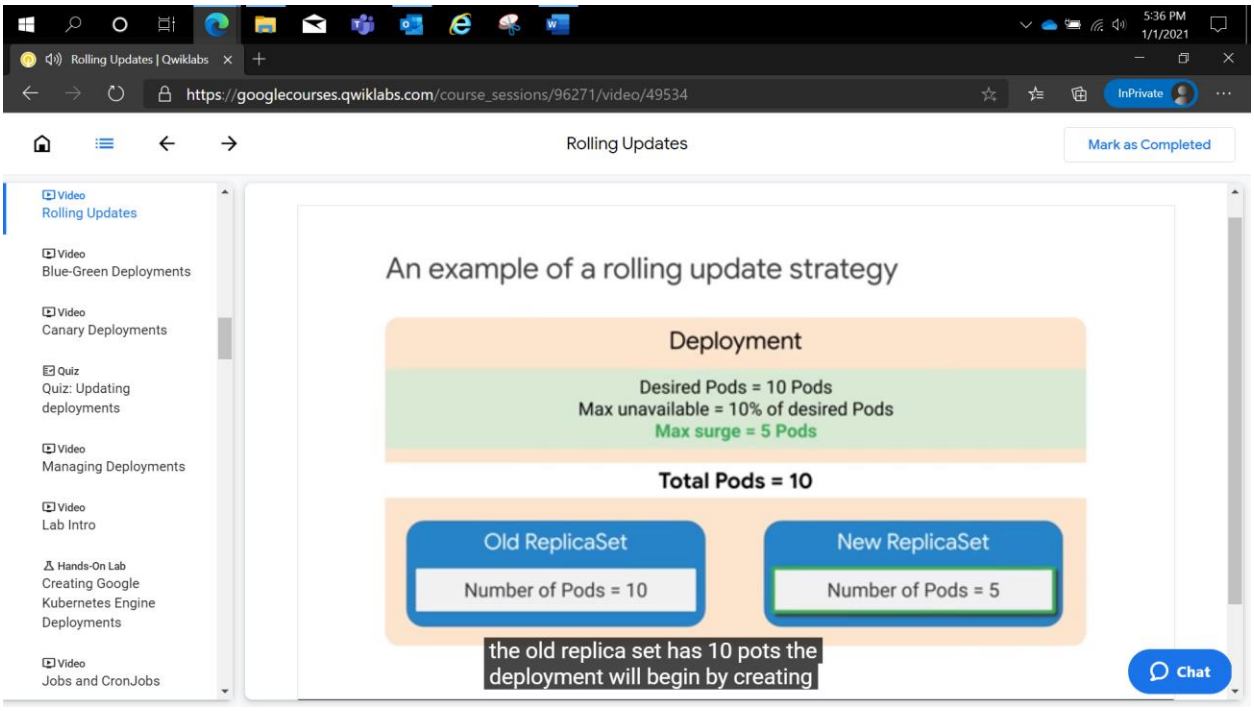

Updating Deployments | Q
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49533
InPrivate
5:31 PM
1/1/2021
Updating Deployments
Mark as Completed
Video
Services and Scaling
Quiz
Quiz: Deployments
Video
Updating Deployments
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
The process behind updating a Deployment
Deployment
OLD ReplicaSet
Pod
NEW ReplicaSet
Pod
Pod
Pod
set this is an example of a rolling update strategy also known as a
Chat

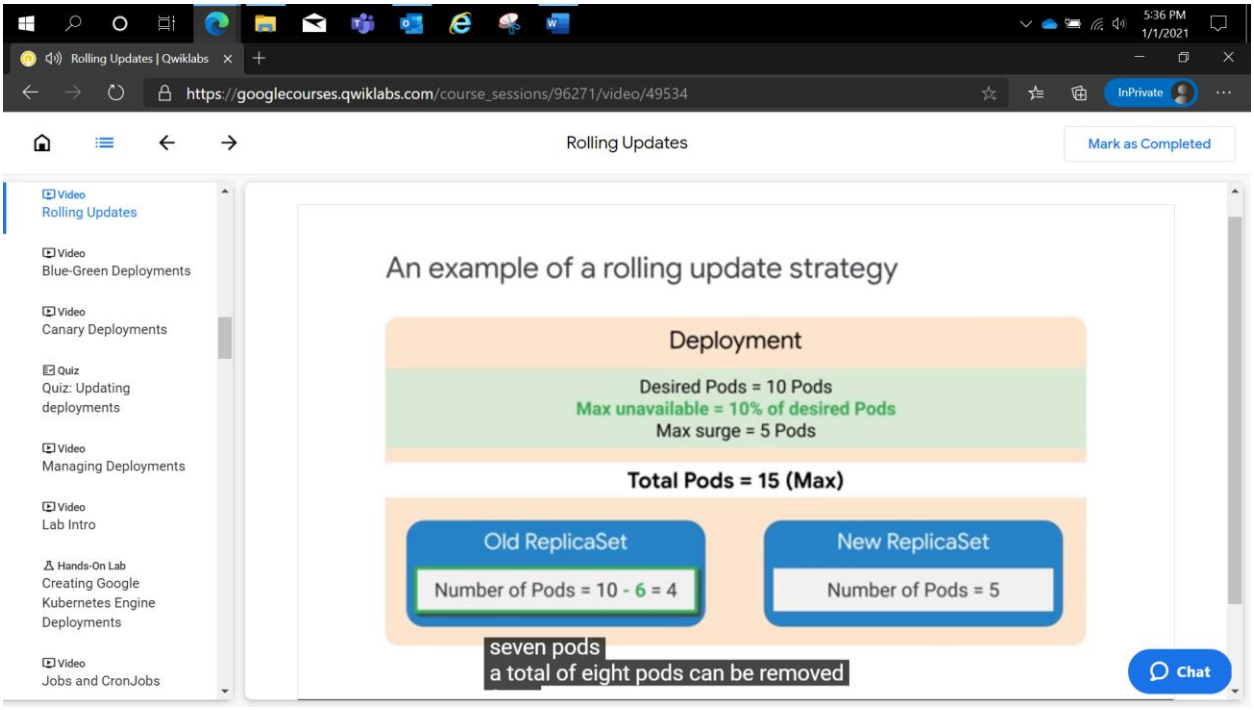

Rolling Updates | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49534
InPrivate
5:35 PM
1/1/2021
Rolling Updates
Mark as Completed
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Set parameters for your rolling update strategy
Deployment
Max unavailable = 25%
Old ReplicaSet
Pod
Pod
Pod
Pod
means you want to have at least 75 percent
Chat

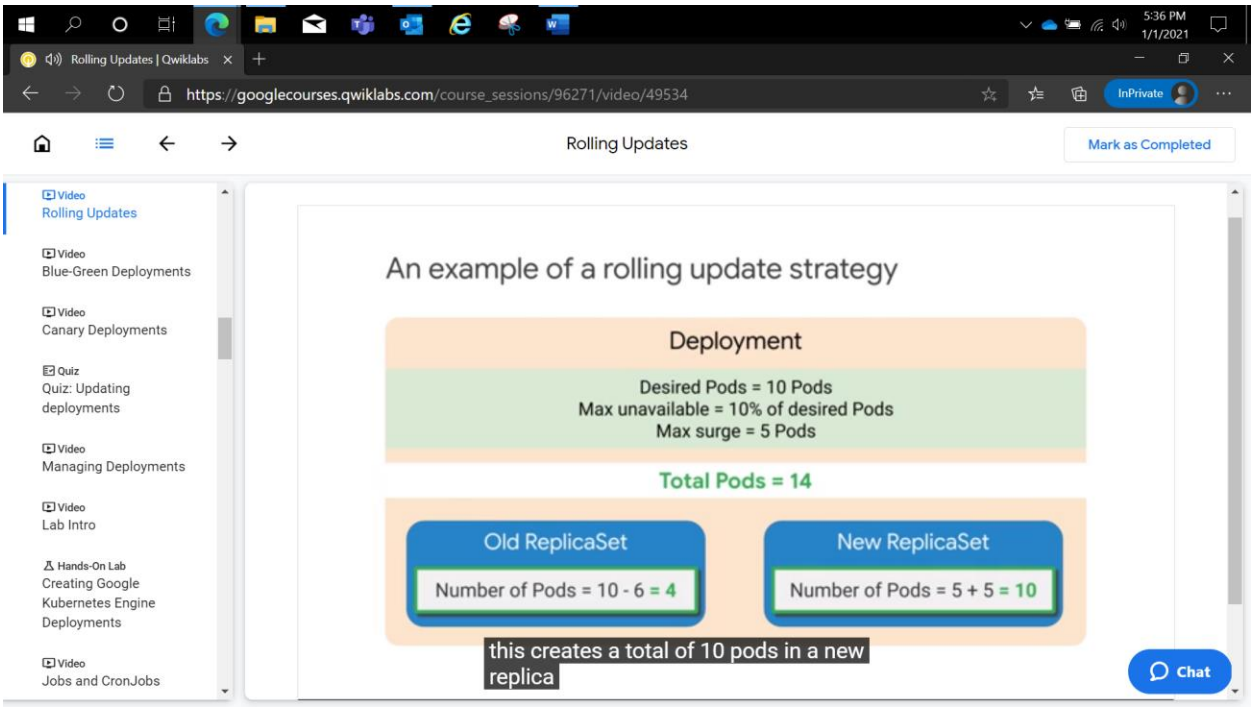

Rolling Updates | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49534
InPrivate
5:35 PM
1/1/2021
Rolling Updates
Mark as Completed
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Set parameters for your rolling update strategy
Deployment
Max surge = 2
New ReplicaSet
Pod
Pod
Settings
set
for example you can specify
Chat

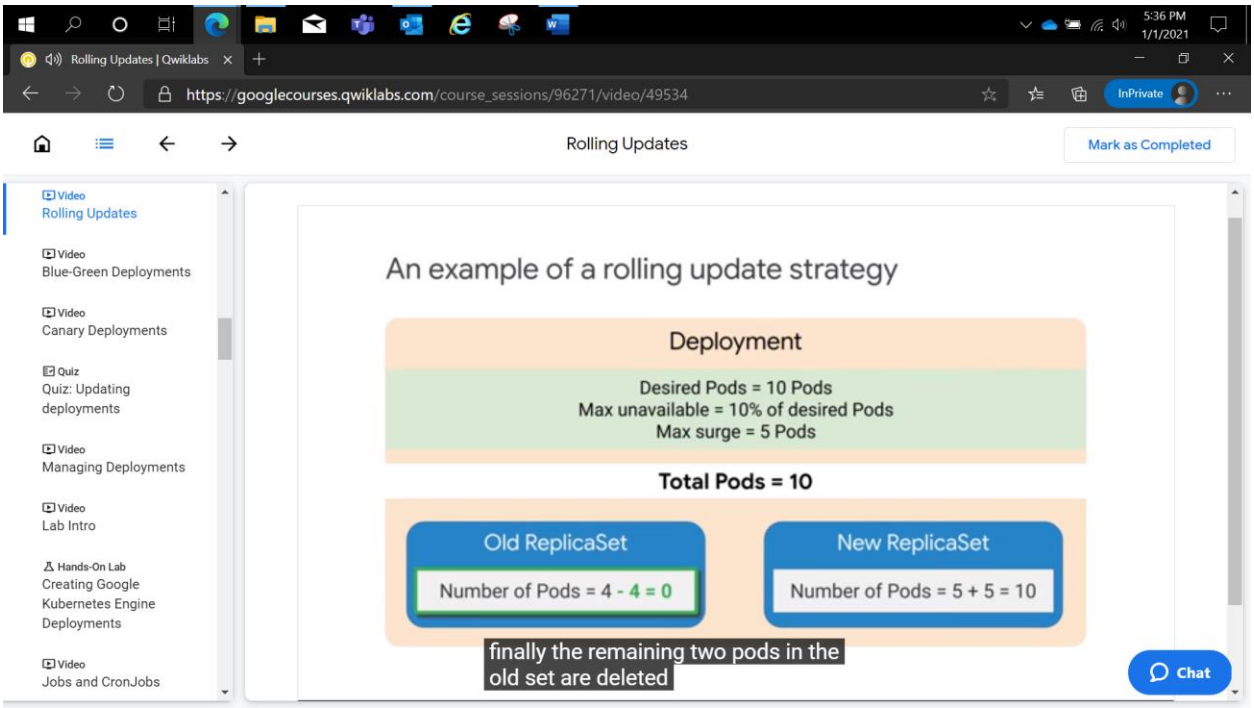

5:35 PM
1/1/2021
Rolling Updates | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49534
InPrivate
Rolling Updates
Mark as Completed
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Set parameters for your rolling update strategy
Deployment
Max surge = 25%
Old ReplicaSet
Pod
Pod
New ReplicaSet
Pod
Pod
Pod
the deployment controller looks at the total number
Settings
Chat

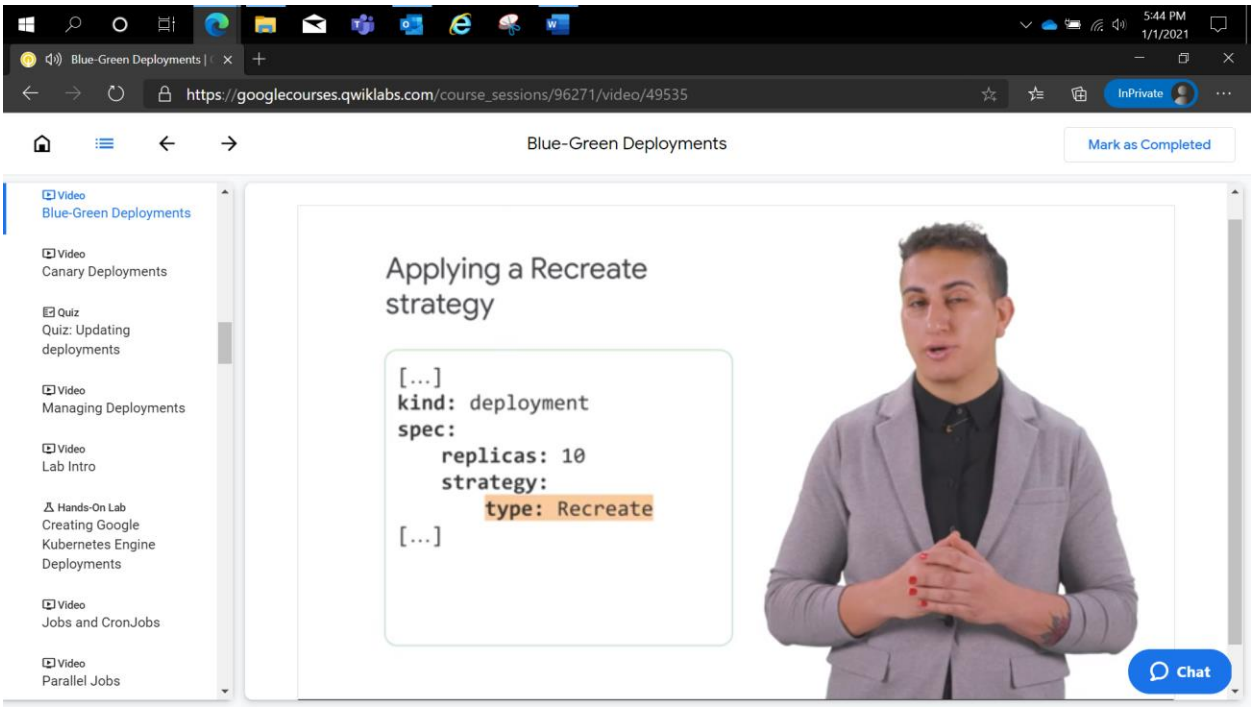

5:36 PM
1/1/2021
Rolling Updates | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49534
InPrivate
Rolling Updates
Mark as Completed
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
An example of a rolling update strategy
[...]
kind: deployment
spec:
replicas: 10
strategy:
type: RollingUpdate
rollingUpdate:
maxSurge: 5
maxUnavailable: 10%
[...]
desired number of pods set to 10
max unavailable set to 30 percent
Settings
Chat

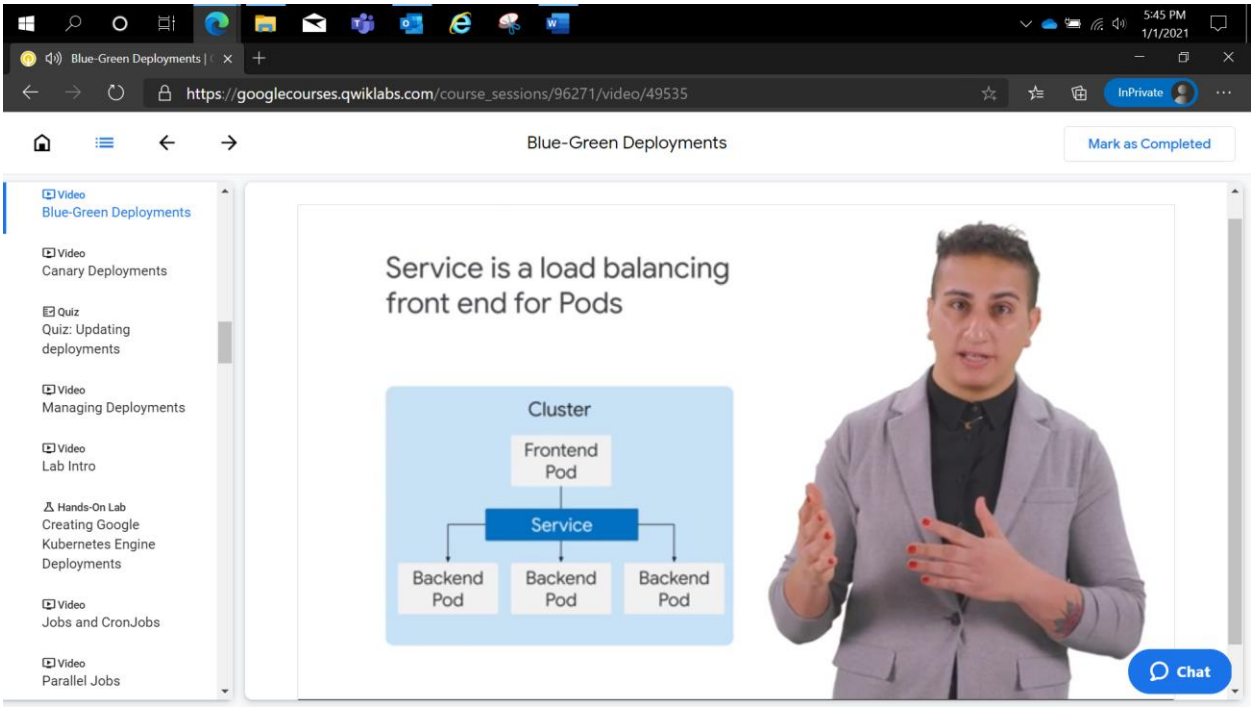

Rolling Updates | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49534
InPrivate
5:36 PM
1/1/2021
Rolling Updates
Mark as Completed
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
An example of a rolling update strategy
Deployment
Desired Pods = 10 Pods
Max unavailable = 10% of desired Pods
Max surge = 5 Pods
Total Pods = 10
Old ReplicaSet
Number of Pods = 10
New ReplicaSet
Number of Pods = 5
the old replica set has 10 pots the
deployment will begin by creating
Chat

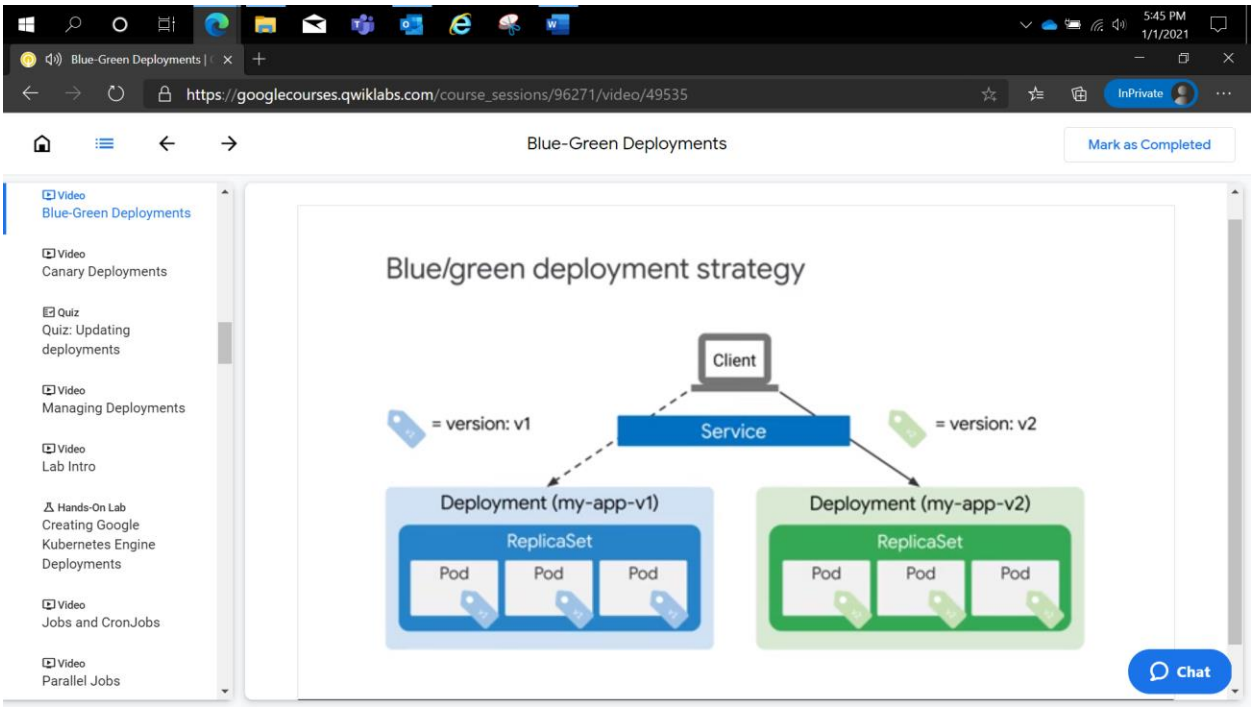

Rolling Updates | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49534
InPrivate
5:36 PM
1/1/2021
Rolling Updates
Mark as Completed
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
An example of a rolling update strategy
Deployment
Desired Pods = 10 Pods
Max unavailable = 10% of desired Pods
Max surge = 5 Pods
Total Pods = 15 (Max)
Old ReplicaSet
Number of Pods = 10 - 6 = 4
New ReplicaSet
Number of Pods = 5
seven pods
a total of eight pods can be removed
Chat

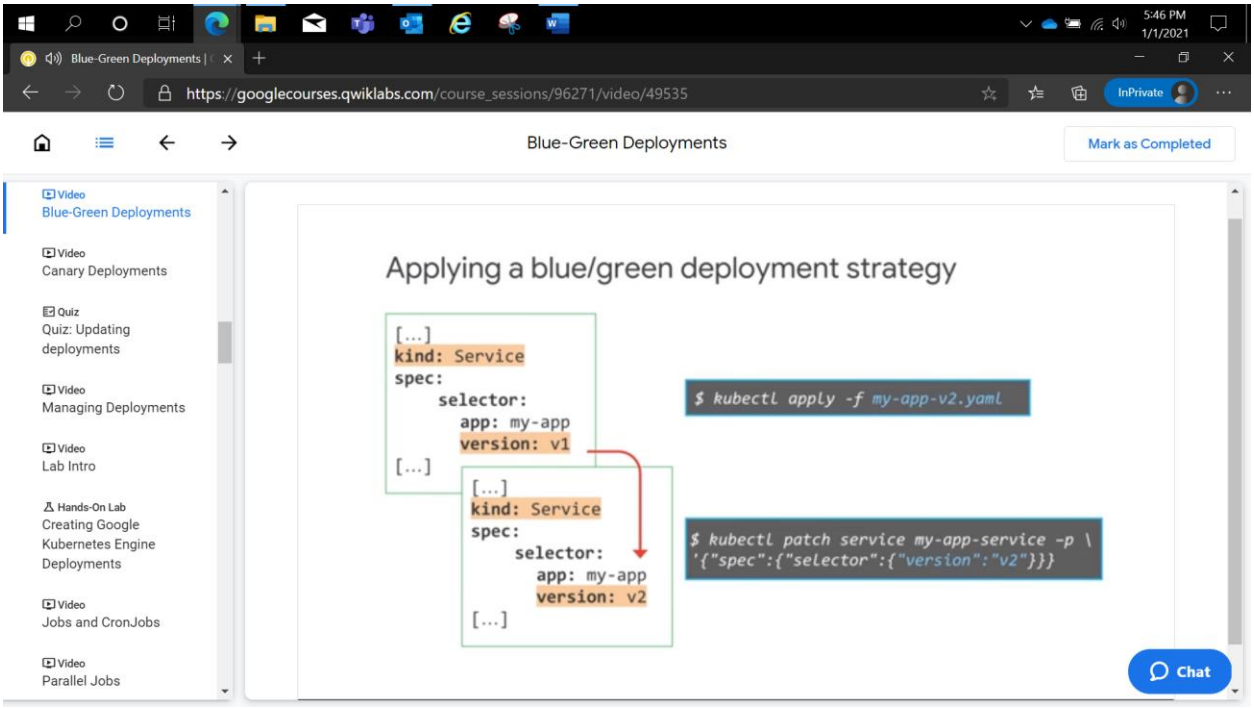

Rolling Updates | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49534
InPrivate
5:36 PM
1/1/2021
Rolling Updates
Mark as Completed
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
An example of a rolling update strategy
Deployment
Desired Pods = 10 Pods
Max unavailable = 10% of desired Pods
Max surge = 5 Pods
Total Pods = 14
Old ReplicaSet
Number of Pods = 10 - 6 = 4
New ReplicaSet
Number of Pods = 5 + 5 = 10
this creates a total of 10 pods in a new replica
Chat

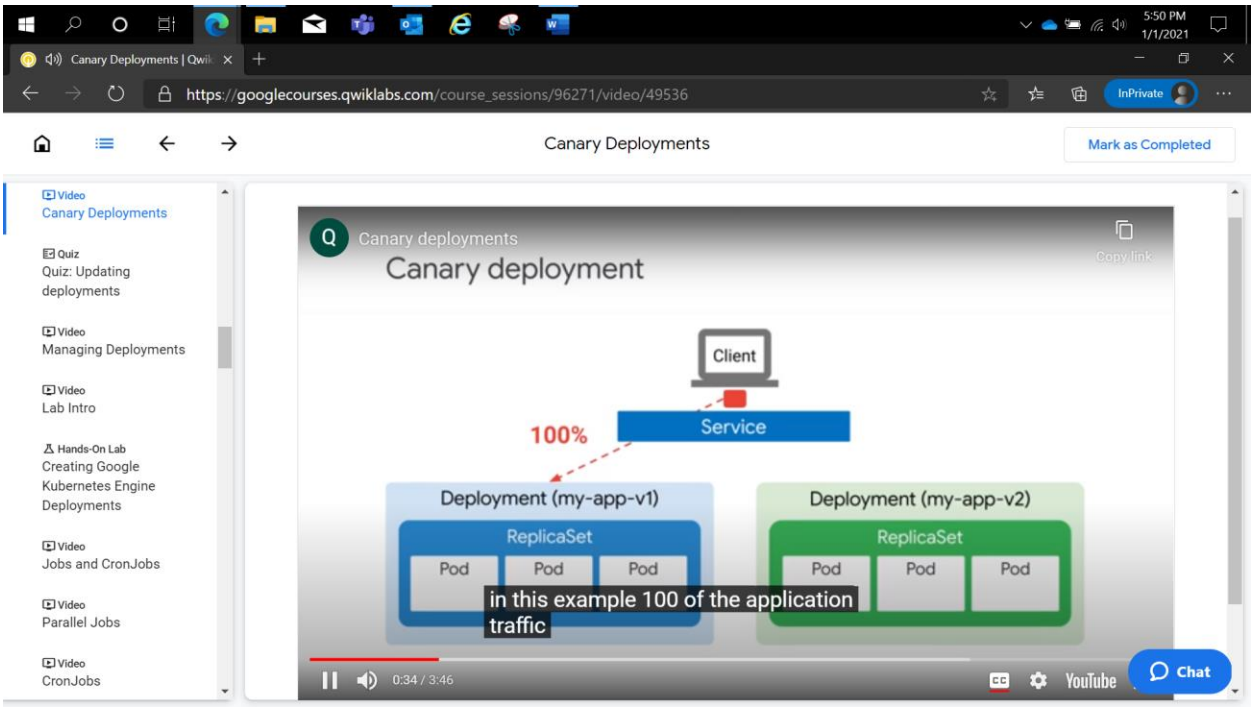

Rolling Updates | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49534
InPrivate
5:36 PM
1/1/2021
Rolling Updates
Mark as Completed
Video
Rolling Updates
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
An example of a rolling update strategy
Deployment
Desired Pods = 10 Pods
Max unavailable = 10% of desired Pods
Max surge = 5 Pods
Total Pods = 10
Old ReplicaSet
Number of Pods = 4 - 4 = 0
New ReplicaSet
Number of Pods = 5 + 5 = 10
finally the remaining two pods in the old set are deleted
Chat

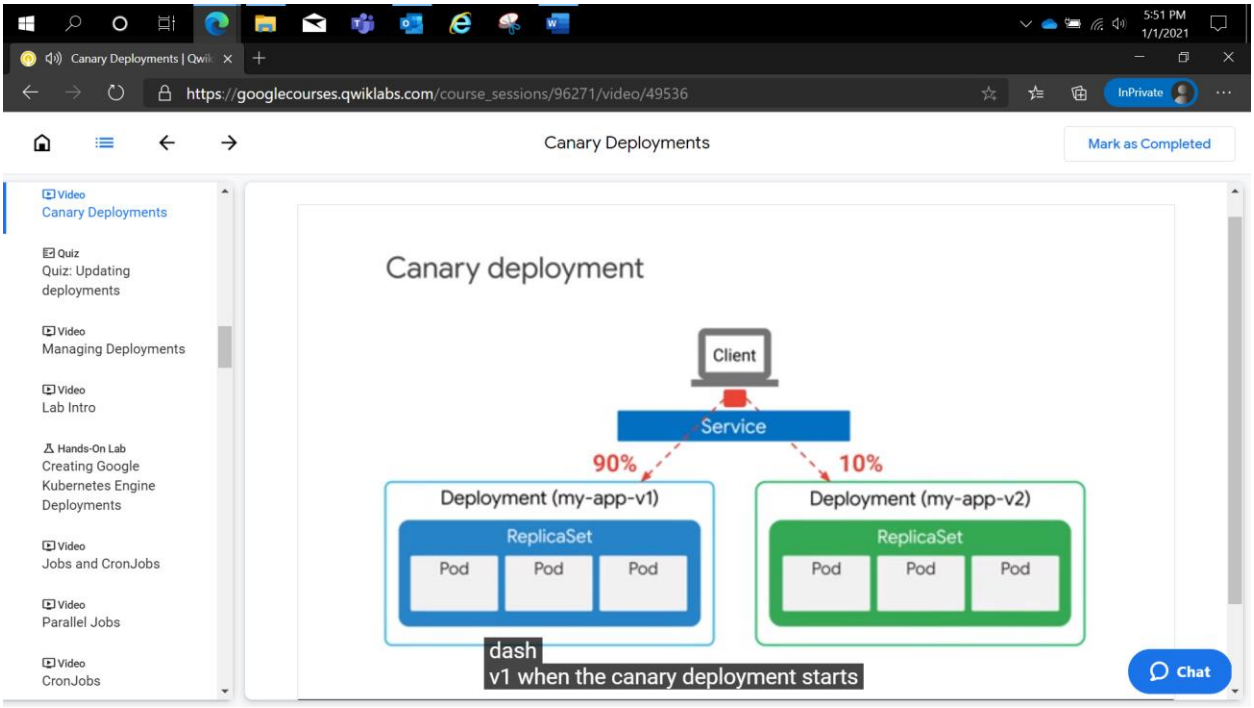

5:44 PM
1/1/2021
Blue-Green Deployments |
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49535
InPrivate
Blue-Green Deployments
Mark as Completed
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Video
Parallel Jobs
Applying a Recreate strategy
[...]
kind: deployment
spec:
replicas: 10
strategy:
type: Recreate
[...]
Chat

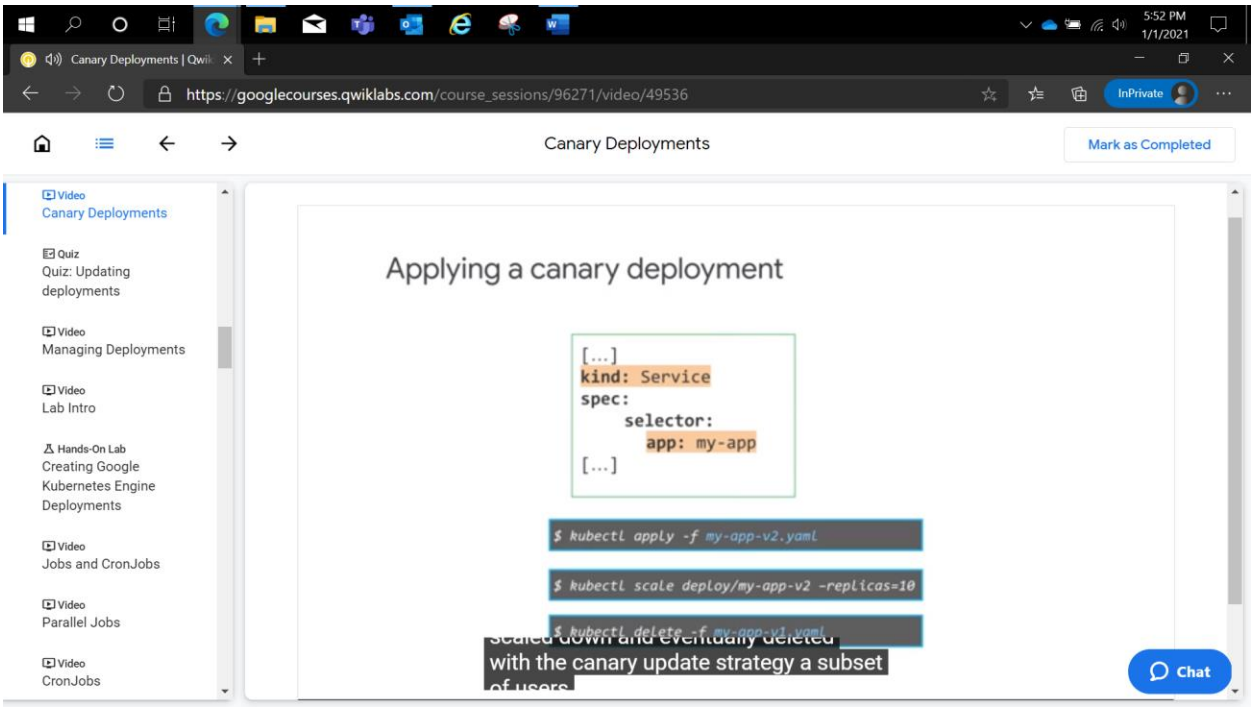

5:45 PM
1/1/2021
Blue-Green Deployments |
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49535
InPrivate
Blue-Green Deployments
Mark as Completed
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Video
Parallel Jobs
Service is a load balancing front end for Pods
Cluster
Frontend Pod
Service
Backend Pod
Backend Pod
Backend Pod
Chat

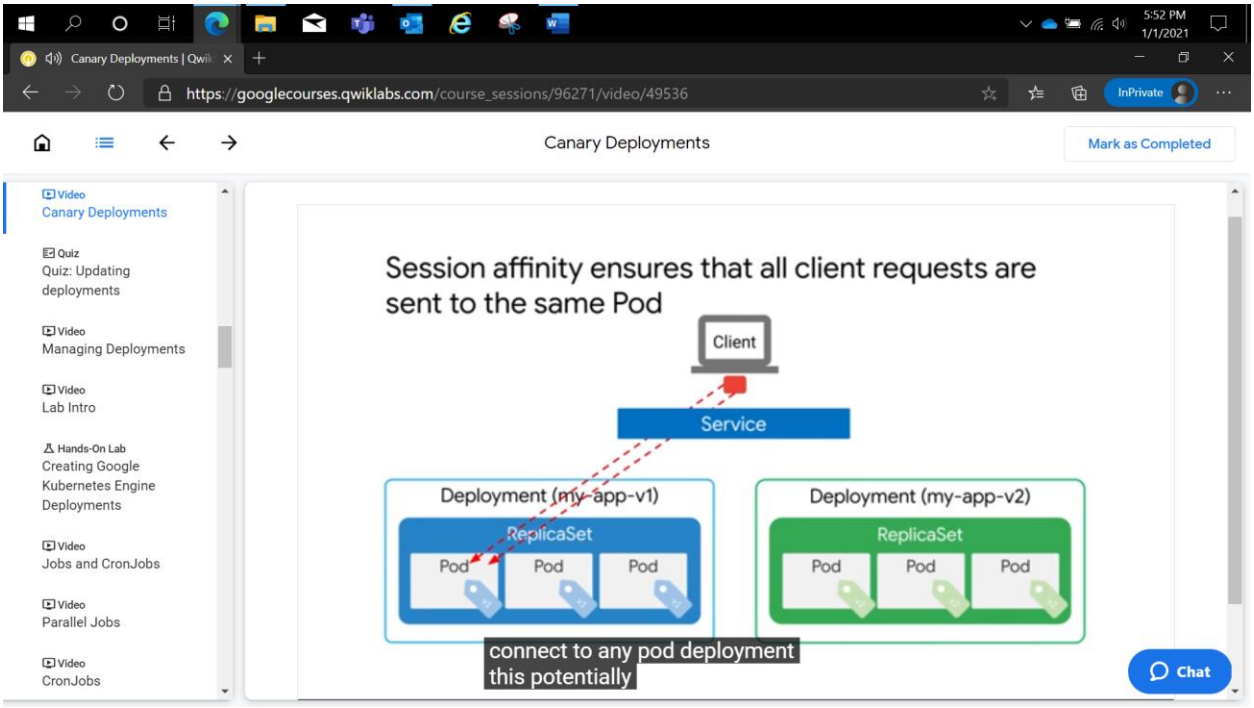

5:45 PM
1/1/2021
Blue-Green Deployments
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49535
InPrivate
Blue-Green Deployments
Mark as Completed
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Video
Parallel Jobs
Blue/green deployment strategy
Client
= version: v1
Service
= version: v2
Deployment (my-app-v1)
ReplicaSet
Pod
Pod
Pod
Deployment (my-app-v2)
ReplicaSet
Pod
Pod
Pod
Chat

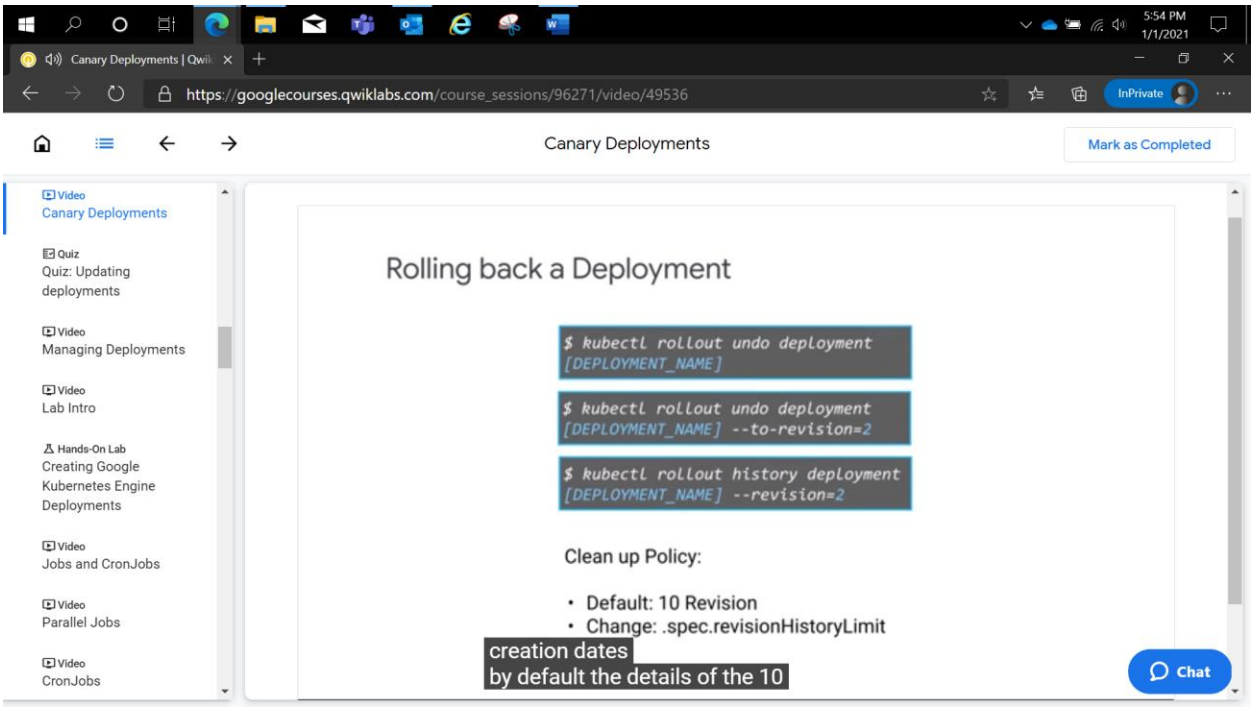

5:46 PM
1/1/2021
Blue-Green Deployments
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49535
InPrivate
Blue-Green Deployments
Mark as Completed
Video
Blue-Green Deployments
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Video
Parallel Jobs
Applying a blue/green deployment strategy
[...]
kind: Service
spec:
selector:
app: my-app
version: v1
[...]
$ kubectl apply -f my-app-v2.yaml
[...]
kind: Service
spec:
selector:
app: my-app
version: v2
[...]
$ kubectl patch service my-app-service -p \
'{"spec":{"selector":{"version":"v2"}}}'
Chat

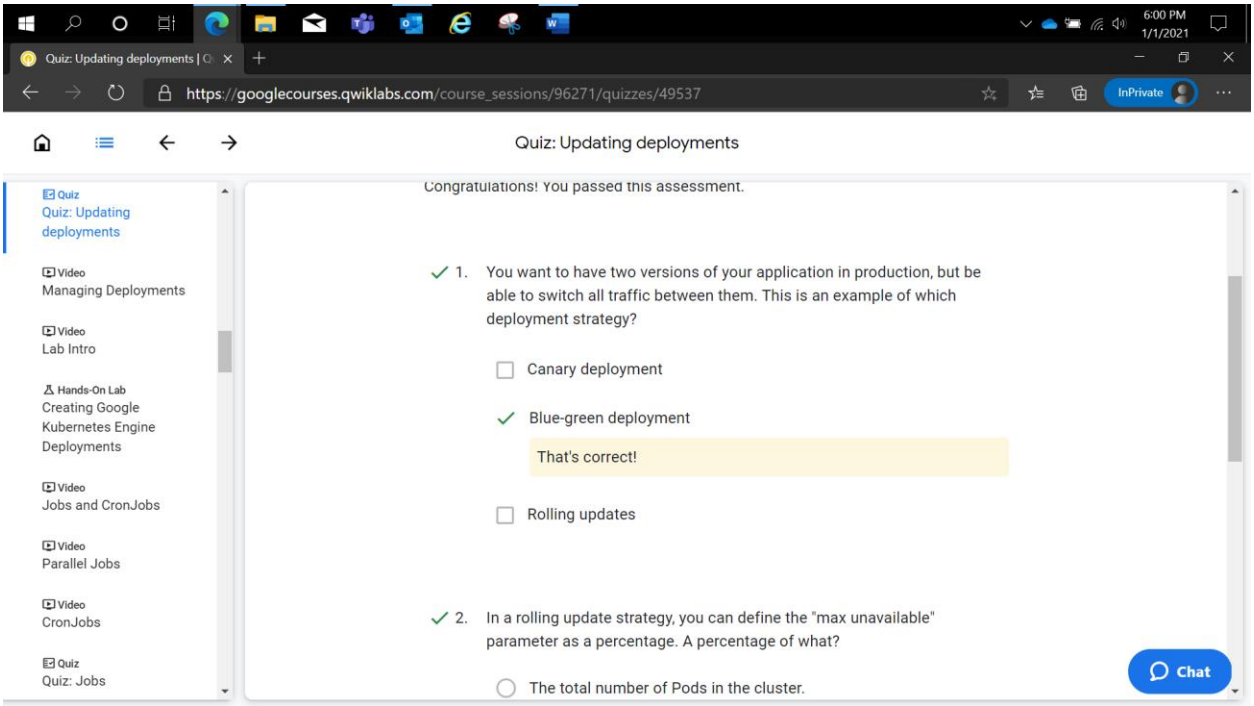

5:50 PM
1/1/2021
Canary Deployments | Qwik
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49536
InPrivate
Canary Deployments
Mark as Completed
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Video
Parallel Jobs
Video
CronJobs
Canary deployments
Canary deployment
Copy link
Client
Service
100%
Deployment (my-app-v1)
ReplicaSet
Pod
Pod
Pod
Deployment (my-app-v2)
ReplicaSet
Pod
Pod
Pod
in this example 100 of the application traffic
0:34 / 3:46
YouTube
Chat

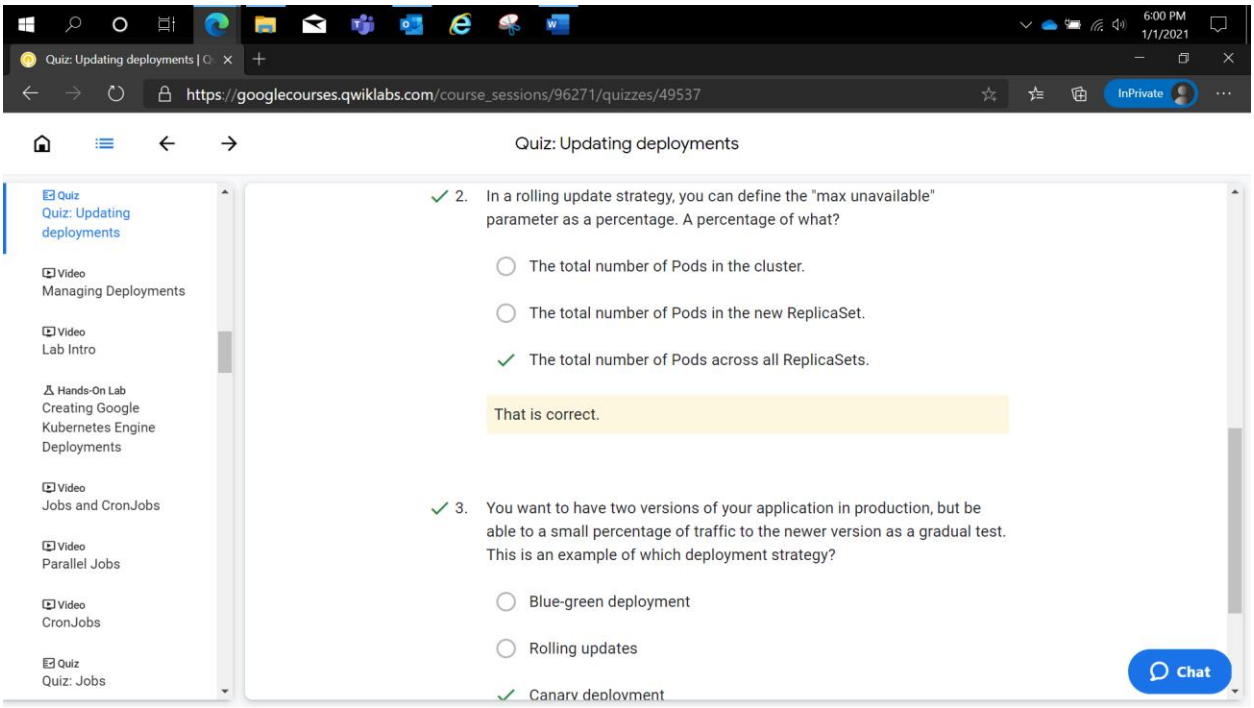

5:51 PM
1/1/2021
Canary Deployments | Qwik
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49536
InPrivate
Canary Deployments
Mark as Completed
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Video
Parallel Jobs
Video
CronJobs
Canary deployment
Client
Service
90%
10%
Deployment (my-app-v1)
ReplicaSet
Pod
Pod
Pod
Deployment (my-app-v2)
ReplicaSet
Pod
Pod
Pod
dash
v1 when the canary deployment starts
Chat

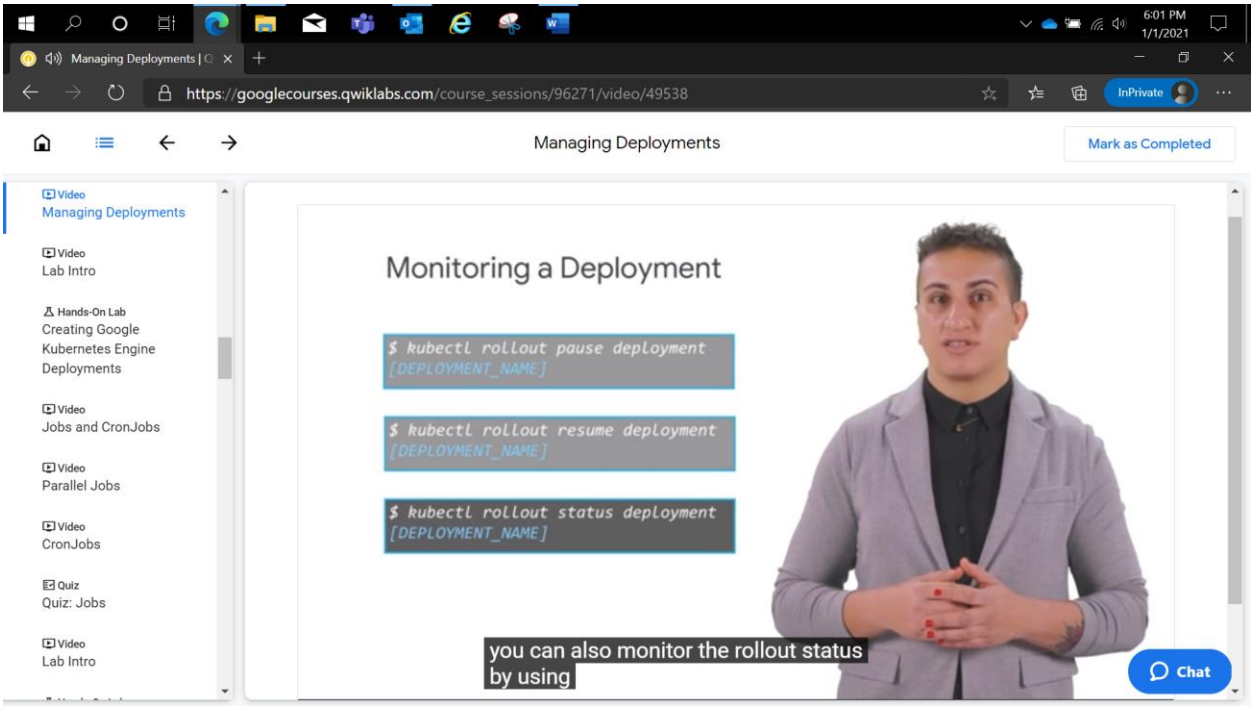

Canary Deployments | Qwik
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49536
InPrivate
5:52 PM
1/1/2021
Canary Deployments
Mark as Completed
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Video
Parallel Jobs
Video
CronJobs
Applying a canary deployment
[...]
kind: Service
spec:
selector:
app: my-app
[...]
$ kubectl apply -f my-app-v2.yaml
$ kubectl scale deploy/my-app-v2 -replicas=10
with the canary update strategy a subset
Chat

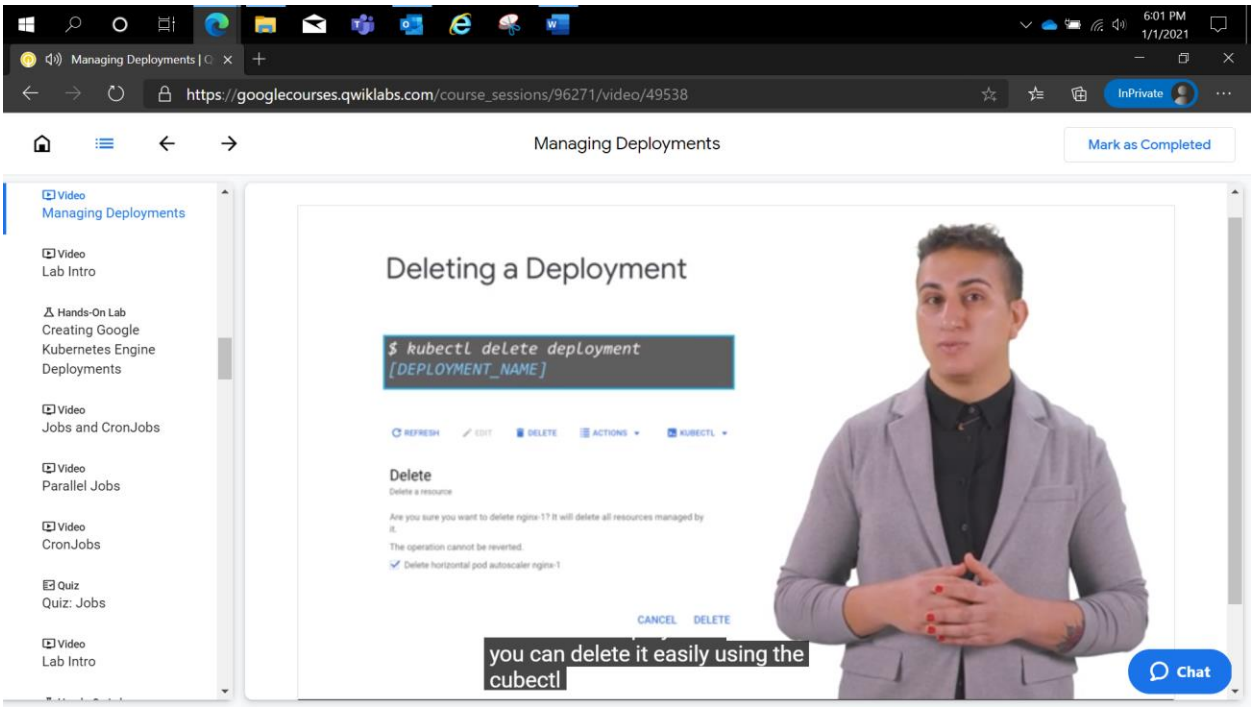

Canary Deployments | Qwik
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49536
InPrivate
5:52 PM
1/1/2021
Canary Deployments
Mark as Completed
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Video
Parallel Jobs
Video
CronJobs
Session affinity ensures that all client requests are sent to the same Pod
Client
Service
Deployment (my-app-v1)
ReplicaSet
Pod
Pod
Pod
Deployment (my-app-v2)
ReplicaSet
Pod
Pod
Pod
connect to any pod deployment
this potentially
Chat


Canary Deployments | Qwik
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49536
InPrivate
5:54 PM
1/1/2021
Canary Deployments
Mark as Completed
Video
Canary Deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Video
Parallel Jobs
Video
CronJobs
Rolling back a Deployment
$ kubectl rollout undo deployment [DEPLOYMENT_NAME]
$ kubectl rollout undo deployment [DEPLOYMENT_NAME] --to-revision=2
$ kubectl rollout history deployment [DEPLOYMENT_NAME] --revision=2
Clean up Policy:
• Default: 10 Revision
• Change: .spec.revisionHistoryLimit
creation dates
by default the details of the 10
Chat


Quiz: Updating deployments | Q
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49537
InPrivate
6:00 PM
1/1/2021
Quiz: Updating deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Congratulations! You passed this assessment.
1. You want to have two versions of your application in production, but be able to switch all traffic between them. This is an example of which deployment strategy?
Canary deployment
Blue-green deployment
That's correct!
Rolling updates
2. In a rolling update strategy, you can define the "max unavailable" parameter as a percentage. A percentage of what?
The total number of Pods in the cluster.
Chat


Quiz: Updating deployments | C
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49537
InPrivate
6:00 PM
1/1/2021
Quiz: Updating deployments
Quiz
Quiz: Updating deployments
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
2. In a rolling update strategy, you can define the "max unavailable" parameter as a percentage. A percentage of what?
The total number of Pods in the cluster.
The total number of Pods in the new ReplicaSet.
The total number of Pods across all ReplicaSets.
That is correct.
3. You want to have two versions of your application in production, but be able to a small percentage of traffic to the newer version as a gradual test. This is an example of which deployment strategy?
Blue-green deployment
Rolling updates
Canary deployment
Chat


Managing Deployments | C
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49538
InPrivate
6:01 PM
1/1/2021
Managing Deployments
Mark as Completed
Video
Managing Deployments
Video
Lab Intro
Hands-On Lab
Creating Google Kubernetes Engine Deployments
Video
Jobs and CronJobs
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Monitoring a Deployment
$ kubectl rollout pause deployment [DEPLOYMENT_NAME]
$ kubectl rollout resume deployment [DEPLOYMENT_NAME]
$ kubectl rollout status deployment [DEPLOYMENT_NAME]
you can also monitor the rollout status by using
Chat



Creating Google Kubernetes Engine Deployments

1 hourFree

Rate Lab

Overview

In this lab, you explore the basics of using deployment manifests. Manifests are files that contain configurations required for a deployment that can be used across different Pods. Manifests are easy to change.

Objectives

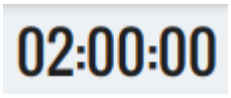
In this lab, you learn how to perform the following tasks:

- Create deployment manifests, deploy to cluster, and verify Pod rescheduling as nodes are disabled
- Trigger manual scaling up and down of Pods in deployments
- Trigger deployment rollout (rolling update to new version) and rollbacks
- Perform a Canary deployment

Task 0. Lab Setup

Access Qwiklabs

For each lab, you get a new GCP project and set of resources for a fixed time at no cost.

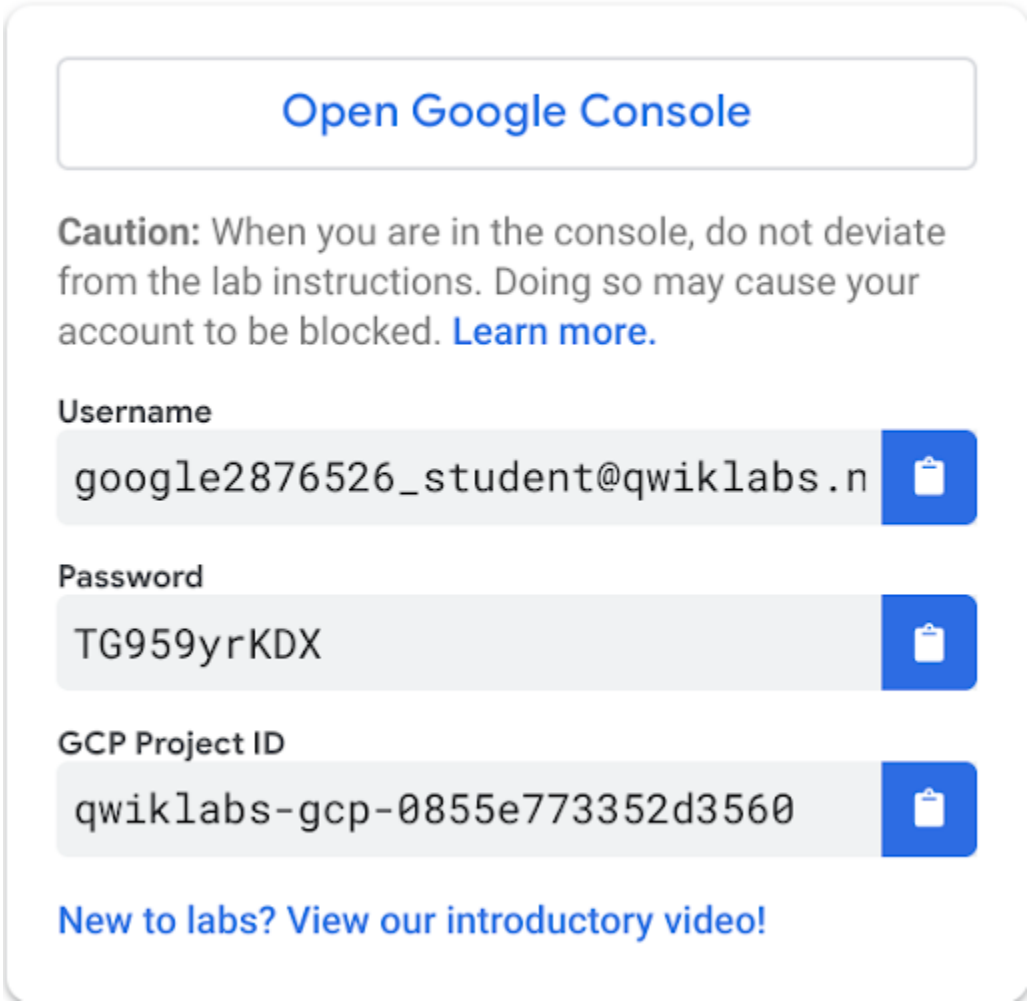
1. Make sure you signed into Qwiklabs using an **incognito window**.
2. Note the lab's access time (for example, 

and make sure you can finish in that time block.

There is no pause feature. You can restart if needed, but you have to start at the beginning.

3. When ready, click 

.
4. Note your lab credentials. You will use them to sign in to Cloud Platform Console.



5. Click **Open Google Console**.

6. Click **Use another account** and copy/paste credentials for **this** lab into the prompts.

If you use other credentials, you'll get errors or **incur charges**.

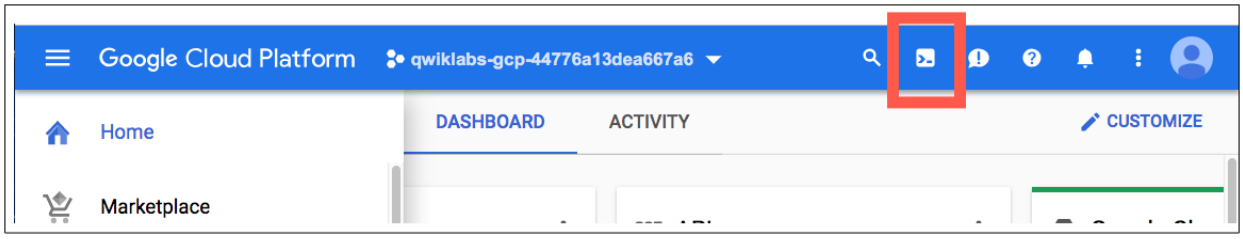
7. Accept the terms and skip the recovery resource page.

Do not click **End Lab** unless you are finished with the lab or want to restart it. This clears your work and removes the project.
After you complete the initial sign-in steps, the project dashboard appears.

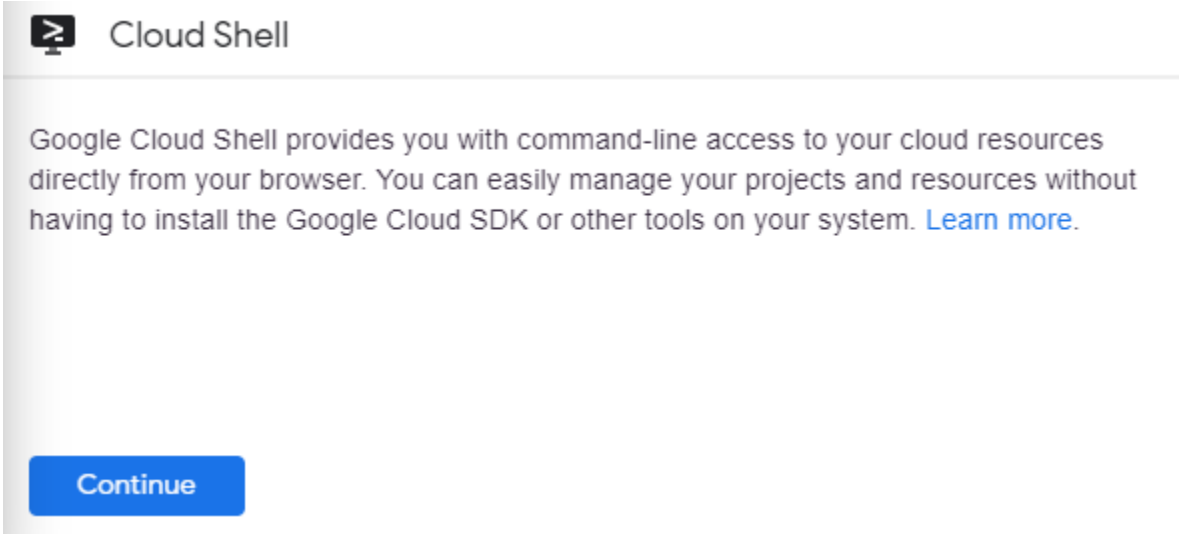
Activate Google Cloud Shell

Google Cloud Shell is a virtual machine that is loaded with development tools. It offers a persistent 5GB home directory and runs on the Google Cloud. Google Cloud Shell provides command-line access to your GCP resources.

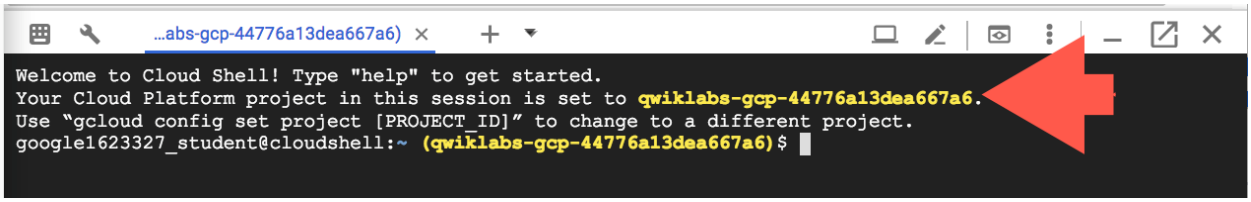
1. In GCP console, on the top right toolbar, click the Open Cloud Shell button.



2. Click **Continue**.



It takes a few moments to provision and connect to the environment. When you are connected, you are already authenticated, and the project is set to your *PROJECT_ID*. For example:



gcloud is the command-line tool for Google Cloud Platform. It comes pre-installed on Cloud Shell and supports tab-completion.

You can list the active account name with this command:

```
gcloud auth list
```
content_copy

Output:

```
Credentialed accounts:
 - <myaccount>@<mydomain>.com (active)
```
content_copy

Example output:

```
Credentialed accounts:
 - google1623327_student@qwiklabs.net
```
content_copy

You can list the project ID with this command:

```
gcloud config list project
```

```
content_copy
```

Output:

```
[core]
project = <project_ID>content_copy
```

Example output:

```
[core]
project = qwiklabs-gcp-44776a13dea667a6content_copy
```

Full documentation of **gcloud** is available on Google Cloud gcloud Overview .

Task 1. Create deployment manifests and deploy to the cluster

In this task, you create a deployment manifest for a Pod inside the cluster.

Connect to the lab GKE cluster

1. In Cloud Shell, type the following command to set the environment variable for the zone and cluster name.

```
export my_zone=us-central1-a
export my_cluster=standard-cluster-1
content_copy
```

2. Configure kubectl tab completion in Cloud Shell.

```
source <(kubectl completion bash)
content_copy
```

3. In Cloud Shell, configure access to your cluster for the kubectl command-line tool, using the following command:

```
gcloud container clusters get-credentials $my_cluster --zone $my_zone
content_copy
```

4. In Cloud Shell enter the following command to clone the repository to the lab Cloud Shell.

```
git clone https://github.com/GoogleCloudPlatform/training-data-analyst
```
content_copy

5. Create a soft link as a shortcut to the working directory.

```
ln -s ~/training-data-analyst/courses/ak8s/v1.1 ~/ak8s
```
content_copy

6. Change to the directory that contains the sample files for this lab.

```
cd ~/ak8s/Deployments/
```
content_copy

Create a deployment manifest

You will create a deployment using a sample deployment manifest called `nginx-deployment.yaml` that has been provided for you. This deployment is configured to run three Pod replicas with a single nginx container in each Pod listening on TCP port 80.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:1.7.9
        ports:
        - containerPort: 80
```
content_copy

1. To deploy your manifest, execute the following command:

```
kubectl apply -f ./nginx-deployment.yaml
```

```
content_copy
```

Click *Check my progress* to verify the objective.

Create and deploy manifest nginx deployment

Check my progress

2. To view a list of deployments, execute the following command:

```
kubectl get deployments
content_copy
```

The output should look like this example.

Output (do not copy)

```
NAME               READY      UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   0/3        3            0           3s
content_copy
```

3. Wait a few seconds, and repeat the command until the number listed for CURRENT deployments reported by the command matches the number of DESIRED deployments.

The final output should look like the example.

Output (do not copy)

```
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   3/3     3            3           42s
content_copy
```

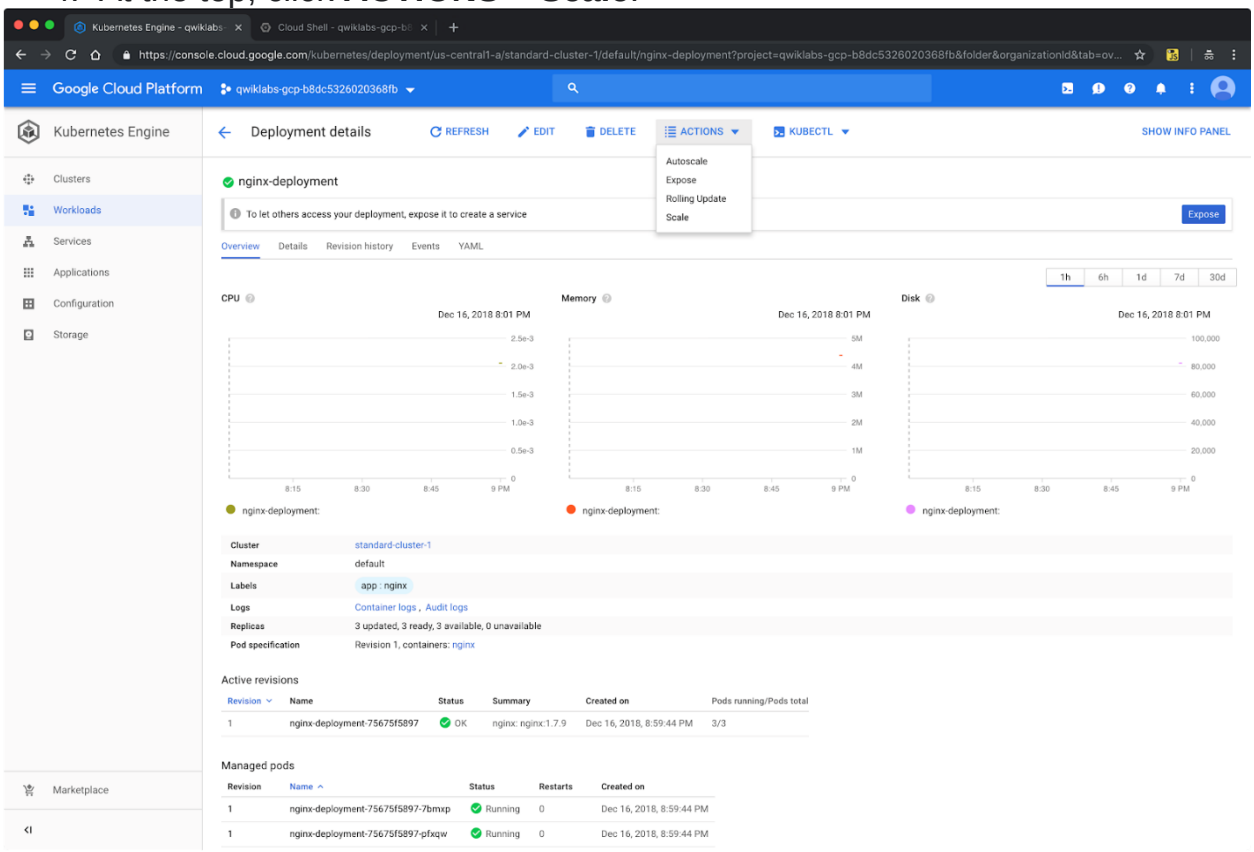
Task 2. Manually scale up and down the number of Pods in deployments

Sometimes, you want to shut down a Pod instance. Other times, you want ten Pods running. In Kubernetes, you can scale a specific Pod to the desired number of instances. To shut them down, you scale to zero.

In this task, you scale Pods up and down in the Google Cloud Console and Cloud Shell.

Scale Pods up and down in the console

1. Switch to the Google Cloud Console tab.
2. On the **Navigation menu** ( ), click **Kubernetes Engine** > **Workloads**.
3. Click **nginx-deployment** (your deployment) to open the Deployment details page.
4. At the top, click **ACTIONS > Scale**.



5. Type **1** and click **SCALE**.

This action scales down your cluster. You should see the Pod status being updated under **Managed Pods**. You might have to click **Refresh**.

Scale Pods up and down in the shell

1. Switch back to the Cloud Shell browser tab.

2. In the Cloud Shell, to view a list of Pods in the deployments, execute the following command:

```
kubectl get deployments
```
content_copy

Output (do not copy)

```
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   1/1     1            1           3m
```
content_copy

3. To scale the Pod back up to three replicas, execute the following command:

```
kubectl scale --replicas=3 deployment nginx-deployment
```
content_copy

4. To view a list of Pods in the deployments, execute the following command:

```
kubectl get deployments
```
content_copy

Output (do not copy)

```
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   3/3     3            3           4m
```
content_copy

Task 3. Trigger a deployment rollout and a deployment rollback

A deployment's rollout is triggered if and only if the deployment's Pod template (that is, `.spec.template`) is changed, for example, if the labels or container

images of the template are updated. Other updates, such as scaling the deployment, do not trigger a rollout.

In this task, you trigger deployment rollout, and then you trigger deployment rollback.

Trigger a deployment rollout

1. To update the version of nginx in the deployment, execute the following command:

```
kubectl set image deployment.v1.apps/nginx-deployment nginx=nginx:1.9.1 --
record
```
content_copy

This updates the container image in your Deployment to nginx v1.9.1.

Click *Check my progress* to verify the objective.

Update version of nginx in the deployment

Check my progress

2. To view the rollout status, execute the following command:

```
kubectl rollout status deployment.v1.apps/nginx-deployment
```
content_copy

The output should look like the example.

Output (do not copy)

```
Waiting for rollout to finish: 1 out of 3 new replicas updated...
Waiting for rollout to finish: 1 out of 3 new replicas updated...
Waiting for rollout to finish: 1 out of 3 new replicas updated...
Waiting for rollout to finish: 2 out of 3 new replicas updated...
Waiting for rollout to finish: 2 out of 3 new replicas updated...
Waiting for rollout to finish: 2 out of 3 new replicas updated...
Waiting for rollout to finish: 1 old replicas pending termination...
Waiting for rollout to finish: 1 old replicas pending termination...
deployment "nginx-deployment" successfully rolled out
```
content_copy

3. To verify the change, get the list of deployments.

```
kubectl get deployments
```
content_copy

The output should look like the example.

Output (do not copy)

```
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   3/3     3            3           6m
content_copy
```

4. View the rollout history of the deployment.

```
kubectl rollout history deployment nginx-deployment
content_copy
```

The output should look like the example. Your output might not be an exact match.

Output (do not copy)

```
deployments "nginx-deployment"
REVISION  CHANGE-CAUSE
1         <none>
2         kubectl set image deployment.v1.apps/nginx-deployment
nginx=nginx:1.9.1 --record=true
content_copy
```

Trigger a deployment rollback

To roll back an object's rollout, you can use the `kubectl rollout undo` command.

1. To roll back to the previous version of the nginx deployment, execute the following command:

```
kubectl rollout undo deployments nginx-deployment
content_copy
```

2. View the updated rollout history of the deployment.

```
kubectl rollout history deployment nginx-deployment
content_copy
```

The output should look like the example. Your output might not be an exact match.

Output (do not copy)

```
deployments "nginx-deployment"
REVISION  CHANGE-CAUSE
2         kubectl set image deployment.v1.apps/nginx-deployment
nginx=nginx:1.9.1 --record=true
3         <none>
```
content_copy

3. View the details of the latest deployment revision

```
kubectl rollout history deployment/nginx-deployment --revision=3
```
content_copy

The output should look like the example. Your output might not be an exact match but it will show that the current revision has rolled back to `nginx:1.7.9`.

Output (do not copy)

```
deployments "nginx-deployment" with revision #3
Pod Template:
  Labels:       app=nginx
        pod-template-hash=3123191453
  Containers:
   nginx:
    Image:      nginx:1.7.9
    Port:       80/TCP
    Host Port:  0/TCP
    Environment:        <none>
    Mounts:     <none>
  Volumes:      <none>
```
content_copy

Task 4. Define the service type in the manifest

In this task, you create and verify a service that controls inbound traffic to an application. Services can be configured as ClusterIP, NodePort or LoadBalancer types. In this lab you configure a LoadBalancer.

Define service types in the manifest

A manifest file called `service-nginx.yaml` that deploys a LoadBalancer service type has been provided for you. This service is configured to distribute inbound traffic on TCP port 60000 to port 80 on any containers that have the label `app: nginx`.

```
apiVersion: v1
kind: Service
metadata:
  name: nginx
spec:
  type: LoadBalancer
  selector:
    app: nginx
  ports:
  - protocol: TCP
    port: 60000
    targetPort: 80
```
content_copy

1. In the Cloud Shell, to deploy your manifest, execute the following command:

```
kubectl apply -f ./service-nginx.yaml
```
content_copy

This manifest defines a service and applies it to Pods that correspond to the selector. In this case, the manifest is applied to the nginx container that you deployed in task 1. This service also applies to any other Pods with the `app: nginx` label, including any that are created after the service.

Click *Check my progress* to verify the objective.

Deploy manifest file that deploys LoadBalancer service type

Check my progress

Verify the LoadBalancer creation

1. To view the details of the nginx service, execute the following command:

```
kubectl get service nginx
```
content_copy

The output should look like the example.

Output (do not copy)

```
NAME       CLUSTER_IP      EXTERNAL_IP      PORT(S)    SELECTOR    AGE
nginx      10.X.X.X        X.X.X.X          60000/TCP    run=nginx    1m
content_copy
```

2. When the external IP appears, open `http://[EXTERNAL_IP]:60000/` in a new browser tab to see the server being served through network load balancing.

It may take a few seconds before the **ExternalIP** field is populated for your service. This is normal. Just re-run the `kubectl get services nginx` command every few seconds until the field is populated.

Task 5. Perform a canary deployment

A canary deployment is a separate deployment used to test a new version of your application. A single service targets both the canary and the normal deployments. And it can direct a subset of users to the canary version to mitigate the risk of new releases. The manifest file `nginx-canary.yaml` that is provided for you deploys a single pod running a newer version of nginx than your main deployment. In this task, you create a canary deployment using this new deployment file.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-canary
  labels:
    app: nginx
spec:
  replicas: 1
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
        track: canary
        Version: 1.9.1
```

```
    spec:
      containers:
      - name: nginx
        image: nginx:1.9.1
        ports:
        - containerPort: 80
content_copy
```

The manifest for the nginx Service you deployed in the previous task uses a label selector to target the Pods with the `app: nginx` label. Both the normal deployment and this new canary deployment have the `app: nginx` label. Inbound connections will be distributed by the service to both the normal and canary deployment Pods. The canary deployment has fewer replicas (Pods) than the normal deployment, and thus it is available to fewer users than the normal deployment.

1. Create the canary deployment based on the configuration file.

```
kubectl apply -f nginx-canary.yaml
content_copy
```

Click *Check my progress* to verify the objective.

Create a Canary Deployment

Check my progress

2. When the deployment is complete, verify that both the nginx and the nginx-canary deployments are present.

```
kubectl get deployments
content_copy
```

3. Switch back to the browser tab that is connected to the external LoadBalancer service ip and refresh the page. You should continue to see the standard "Welcome to nginx" page.

4. Switch back to the Cloud Shell and scale down the primary deployment to 0 replicas.

```
kubectl scale --replicas=0 deployment nginx-deployment
content_copy
```

5. Verify that the only running replica is now the Canary deployment:

```
kubectl get deployments
content_copy
```

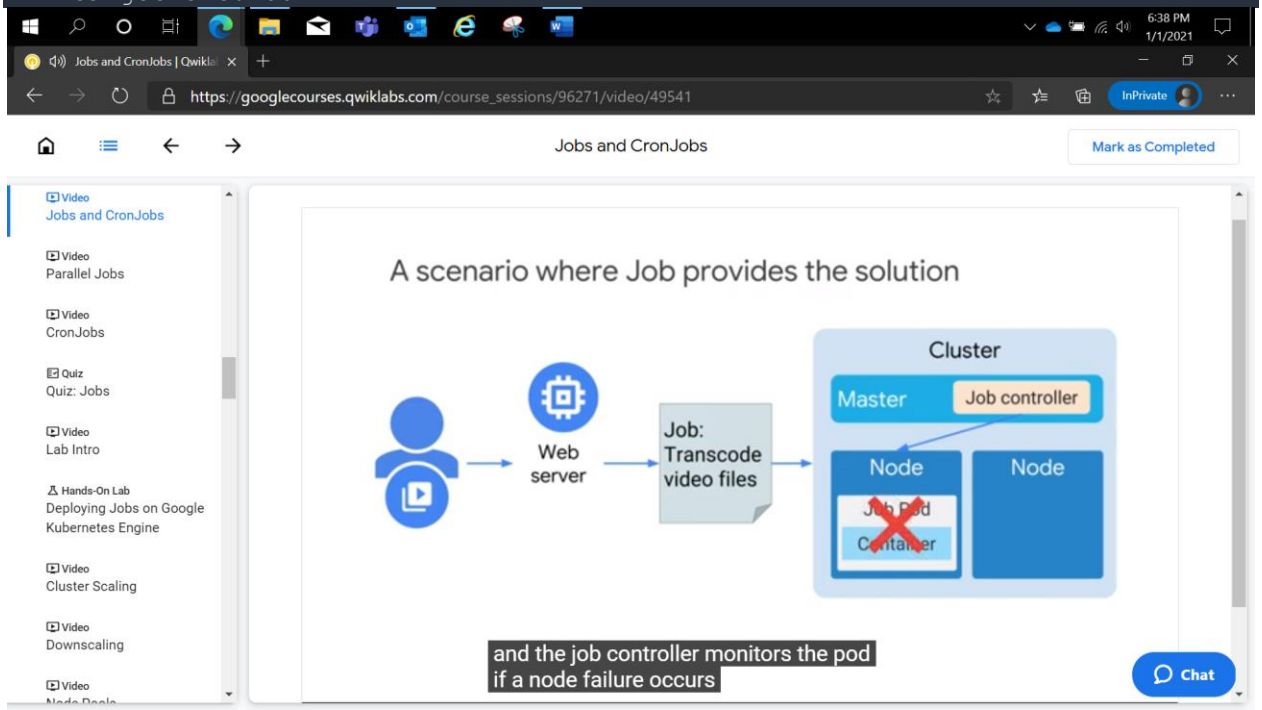
6. Switch back to the browser tab that is connected to the external LoadBalancer service ip and refresh the page. You should continue to see the standard "Welcome to nginx" page showing that the Service is automatically balancing traffic to the canary deployment.

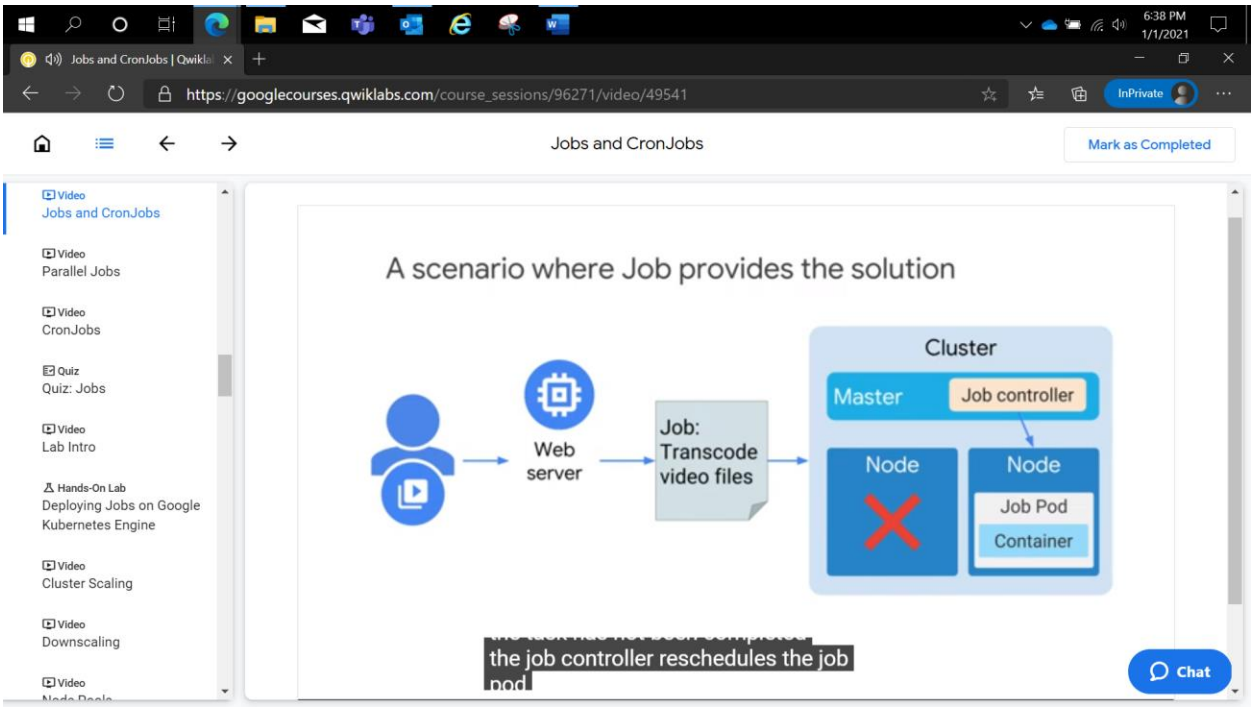
Note: Session affinity

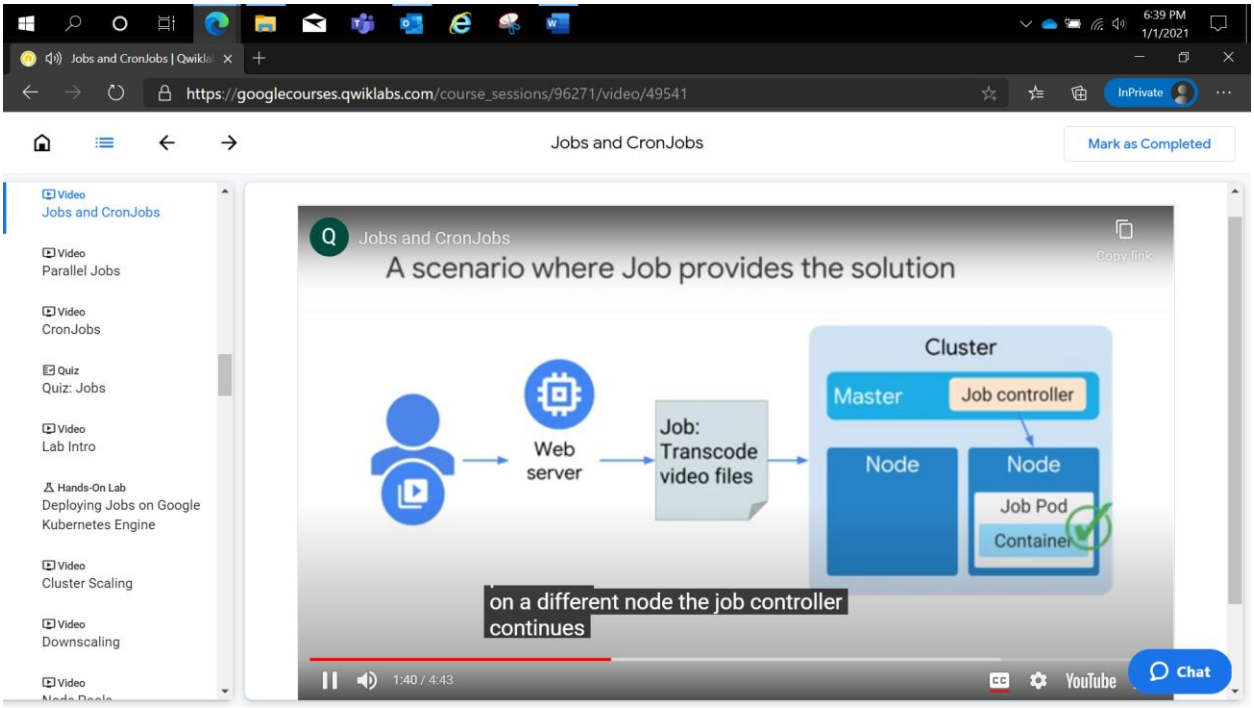
The Service configuration used in the lab does not ensure that all requests from a single client will always connect to the same Pod. Each request is treated separately and can connect to either the normal nginx deployment or to the nginx-canary deployment. This potential to switch between different versions may cause problems if there are significant changes in functionality in the canary release. To prevent this you can set the `sessionAffinity` field to `ClientIP` in the specification of the service if you need a client's first request to determine which Pod will be used for all subsequent connections.

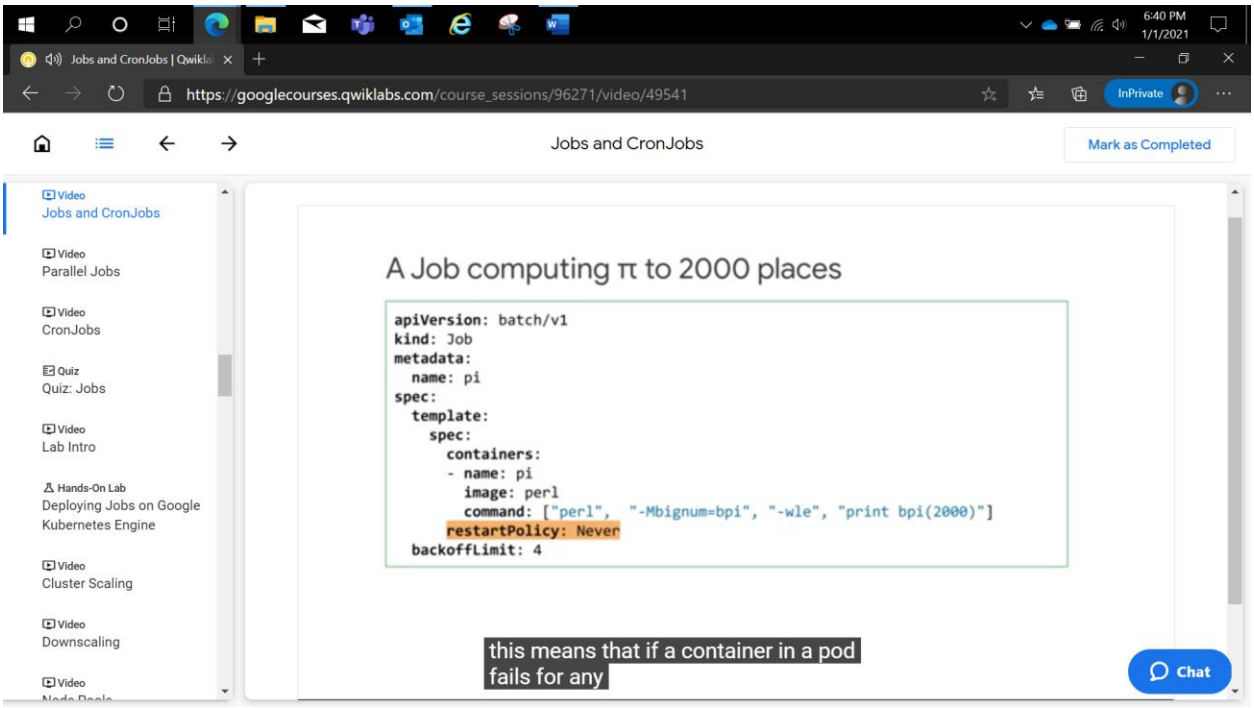
For example:

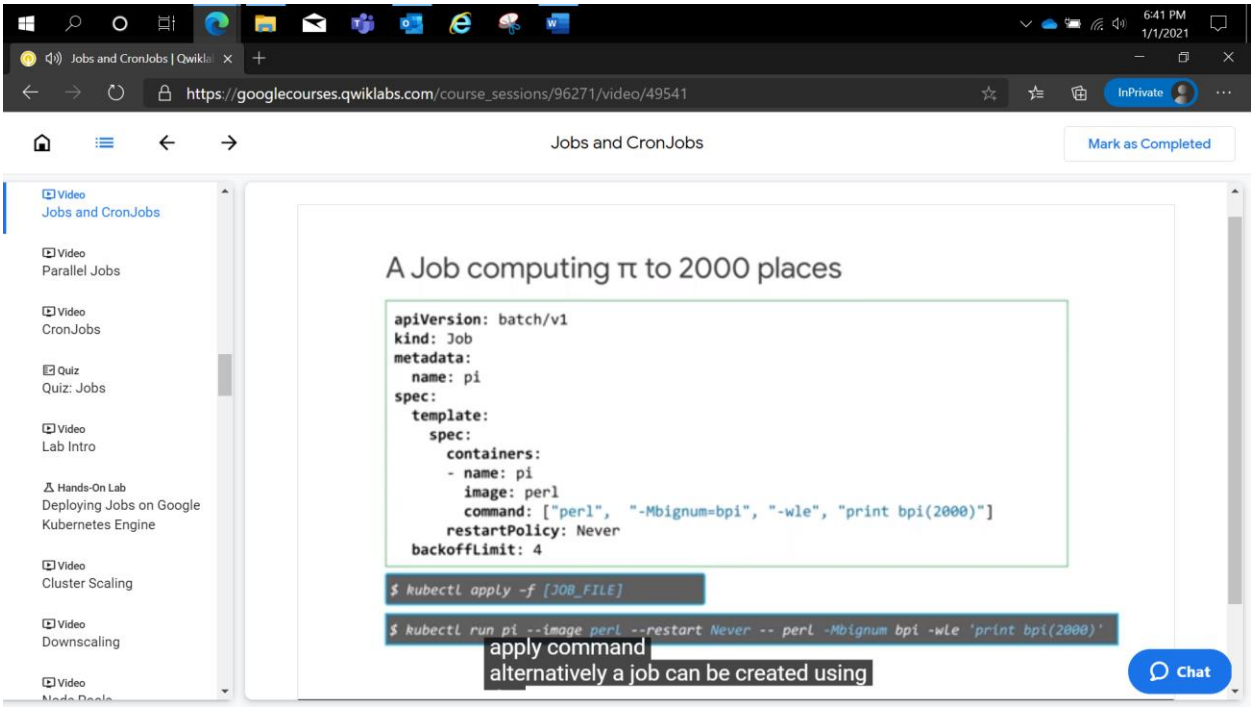
```
apiVersion: v1
kind: Service
metadata:
  name: nginx
spec:
  type: LoadBalancer
  sessionAffinity: ClientIP
  selector:
    app: nginx
  ports:
  - protocol: TCP
    port: 60000
    targetPort: 80
```

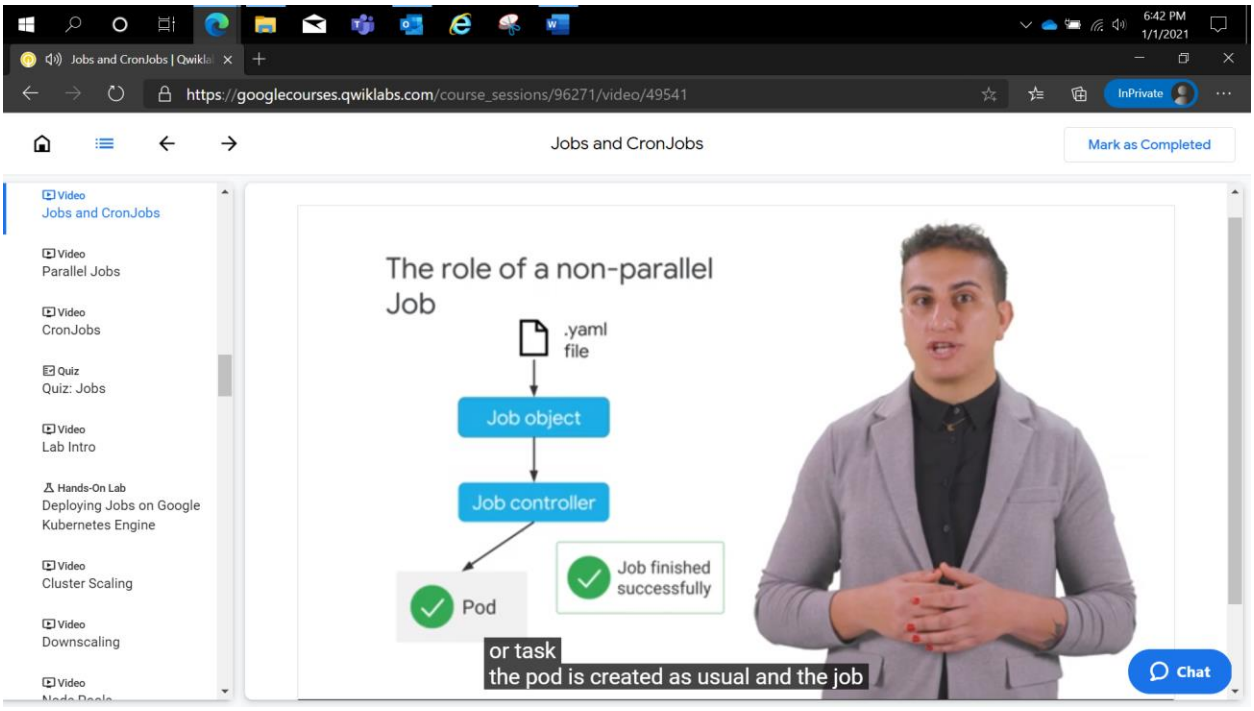


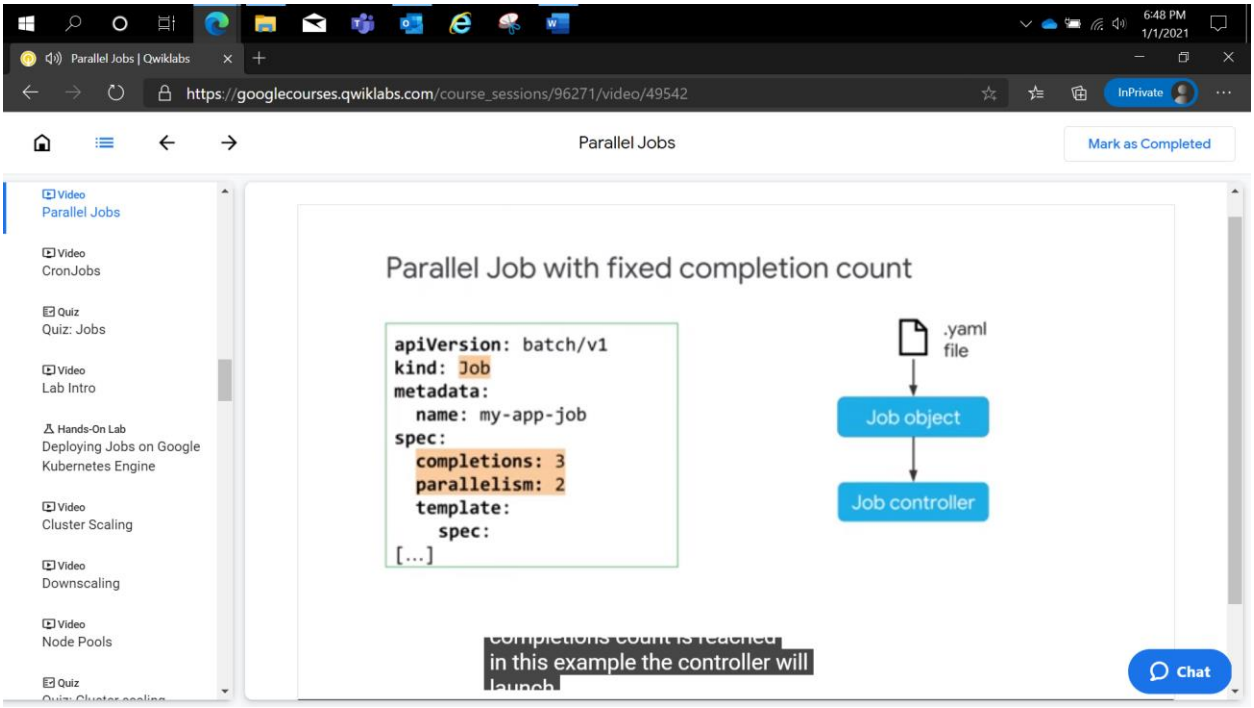

6:38 PM
1/1/2021
Jobs and CronJobs | Qwikl
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49541
InPrivate
Jobs and CronJobs
Mark as Completed
Video
Jobs and CronJobs
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
A scenario where Job provides the solution
Web server
Job: Transcode video files
Cluster
Master
Job controller
Node
Node
Job Pod
Container
the job controller reschedules the job
Chat

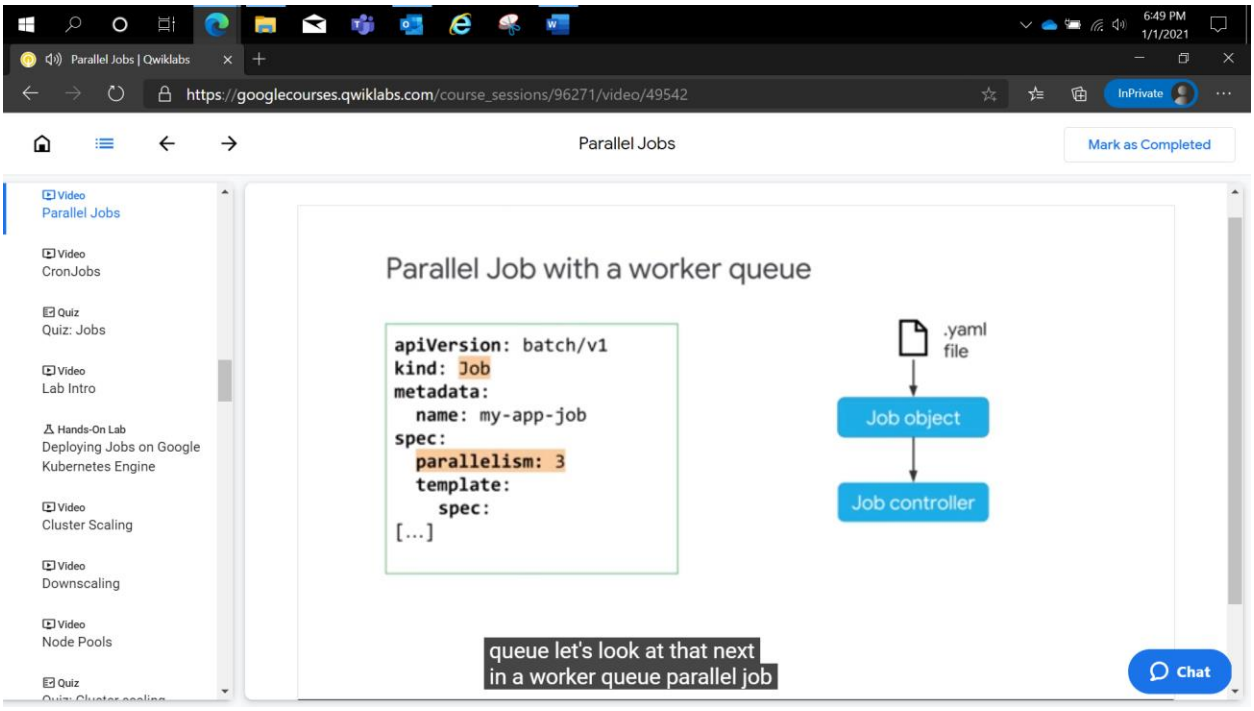

6:39 PM
1/1/2021
Jobs and CronJobs | Qwikl
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49541
InPrivate
Jobs and CronJobs
Mark as Completed
Video
Jobs and CronJobs
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
Jobs and CronJobs
A scenario where Job provides the solution
Copy link
Web server
Job: Transcode video files
Cluster
Master
Job controller
Node
Node
Job Pod
Container
on a different node the job controller continues
1:40 / 4:43
YouTube
Chat

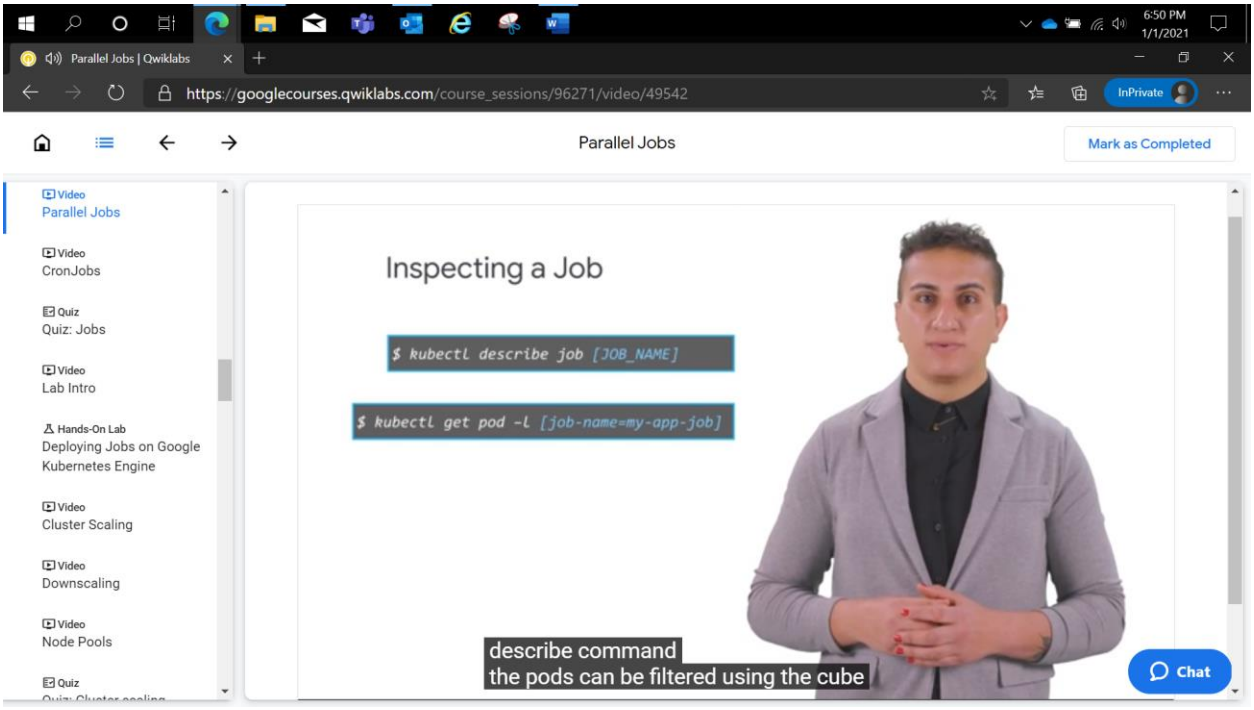

6:40 PM
1/1/2021
Jobs and CronJobs | Qwikla
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49541
InPrivate
Jobs and CronJobs
Mark as Completed
Video
Jobs and CronJobs
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
A Job computing π to 2000 places
apiVersion: batch/v1
kind: Job
metadata:
name: pi
spec:
template:
spec:
containers:
- name: pi
image: perl
command: ["perl", "-Mbignum=bpi", "-wle", "print bpi(2000)"]
restartPolicy: Never
backoffLimit: 4
this means that if a container in a pod fails for any
Chat

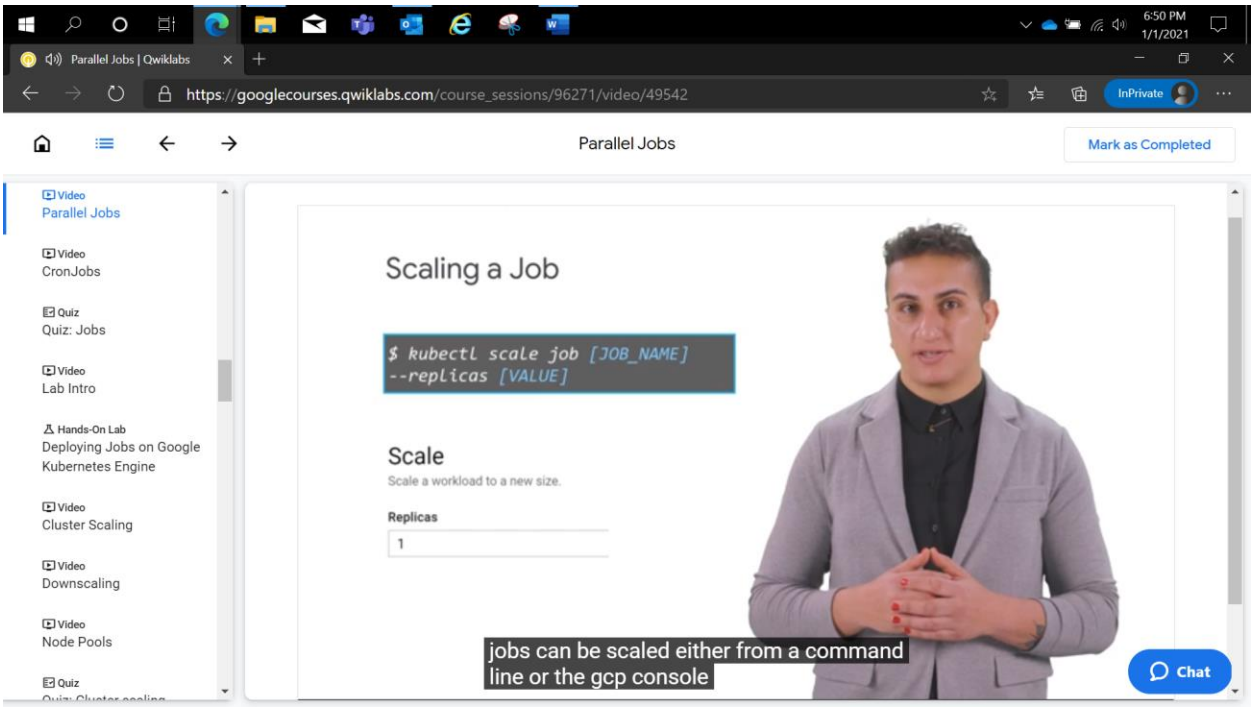

6:41 PM
1/1/2021
Jobs and CronJobs | Qwikla
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49541
InPrivate
Jobs and CronJobs
Mark as Completed
Video
Jobs and CronJobs
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
A Job computing π to 2000 places
apiVersion: batch/v1
kind: Job
metadata:
name: pi
spec:
template:
spec:
containers:
- name: pi
image: perl
command: ["perl", "-Mbignum=bpi", "-wle", "print bpi(2000)"]
restartPolicy: Never
backoffLimit: 4
$ kubectl apply -f [JOB_FILE]
$ kubectl run pi --image perl --restart Never -- perl -Mbignum bpi -wle 'print bpi(2000)'
apply command
alternatively a job can be created using
Chat

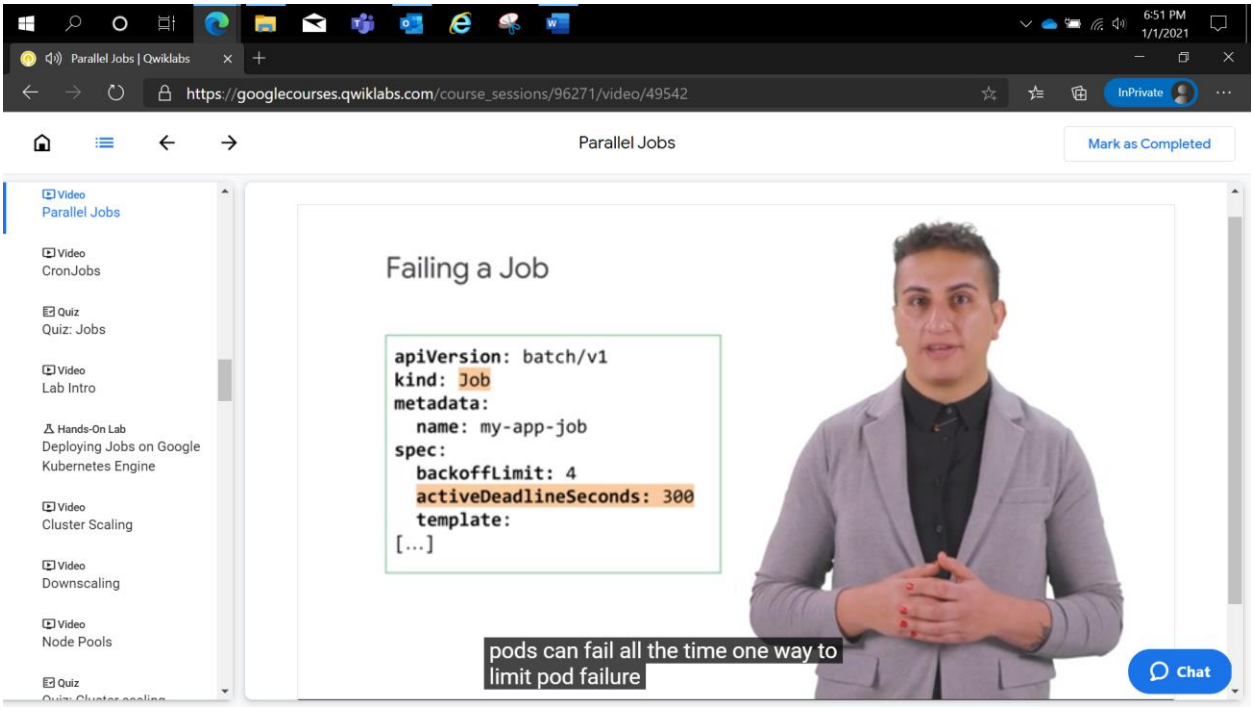

6:42 PM
1/1/2021
Jobs and CronJobs | Qwikla
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49541
InPrivate
Jobs and CronJobs
Mark as Completed
Video
Jobs and CronJobs
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
The role of a non-parallel Job
.yaml file
Job object
Job controller
Pod
Job finished successfully
or task
the pod is created as usual and the job
Chat

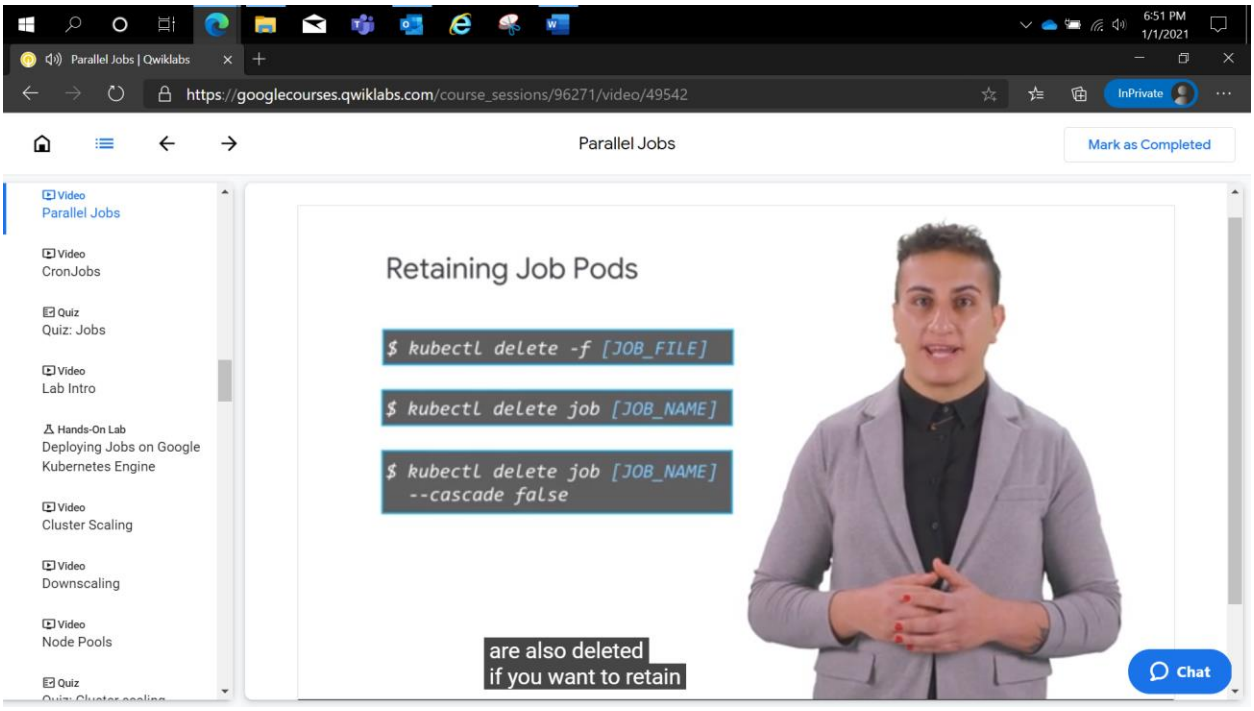

6:48 PM
1/1/2021
Parallel Jobs | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49542
InPrivate
Parallel Jobs
Mark as Completed
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
Node Pools
Quiz
Parallel Job with fixed completion count
apiVersion: batch/v1
kind: Job
metadata:
name: my-app-job
spec:
completions: 3
parallelism: 2
template:
spec:
[...]
.yaml file
Job object
Job controller
in this example the controller will
Chat

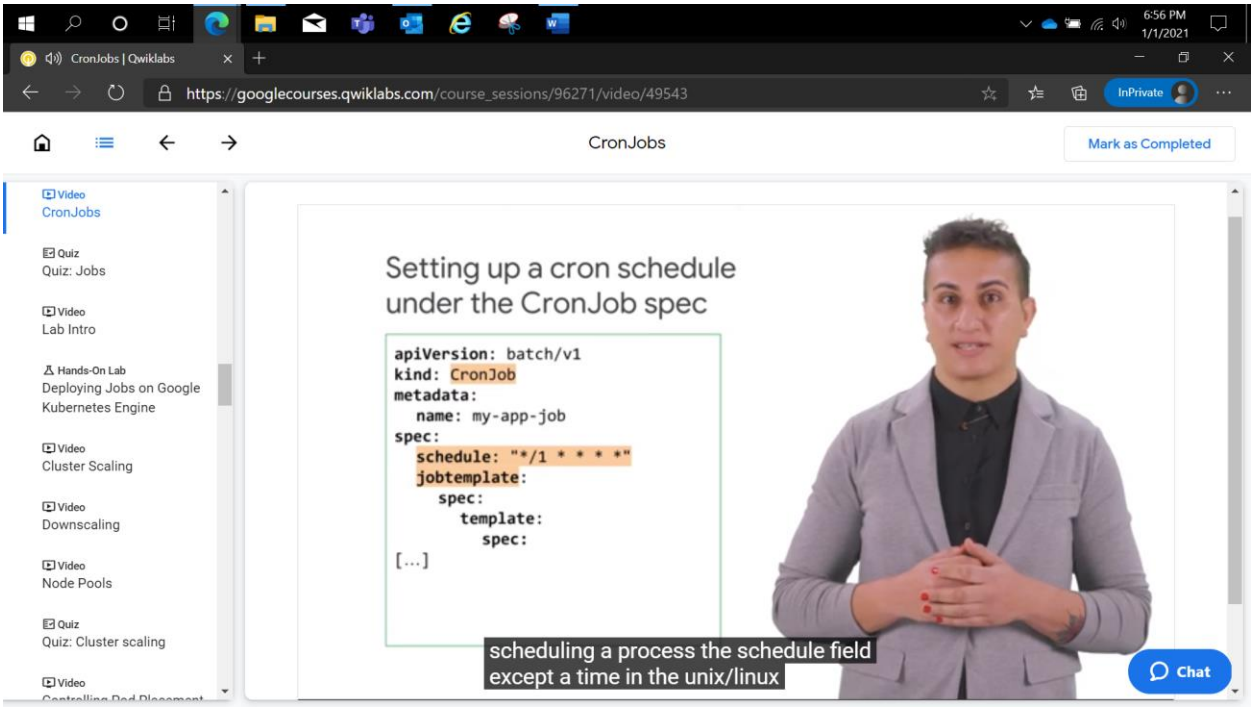

6:49 PM
1/1/2021
Parallel Jobs | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49542
InPrivate
Parallel Jobs
Mark as Completed
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
Node Pools
Quiz
Parallel Job with a worker queue
apiVersion: batch/v1
kind: Job
metadata:
name: my-app-job
spec:
parallelism: 3
template:
spec:
[...]
.yaml file
Job object
Job controller
queue let's look at that next
in a worker queue parallel job
Chat

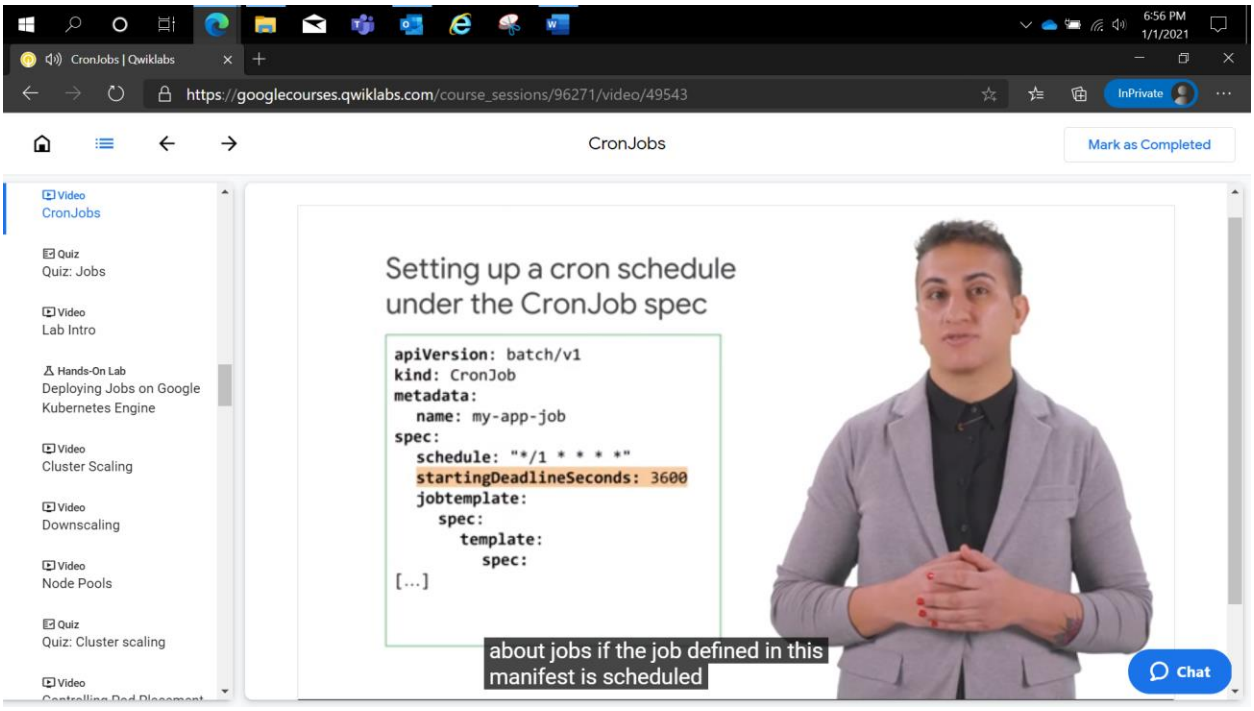

6:50 PM
1/1/2021
Parallel Jobs | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49542
InPrivate
Parallel Jobs
Mark as Completed
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
Node Pools
Quiz
Inspecting a Job
$ kubectl describe job [JOB_NAME]
$ kubectl get pod -l [job-name=my-app-job]
describe command
the pods can be filtered using the cube
Chat

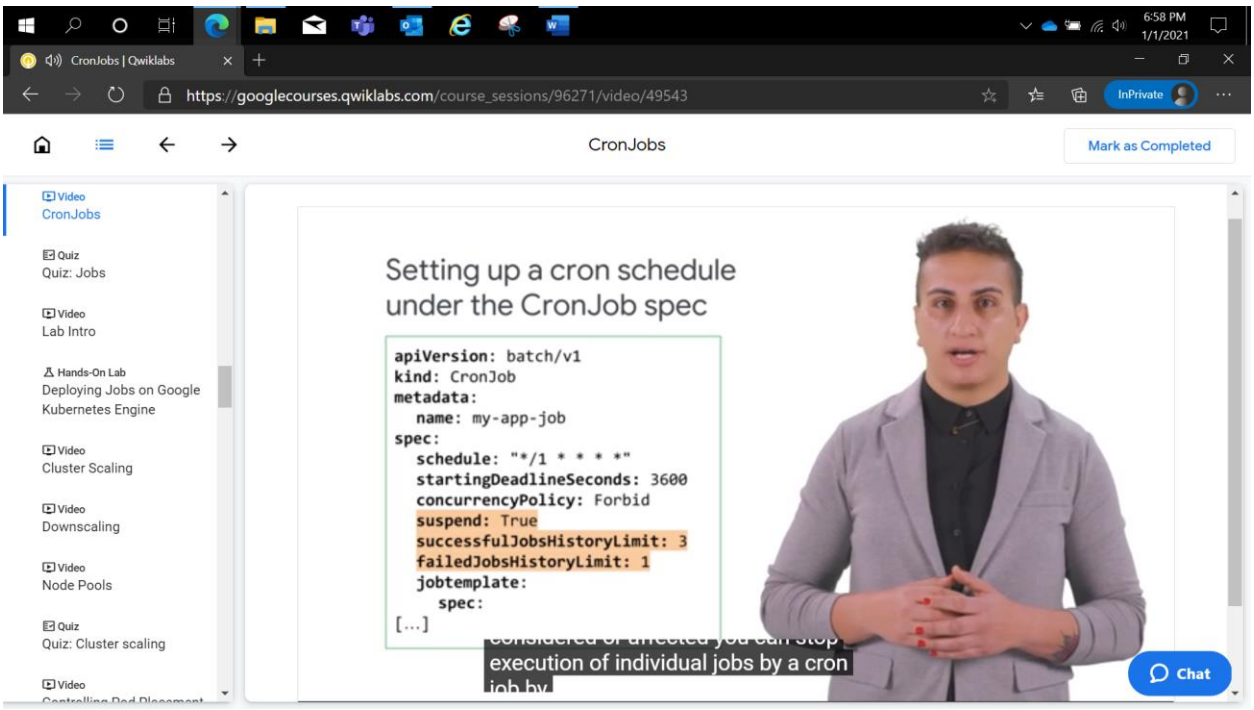

6:50 PM
1/1/2021
Parallel Jobs | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49542
InPrivate
Parallel Jobs
Mark as Completed
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
Node Pools
Quiz
Scaling a Job
$ kubectl scale job [JOB_NAME] --replicas [VALUE]
Scale
Scale a workload to a new size.
Replicas
1
jobs can be scaled either from a command line or the gcp console
Chat

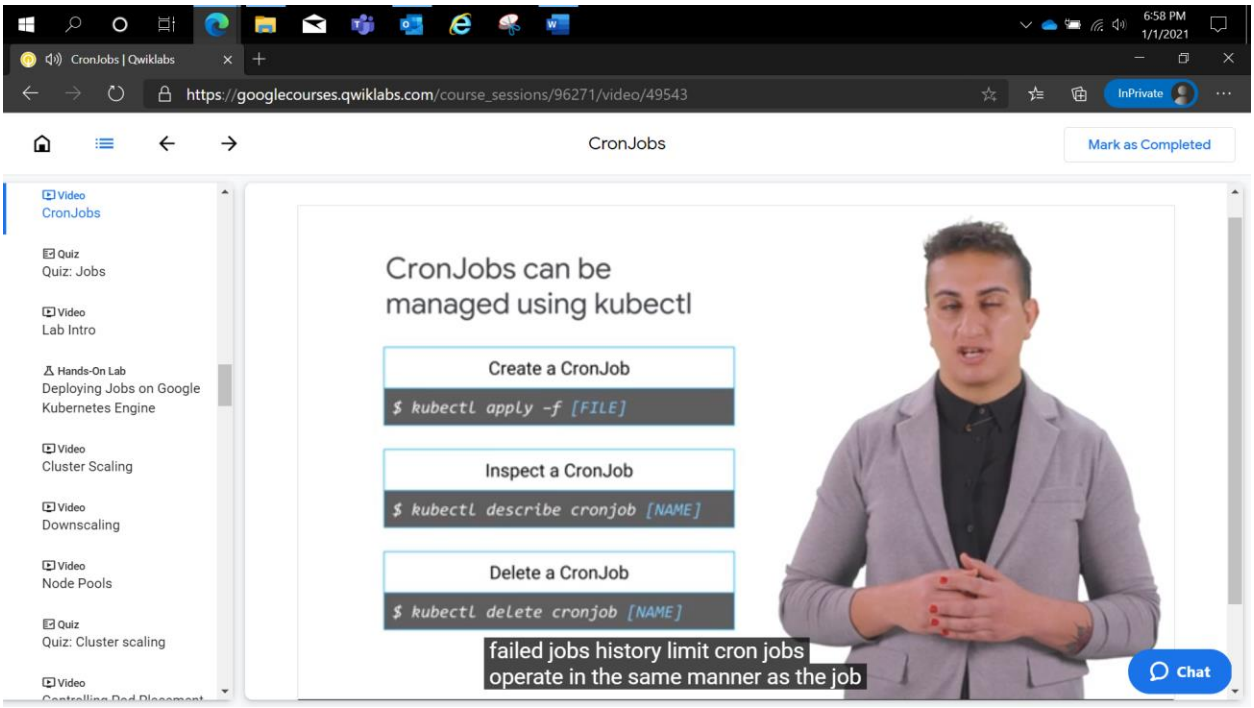

6:51 PM
1/1/2021
Parallel Jobs | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49542
InPrivate
Parallel Jobs
Mark as Completed
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
Node Pools
Quiz
Failing a Job
apiVersion: batch/v1
kind: Job
metadata:
name: my-app-job
spec:
backoffLimit: 4
activeDeadlineSeconds: 300
template:
[...]
pods can fail all the time one way to limit pod failure
Chat

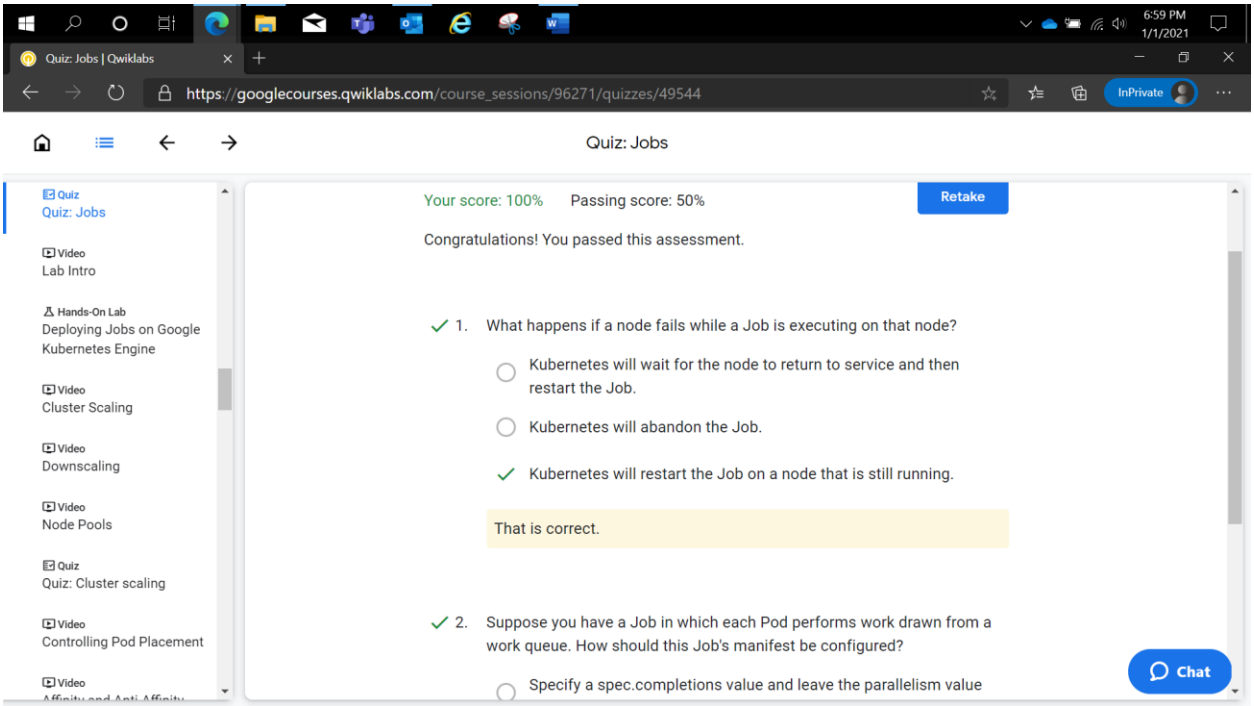

Parallel Jobs | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49542
InPrivate
6:51 PM
1/1/2021
Parallel Jobs
Mark as Completed
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
Node Pools
Quiz
Retaining Job Pods
$ kubectl delete -f [JOB_FILE]
$ kubectl delete job [JOB_NAME]
$ kubectl delete job [JOB_NAME]
--cascade false
are also deleted
if you want to retain
Chat

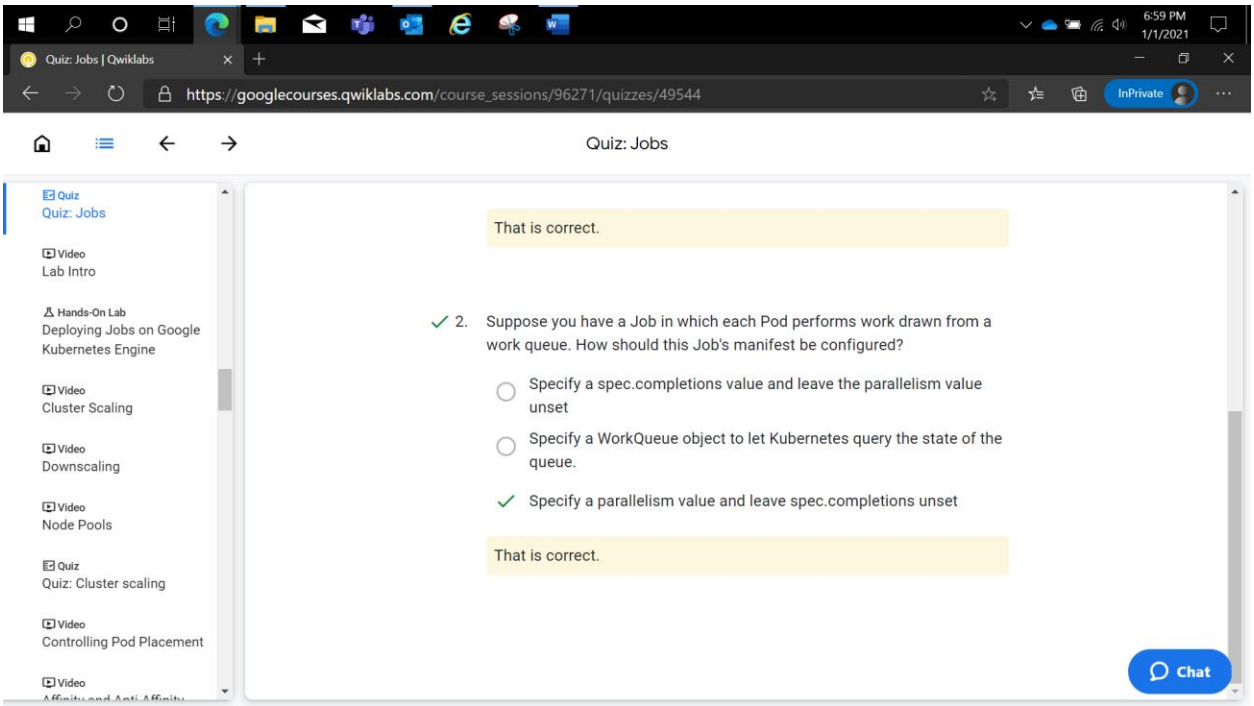

CronJobs | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49543
InPrivate
6:56 PM
1/1/2021
CronJobs
Mark as Completed
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
Node Pools
Quiz
Quiz: Cluster scaling
Video
Setting up a cron schedule under the CronJob spec
apiVersion: batch/v1
kind: CronJob
metadata:
name: my-app-job
spec:
schedule: "*/1 * * * *"
jobtemplate:
spec:
template:
spec:
[...]
scheduling a process the schedule field
except a time in the unix/linux
Chat


6:56 PM
1/1/2021
CronJobs | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49543
InPrivate
CronJobs
Mark as Completed
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
Node Pools
Quiz
Quiz: Cluster scaling
Video
Setting up a cron schedule under the CronJob spec
apiVersion: batch/v1
kind: CronJob
metadata:
name: my-app-job
spec:
schedule: "*/1 * * * *"
startingDeadlineSeconds: 3600
jobtemplate:
spec:
template:
spec:
[...]
about jobs if the job defined in this manifest is scheduled
Chat


6:58 PM
1/1/2021
CronJobs | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49543
InPrivate
CronJobs
Mark as Completed
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
Node Pools
Quiz
Quiz: Cluster scaling
Video
Setting up a cron schedule under the CronJob spec
apiVersion: batch/v1
kind: CronJob
metadata:
name: my-app-job
spec:
schedule: "*/1 * * * *"
startingDeadlineSeconds: 3600
concurrencyPolicy: Forbid
suspend: True
successfulJobsHistoryLimit: 3
failedJobsHistoryLimit: 1
jobtemplate:
spec:
[...]
execution of individual jobs by a cron
Chat


6:58 PM
1/1/2021
CronJobs | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49543
InPrivate
CronJobs
Mark as Completed
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
Node Pools
Quiz
Quiz: Cluster scaling
Video
CronJobs can be managed using kubectl
Create a CronJob
$ kubectl apply -f [FILE]
Inspect a CronJob
$ kubectl describe cronjob [NAME]
Delete a CronJob
$ kubectl delete cronjob [NAME]
failed jobs history limit cron jobs operate in the same manner as the job
Chat


6:59 PM
1/1/2021
Quiz: Jobs | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49544
InPrivate
Quiz: Jobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
Node Pools
Quiz
Quiz: Cluster scaling
Video
Controlling Pod Placement
Your score: 100%
Passing score: 50%
Retake
Congratulations! You passed this assessment.
1. What happens if a node fails while a Job is executing on that node?
Kubernetes will wait for the node to return to service and then restart the Job.
Kubernetes will abandon the Job.
Kubernetes will restart the Job on a node that is still running.
That is correct.
2. Suppose you have a Job in which each Pod performs work drawn from a work queue. How should this Job's manifest be configured?
Specify a spec.completions value and leave the parallelism value
Chat



Deploying Jobs on Google Kubernetes Engine

1 hourFree

Rate Lab

Overview

In this lab, you define and run Jobs and CronJobs.

In GKE, a Job is a controller object that represents a finite task. Jobs manage a task as it runs to completion, rather than managing an ongoing desired state such as the maintaining the total number of running Pods.

CronJobs perform finite, time-related tasks that run once or repeatedly at a time that you specify using Job objects to complete their tasks.

Objectives

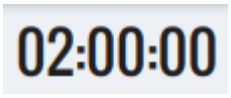
In this lab, you learn how to perform the following tasks:

- Define, deploy and clean up a GKE Job
- Define, deploy and clean up a GKE CronJob

Task 0. Lab Setup

Access Qwiklabs

For each lab, you get a new GCP project and set of resources for a fixed time at no cost.

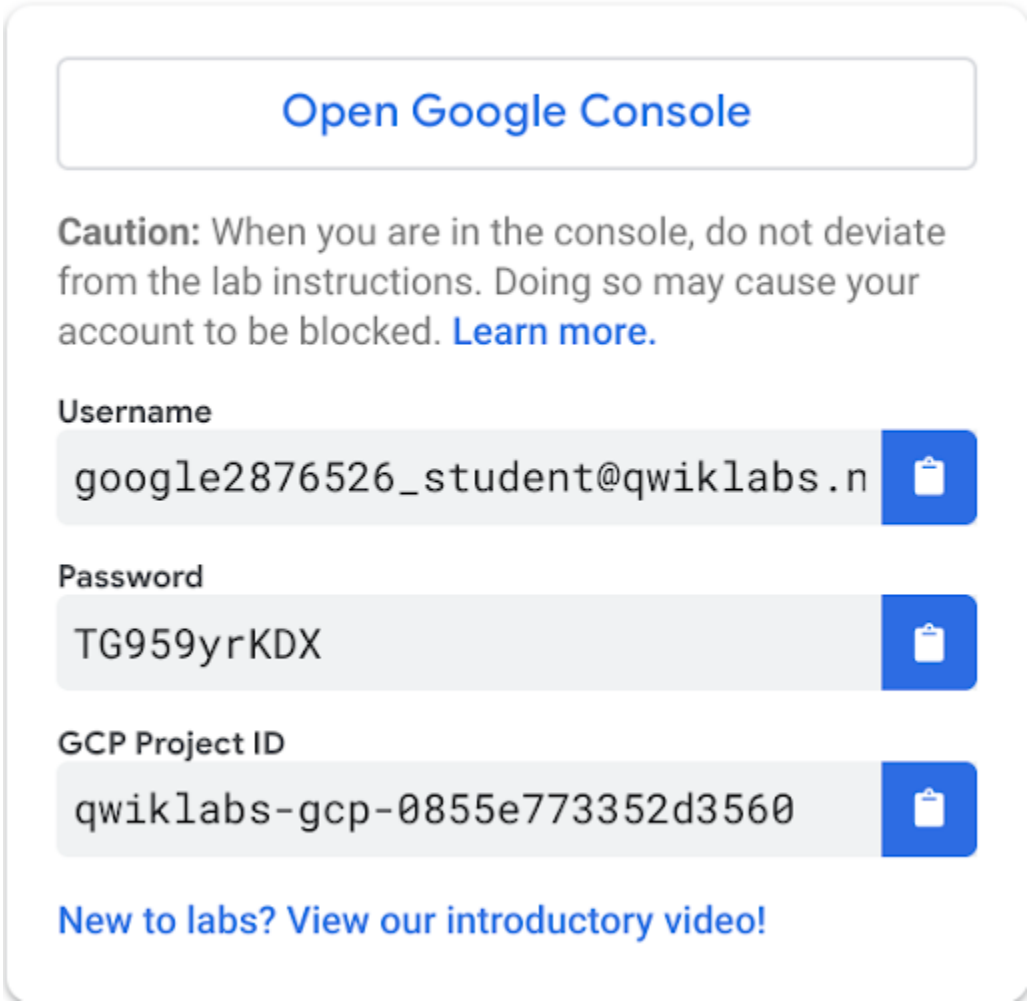
1. Make sure you signed into Qwiklabs using an **incognito window**.
2. Note the lab's access time (for example, 

and make sure you can finish in that time block.

There is no pause feature. You can restart if needed, but you have to start at the beginning.

3. When ready, click 

.
4. Note your lab credentials. You will use them to sign in to Cloud Platform Console.



5. Click **Open Google Console**.

6. Click **Use another account** and copy/paste credentials for **this** lab into the prompts.

If you use other credentials, you'll get errors or **incur charges**.

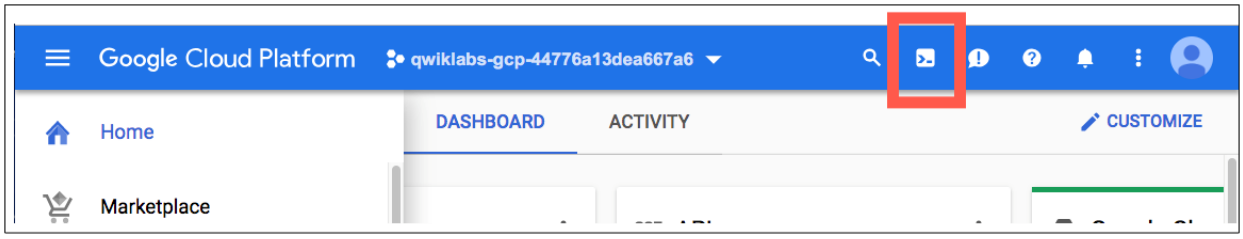
7. Accept the terms and skip the recovery resource page.

Do not click **End Lab** unless you are finished with the lab or want to restart it. This clears your work and removes the project.
After you complete the initial sign-in steps, the project dashboard appears.

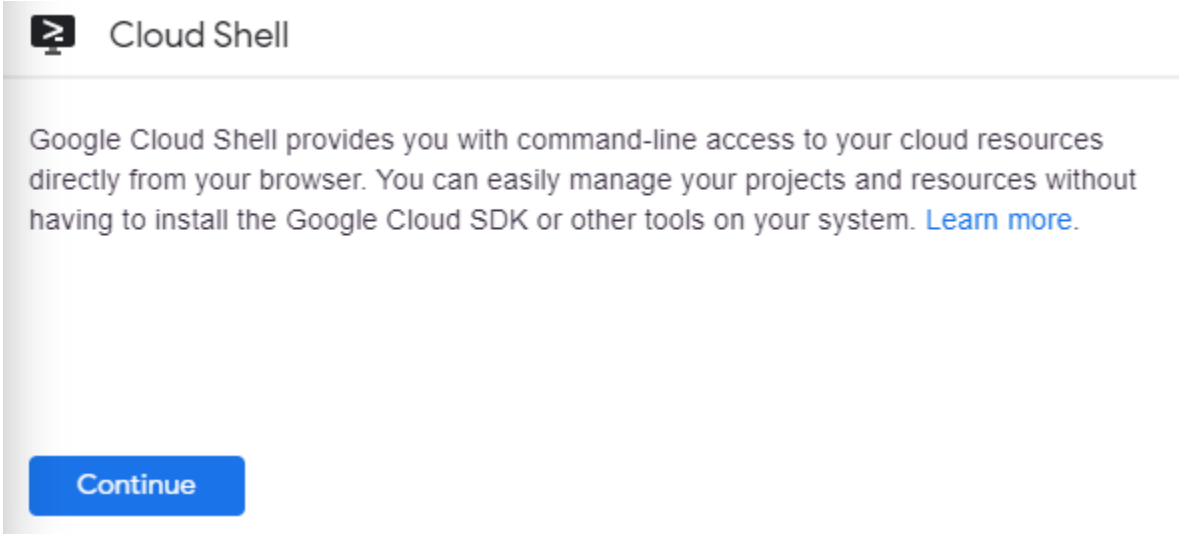
Activate Google Cloud Shell

Google Cloud Shell is a virtual machine that is loaded with development tools. It offers a persistent 5GB home directory and runs on the Google Cloud. Google Cloud Shell provides command-line access to your GCP resources.

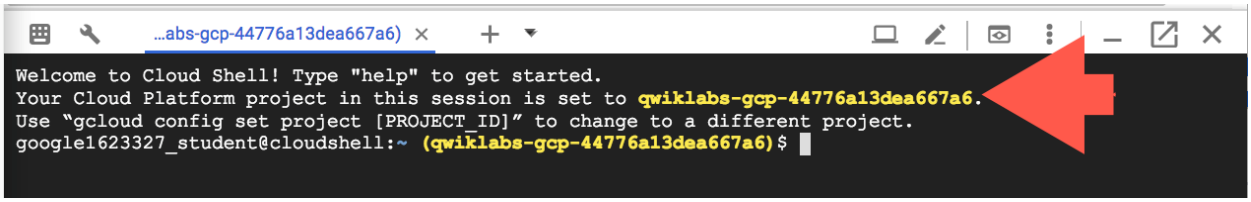
1. In GCP console, on the top right toolbar, click the Open Cloud Shell button.



2. Click **Continue**.



It takes a few moments to provision and connect to the environment. When you are connected, you are already authenticated, and the project is set to your *PROJECT_ID*. For example:



gcloud is the command-line tool for Google Cloud Platform. It comes pre-installed on Cloud Shell and supports tab-completion.

You can list the active account name with this command:

```
gcloud auth list
```
content_copy

Output:

```
Credentialed accounts:
 - <myaccount>@<mydomain>.com (active)
```
content_copy

Example output:

```
Credentialed accounts:
 - google1623327_student@qwiklabs.net
```
content_copy

You can list the project ID with this command:

```
gcloud config list project
```

```
content_copy
```
Output:

```
[core]
project = <project_ID>content_copy
```
Example output:

```
[core]
project = qwiklabs-gcp-44776a13dea667a6content_copy
```
Full documentation of **gcloud** is available on Google Cloud gcloud Overview .

Task 1. Define and deploy a Job manifest

In GKE, a Job is a controller object that represents a finite task.

In this task, you create a Job, inspect its status, and then remove it.

Connect to the lab Google Kubernetes Engine cluster

1. In Cloud Shell, type the following command to set the environment variable for the zone and cluster name.

```
export my_zone=us-central1-a
export my_cluster=standard-cluster-1
content_copy
```
2. Configure kubectl tab completion in Cloud Shell.

```
source <(kubectl completion bash)
content_copy
```
3. In Cloud Shell, configure access to your cluster for the kubectl command-line tool, using the following command:

```
gcloud container clusters get-credentials $my_cluster --zone $my_zone
```

```
content_copy
```

4. In Cloud Shell enter the following command to clone the repository to the lab Cloud Shell.

```
git clone https://github.com/GoogleCloudPlatform/training-data-analyst
content_copy
```

5. Create a soft link as a shortcut to the working directory.

```
ln -s ~/training-data-analyst/courses/ak8s/v1.1 ~/ak8s
content_copy
```

6. Change to the directory that contains the sample files for this lab.

```
cd ~/ak8s/Jobs_CronJobs
content_copy
```

Create and run a Job

You will create a job using a sample deployment manifest called `example-job.yaml` that has been provided for you. This Job computes the value of Pi to 2,000 places and then prints the result.

```
apiVersion: batch/v1
kind: Job
metadata:
  # Unique key of the Job instance
  name: example-job
spec:
  template:
    metadata:
      name: example-job
    spec:
      containers:
      - name: pi
        image: perl
        command: ["perl"]
        args: ["-Mbignum=bpi", "-wle", "print bpi(2000)"]
      # Do not restart containers after they exit
      restartPolicy: Never
content_copy
```

1. To create a Job from this file, execute the following command:

```
kubectl apply -f example-job.yaml
content_copy
```

Click *Check my progress* to verify the objective.

Create and run a Job

Check my progress

2. To check the status of this Job, execute the following command:

```
kubectl describe job example-job
content_copy
```

You will see details of the job, including the Pod statuses indicating how many jobs are still running, how many completed successfully and how many failed.

Output (do not copy)

```
...
Start Time:     Thu, 20 Dec 2018 14:34:09 +0000
Pods Statuses:  0 Running / 1 Succeeded / 0 Failed
...
content_copy
```

3. To view all Pod resources in your cluster, including Pods created by the Job which have completed, execute the following command:

```
kubectl get pods
content_copy
```

Your Pod name might be different from the example output. Make a note of one of the Pod names.

Output (do not copy)

```
NAME                  READY     STATUS      RESTARTS   AGE
example-job-sqljc     0/1       Completed   0          1m
content_copy
```

Clean up and delete the Job

When a Job completes, the Job stops creating Pods. The Job API object is not removed when it completes, which allows you to view its status. Pods created by the Job are not deleted, but they are terminated. Retention of the Pods allows you to view their logs and to interact with them.

1. To get a list of the Jobs in the cluster, execute the following command:

```
kubectl get jobs
content_copy
```

The output should look like the example.

Output (do not copy)

```
NAME           COMPLETIONS   DURATION       AGE
example-job    1/1           75s            2m5s
content_copy
```

2. To retrieve the log file from the Pod that ran the Job execute the following command. You must replace [POD-NAME] with the node name you recorded in the last task

```
kubectl logs [POD-NAME]
content_copy
```

The output will show that the job wrote the first two thousand digits of pi to the Pod log.

3. To delete the Job, execute the following command:

```
kubectl delete job example-job
content_copy
```

If you try to query the logs again the command will fail as the Pod can no longer be found.

Task 2. Define and deploy a CronJob manifest

You can create CronJobs to perform finite, time-related tasks that run once or repeatedly at a time that you specify.

In this task, you create and run a CronJob, and then you clean up and delete the Job.

Create and run a CronJob

The CronJob manifest file `example-cronjob.yaml` has been provided for you. This CronJob deploys a new container every minute that prints the time, date and "Hello, World!".

```
apiVersion: batch/v1beta1
kind: CronJob
metadata:
  name: hello
spec:
  schedule: "*/1 * * * *"
  jobTemplate:
    spec:
      template:
        spec:
          containers:
          - name: hello
            image: busybox
            args:
            - /bin/sh
            - -c
            - date; echo "Hello, World!"
          restartPolicy: OnFailure
content_copy
```

Note

CronJobs use the required `schedule` field, which accepts a time in the Unix standard `crontab` format. All CronJob times are in UTC:

- The first value indicates the minute (between 0 and 59).
- The second value indicates the hour (between 0 and 23).
- The third value indicates the day of the month (between 1 and 31).
- The fourth value indicates the month (between 1 and 12).
- The fifth value indicates the day of the week (between 0 and 6).

The `schedule` field also accepts * and ? as wildcard values. Combining / with ranges specifies that the task should repeat at a regular interval. In the example, `*/1 * * * *` indicates that the task should repeat every minute of every day of every month.

1. To create a Job from this file, execute the following command:

```
kubectl apply -f example-cronjob.yaml
content_copy
```

Click *Check my progress* to verify the objective.

Create and run a CronJob

Check my progress

2. To get a list of the Jobs in the cluster, execute the following command:

```
kubectl get jobs
content_copy
```

The output should look like the example.

Output (do not copy)

```
NAME               COMPLETIONS    DURATION     AGE
hello-1545013620   1/1            2s           18s
content_copy
```

3. To check the status of this Job, execute the following command, where `[job_name]` is the name of your job:

```
kubectl describe job [job_name]
content_copy
```

You will see details of the job, including the Pod statuses showing that one instance of this job was run.

Output (do not copy)

```
...
Start Time:     Thu, 20 Dec 2018 15:24:03 +0000
Pods Statuses:  0 Running / 1 Succeeded / 0 Failed
...
...Created pod: hello-1545319920-twkhl
content_copy
```

4. Make a note of the name of the Pod that was used by this job.

5. View the output of the Job by querying the logs for the Pod. Replace [POD-NAME] with the name of the Pod you recorded in the last step.

```
kubectl logs [POD-NAME]
content_copy
```

This will display the output of the shell script configured in the CronJob:

Output (do not copy)

```
Thu Dec 20 15:31:16 WET 2018
Hello,World!
content_copy
```

6. To view all job resources in your cluster, including all of the Pods created by the CronJob which have completed, execute the following command:

```
kubectl get jobs
content_copy
```

Your job names might be different from the example output. By default Kubernetes sets the Job history limits so that only the last three successful and last failed job are retained so this list will only contain the most recent three of four jobs.

Output (do not copy)

```
NAME                COMPLETIONS   DURATION     AGE
hello-1545013680    1             1            2m
hello-1545013740    1             1            1m
hello-1545013800    1             1            42s
content_copy
```

Clean up and delete the Job

In order to stop the CronJob and clean up the Jobs associated with it you must delete the CronJob.

1. To delete all these jobs, execute the following command:

```
kubectl delete cronjob hello
content_copy
```

2. To verify that the jobs were deleted, execute the following command:

```
kubectl get jobs
content_copy
```

The output should look like the example.

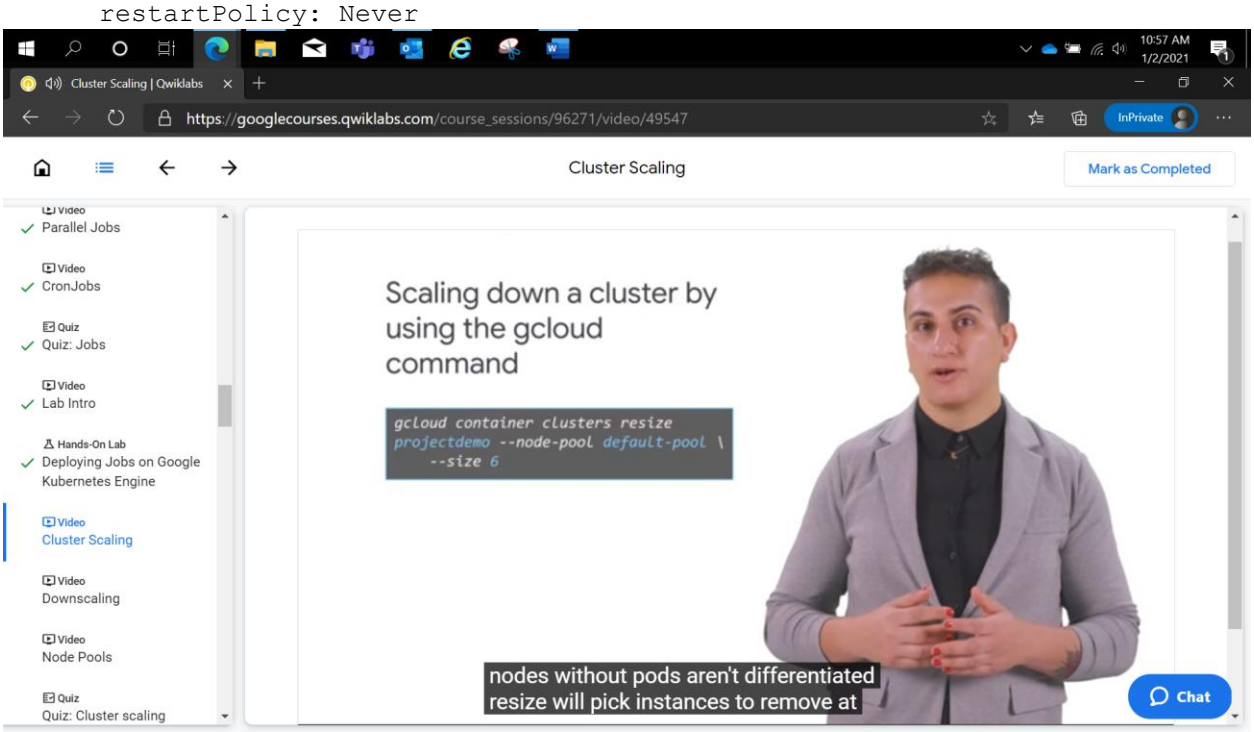
Output (do not copy)

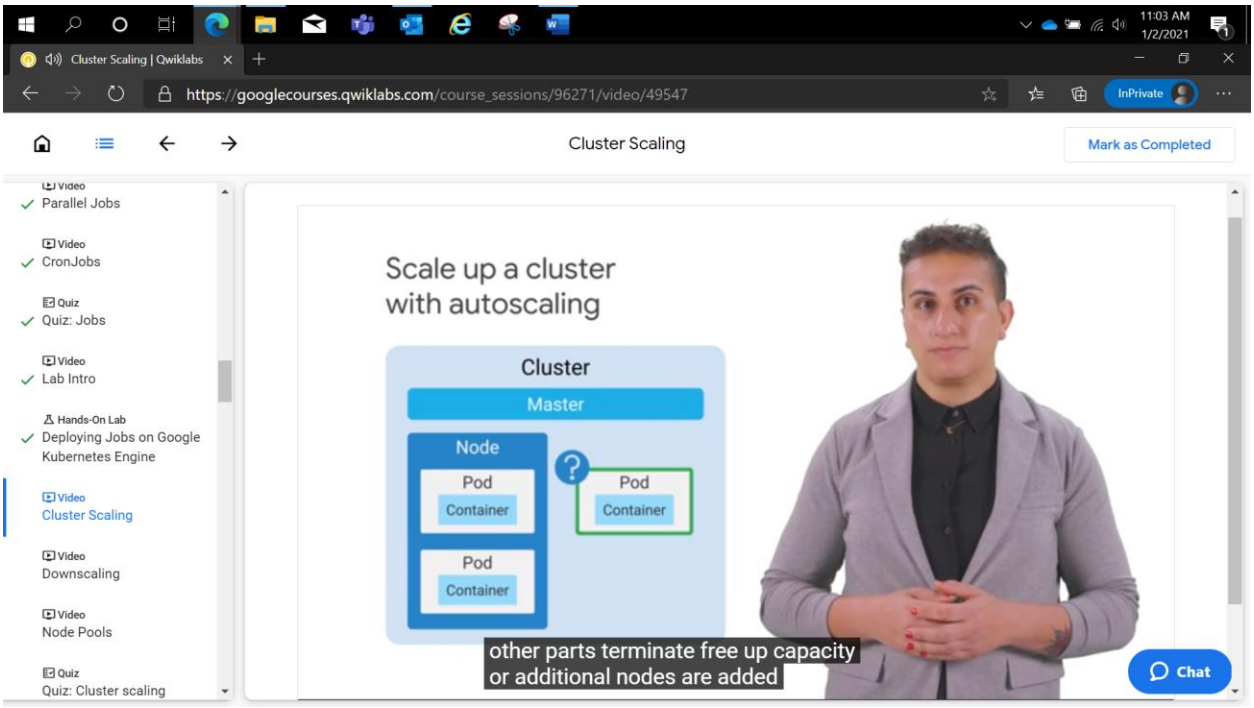
```
No resources found.
content_copy
```

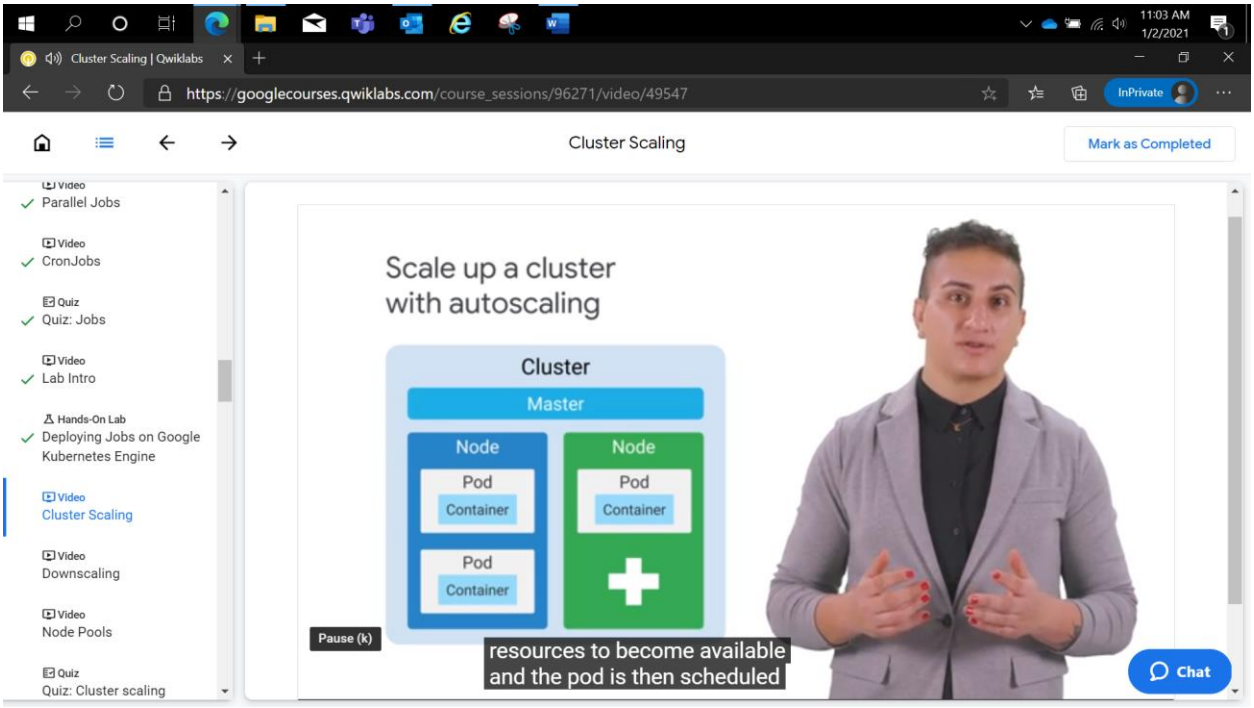
All the Jobs were removed

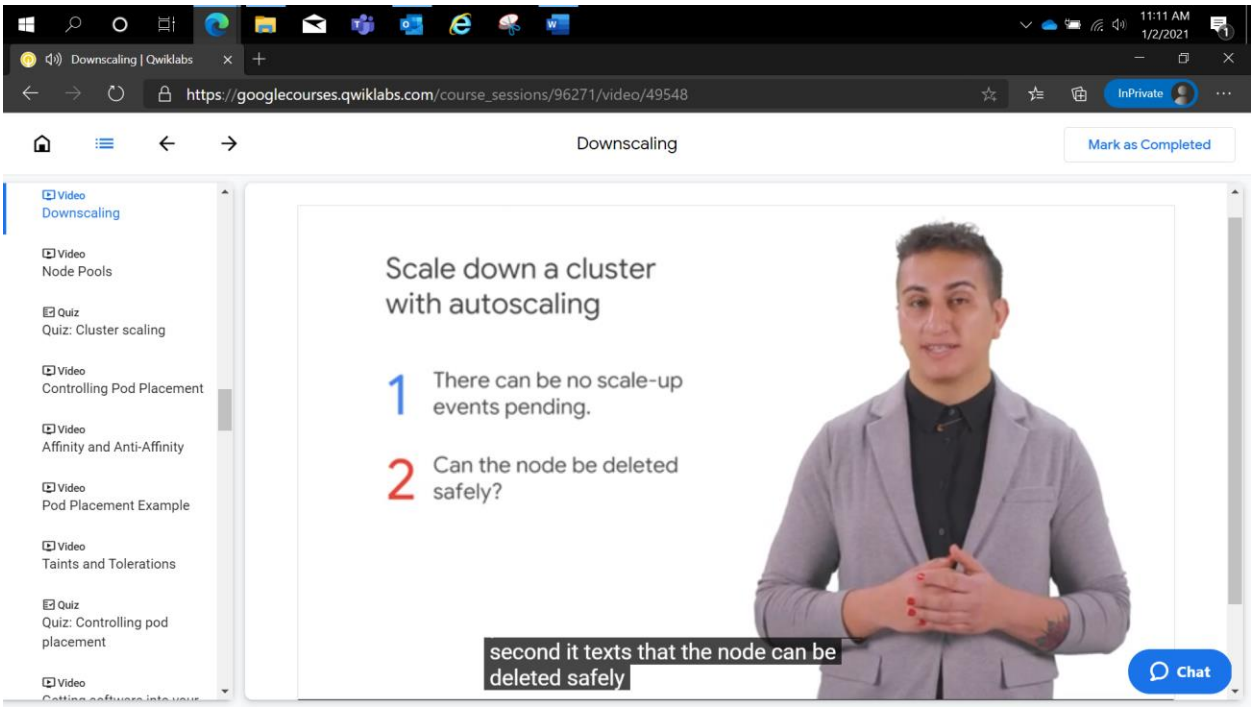
```
student_03_da03603be4b2@cloudshell:~/ak8s/Jobs_CronJobs (qwiklabs-gcp-03-
363474b52250)$ cat example-job.yaml
apiVersion: batch/v1
kind: Job
metadata:
  # Unique key of the Job instance
  name: example-job
```

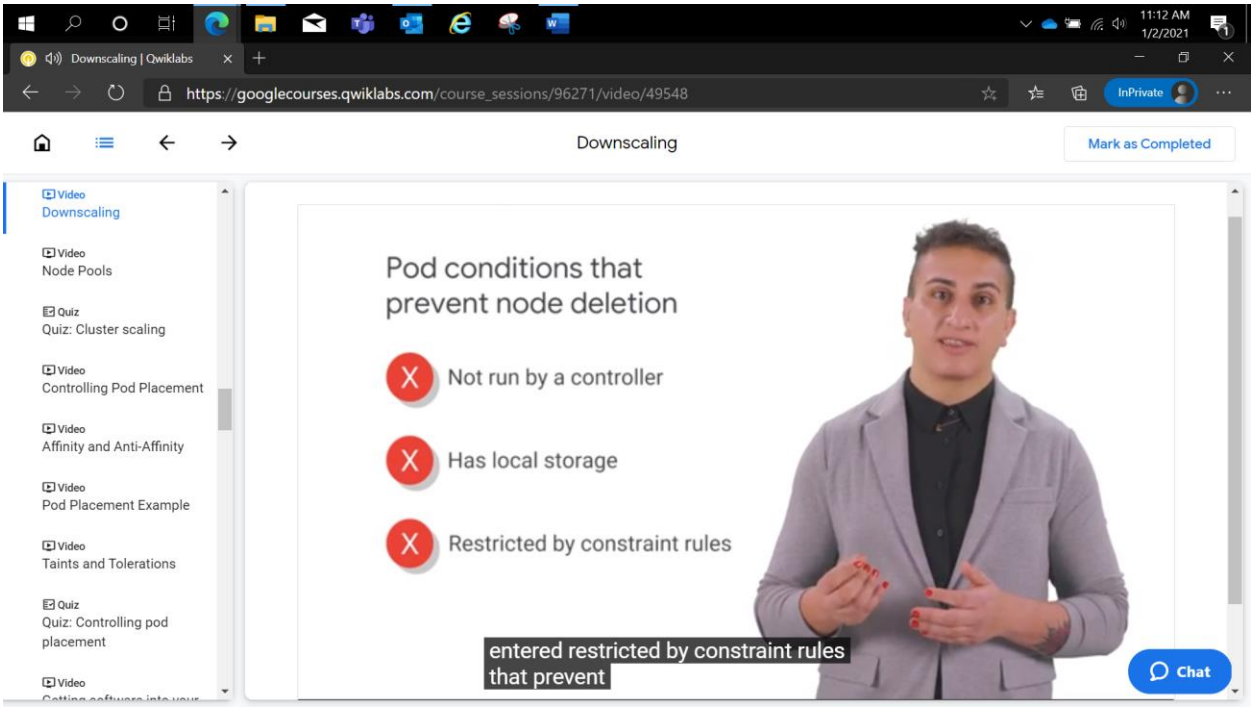
```
spec:
  template:
    metadata:
      name: example-job
    spec:
      containers:
      - name: pi
        image: perl
        command: ["perl"]
        args: ["-Mbignum=bpi", "-wle", "print bpi(2000)"]
      # Do not restart containers after they exit
      restartPolicy: Never
```

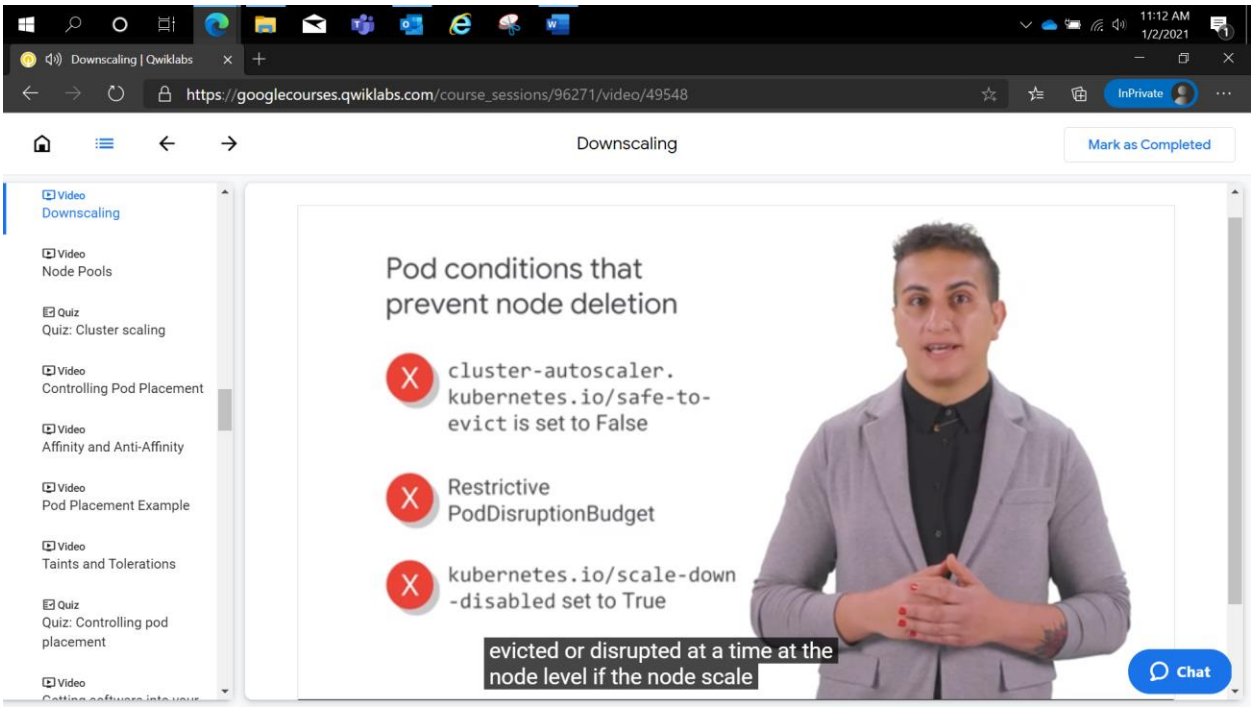


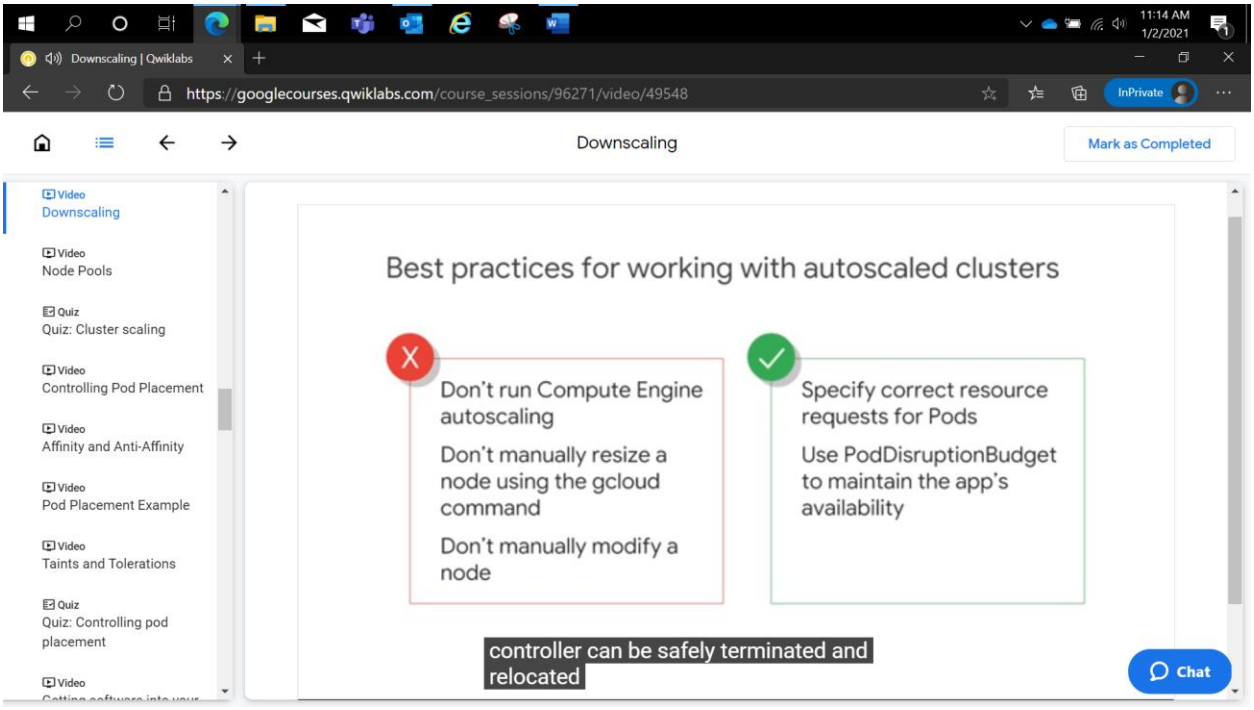

Cluster Scaling | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49547
InPrivate
11:03 AM
1/2/2021
Cluster Scaling
Mark as Completed
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
Node Pools
Quiz
Quiz: Cluster scaling
Scale up a cluster with autoscaling
Cluster
Master
Node
Pod
Container
Pod
Container
Pod
Container
other parts terminate free up capacity or additional nodes are added
Chat

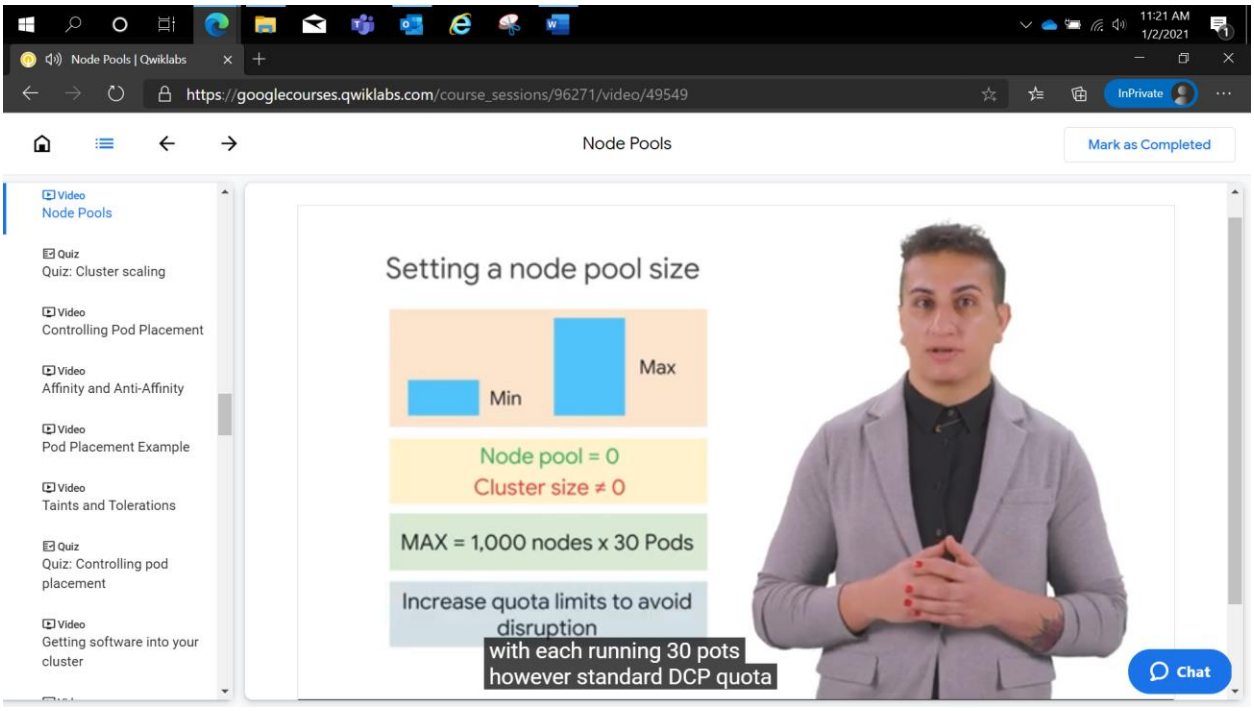

Cluster Scaling | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49547
InPrivate
11:03 AM
1/2/2021
Cluster Scaling
Mark as Completed
Video
Parallel Jobs
Video
CronJobs
Quiz
Quiz: Jobs
Video
Lab Intro
Hands-On Lab
Deploying Jobs on Google Kubernetes Engine
Video
Cluster Scaling
Video
Downscaling
Video
Node Pools
Quiz
Quiz: Cluster scaling
Scale up a cluster with autoscaling
Cluster
Master
Node
Pod
Container
Pod
Container
Node
Pod
Container
Pause (k)
resources to become available and the pod is then scheduled
Chat

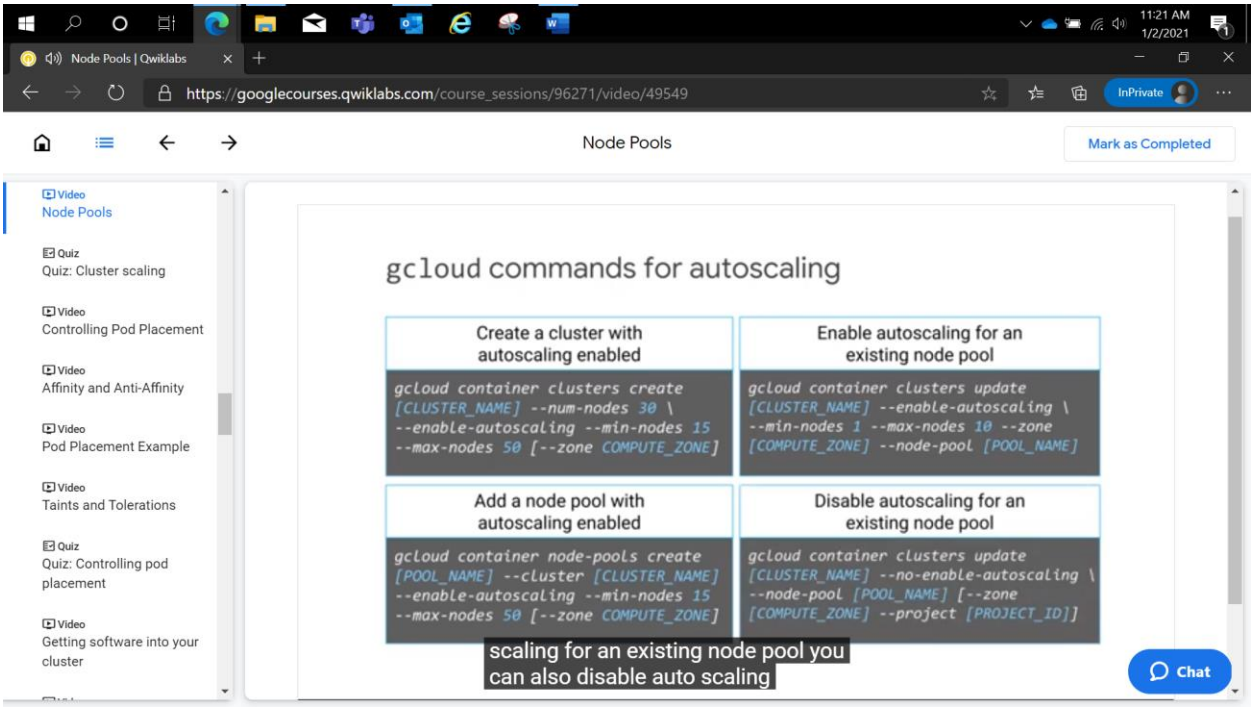

Downscaling | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49548
InPrivate
11:11 AM
1/2/2021
Downscaling
Mark as Completed
Video
Downscaling
Video
Node Pools
Quiz
Quiz: Cluster scaling
Video
Controlling Pod Placement
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Scale down a cluster with autoscaling
1 There can be no scale-up events pending.
2 Can the node be deleted safely?
second it texts that the node can be deleted safely
Chat

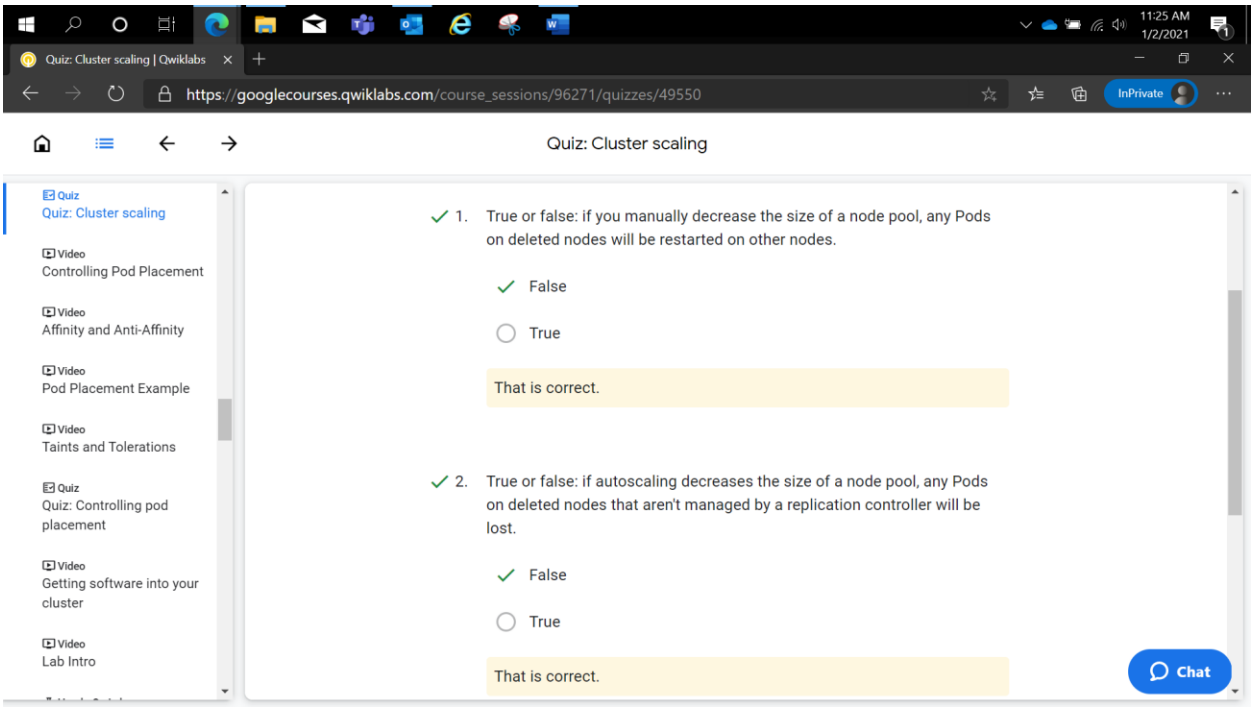

Downscaling | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49548
InPrivate
11:12 AM
1/2/2021
Downscaling
Mark as Completed
Video
Downscaling
Video
Node Pools
Quiz
Quiz: Cluster scaling
Video
Controlling Pod Placement
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Pod conditions that prevent node deletion
X Not run by a controller
X Has local storage
X Restricted by constraint rules
entered restricted by constraint rules that prevent
Chat

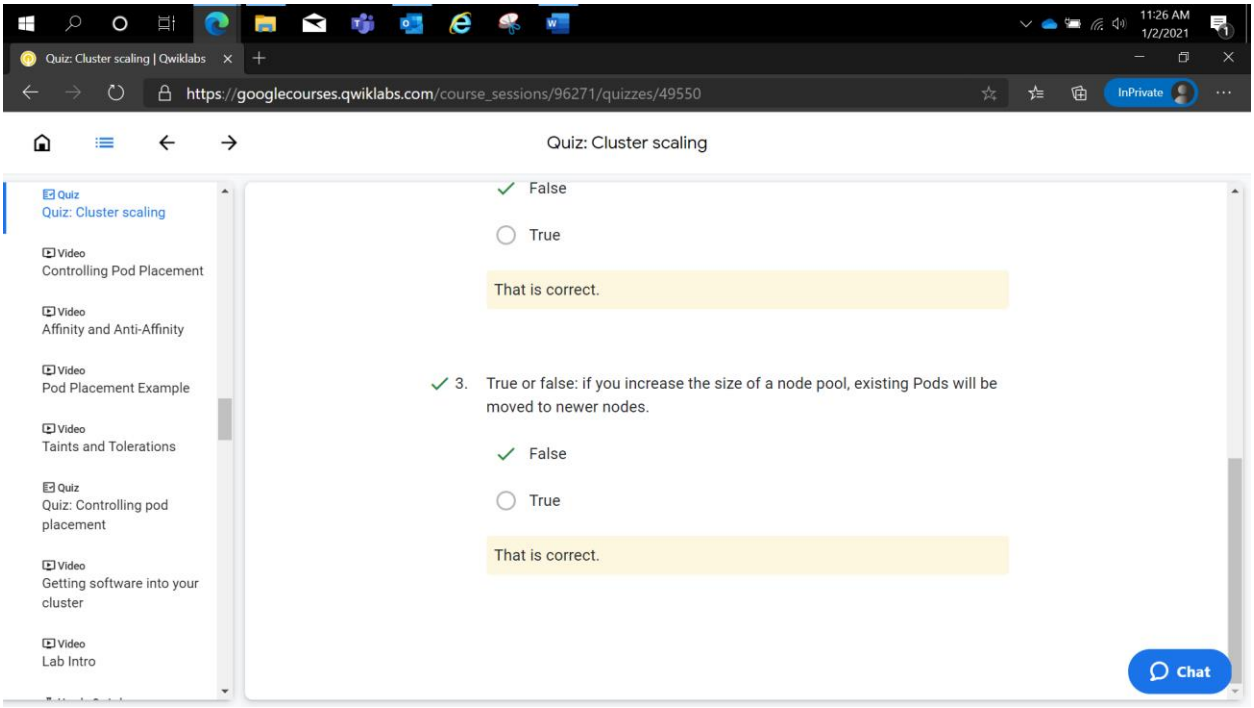

Downscaling | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49548
InPrivate
11:12 AM
1/2/2021
Downscaling
Mark as Completed
Video
Downscaling
Video
Node Pools
Quiz
Quiz: Cluster scaling
Video
Controlling Pod Placement
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Pod conditions that prevent node deletion
cluster-autoscaler.kubernetes.io/safe-to-evict is set to False
Restrictive PodDisruptionBudget
kubernetes.io/scale-down-disabled set to True
evicted or disrupted at a time at the node level if the node scale
Chat

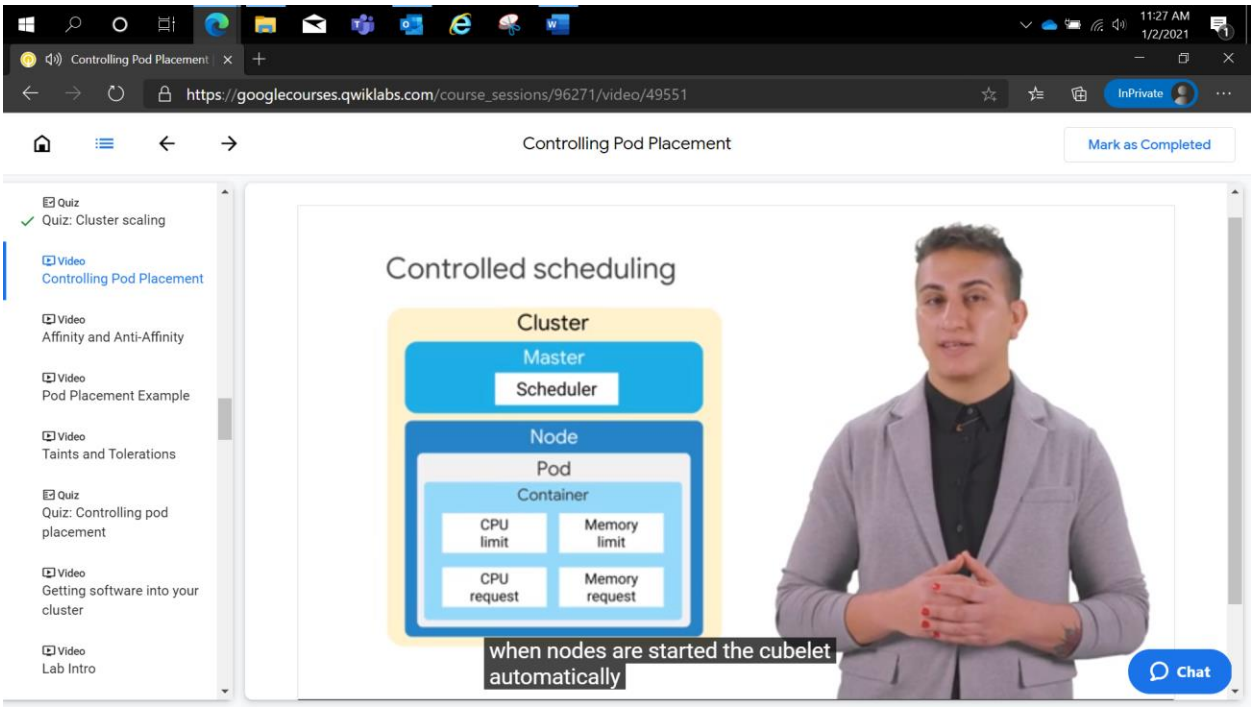

Downscaling | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49548
InPrivate
11:14 AM
1/2/2021
Downscaling
Mark as Completed
Video
Downscaling
Video
Node Pools
Quiz
Quiz: Cluster scaling
Video
Controlling Pod Placement
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Best practices for working with autoscaled clusters
Don't run Compute Engine autoscaling
Don't manually resize a node using the gcloud command
Don't manually modify a node
Specify correct resource requests for Pods
Use PodDisruptionBudget to maintain the app's availability
controller can be safely terminated and relocated
Chat

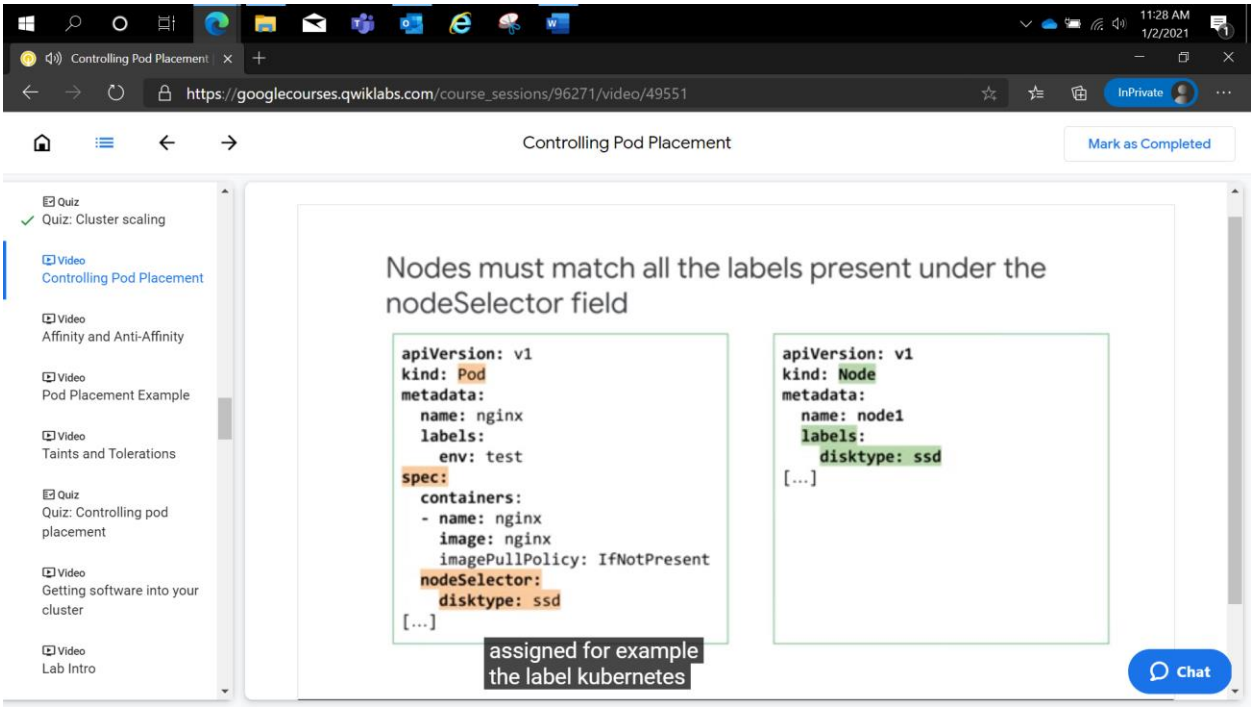

Node Pools | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49549
InPrivate
11:21 AM
1/2/2021
Node Pools
Mark as Completed
Video
Node Pools
Quiz
Quiz: Cluster scaling
Video
Controlling Pod Placement
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Setting a node pool size
Min
Max
Node pool = 0
Cluster size ≠ 0
MAX = 1,000 nodes x 30 Pods
Increase quota limits to avoid disruption
with each running 30 pots
however standard DCP quota
Chat

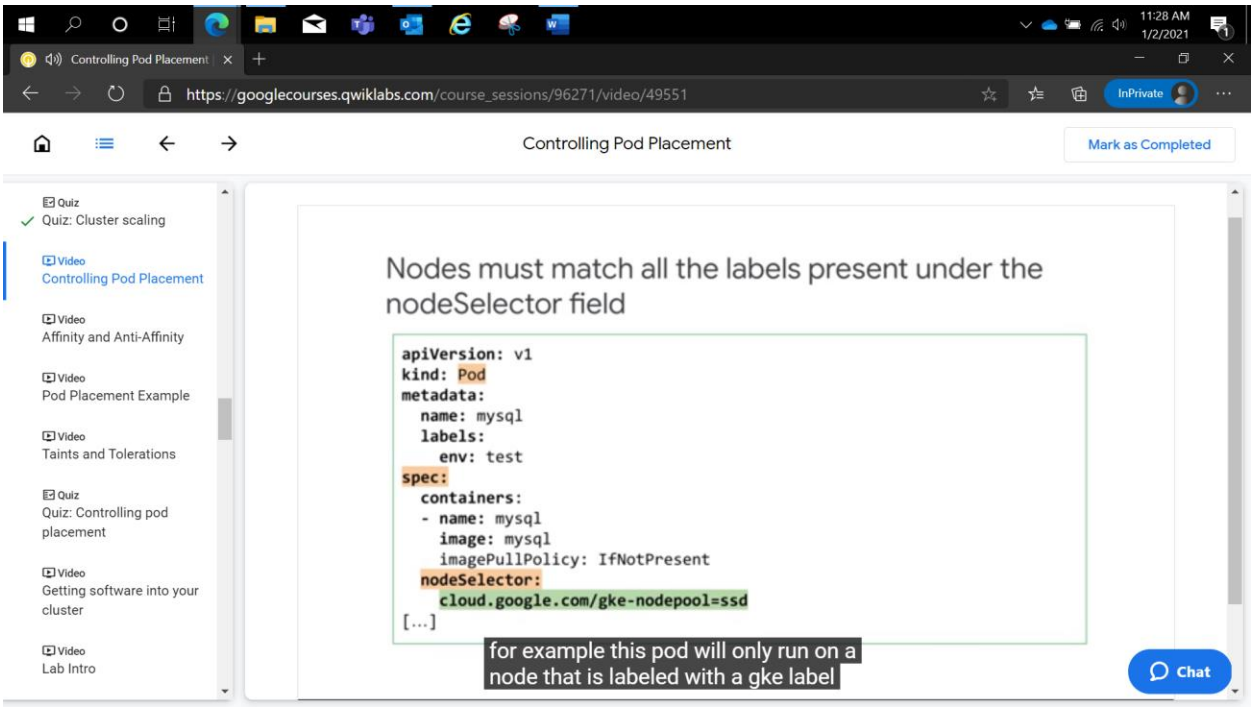

Node Pools | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49549
InPrivate
11:21 AM
1/2/2021
Node Pools
Mark as Completed
Video
Node Pools
Quiz
Quiz: Cluster scaling
Video
Controlling Pod Placement
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
gcloud commands for autoscaling
Create a cluster with autoscaling enabled
gcloud container clusters create [CLUSTER_NAME] --num-nodes 30 \
--enable-autoscaling --min-nodes 15
--max-nodes 50 [--zone COMPUTE_ZONE]
Enable autoscaling for an existing node pool
gcloud container clusters update [CLUSTER_NAME] --enable-autoscaling \
--min-nodes 1 --max-nodes 10 --zone
[COMPUTE_ZONE] --node-pool [POOL_NAME]
Add a node pool with autoscaling enabled
gcloud container node-pools create [POOL_NAME] --cluster [CLUSTER_NAME]
--enable-autoscaling --min-nodes 15
--max-nodes 50 [--zone COMPUTE_ZONE]
Disable autoscaling for an existing node pool
gcloud container clusters update [CLUSTER_NAME] --no-enable-autoscaling \
--node-pool [POOL_NAME] [--zone
[COMPUTE_ZONE] --project [PROJECT_ID]]
scaling for an existing node pool you
can also disable auto scaling
Chat

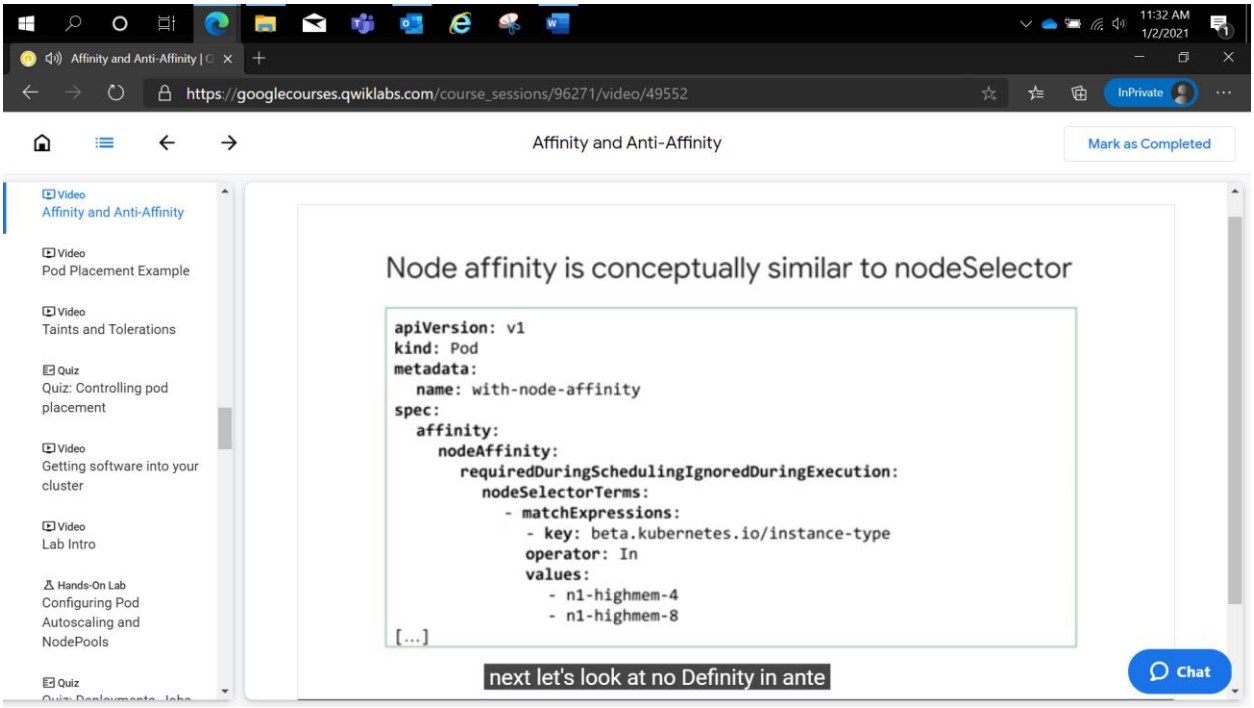

11:25 AM
1/2/2021
Quiz: Cluster scaling | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49550
InPrivate
Quiz: Cluster scaling
Quiz
Quiz: Cluster scaling
Video
Controlling Pod Placement
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
1. True or false: if you manually decrease the size of a node pool, any Pods on deleted nodes will be restarted on other nodes.
False
True
That is correct.
2. True or false: if autoscaling decreases the size of a node pool, any Pods on deleted nodes that aren't managed by a replication controller will be lost.
False
True
That is correct.
Chat

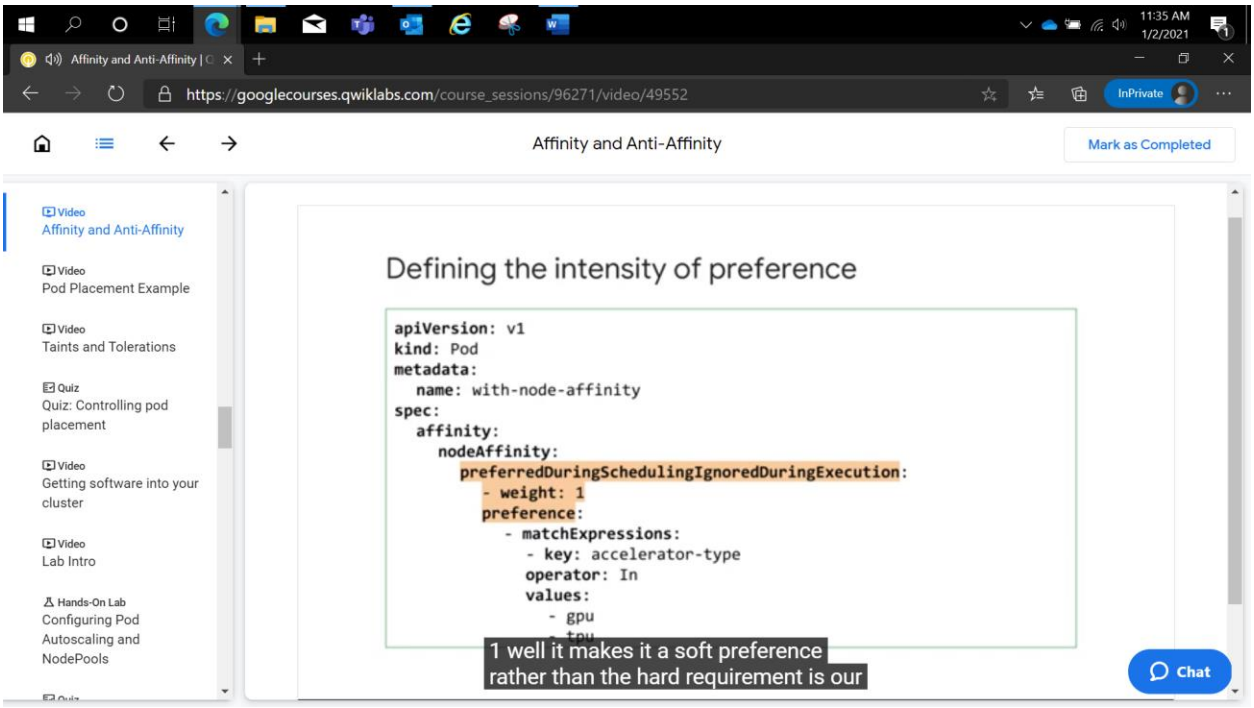

11:26 AM
1/2/2021
Quiz: Cluster scaling | Qwiklabs
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49550
InPrivate
Quiz: Cluster scaling
Quiz
Quiz: Cluster scaling
Video
Controlling Pod Placement
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
False
True
That is correct.
3. True or false: if you increase the size of a node pool, existing Pods will be moved to newer nodes.
False
True
That is correct.
Chat

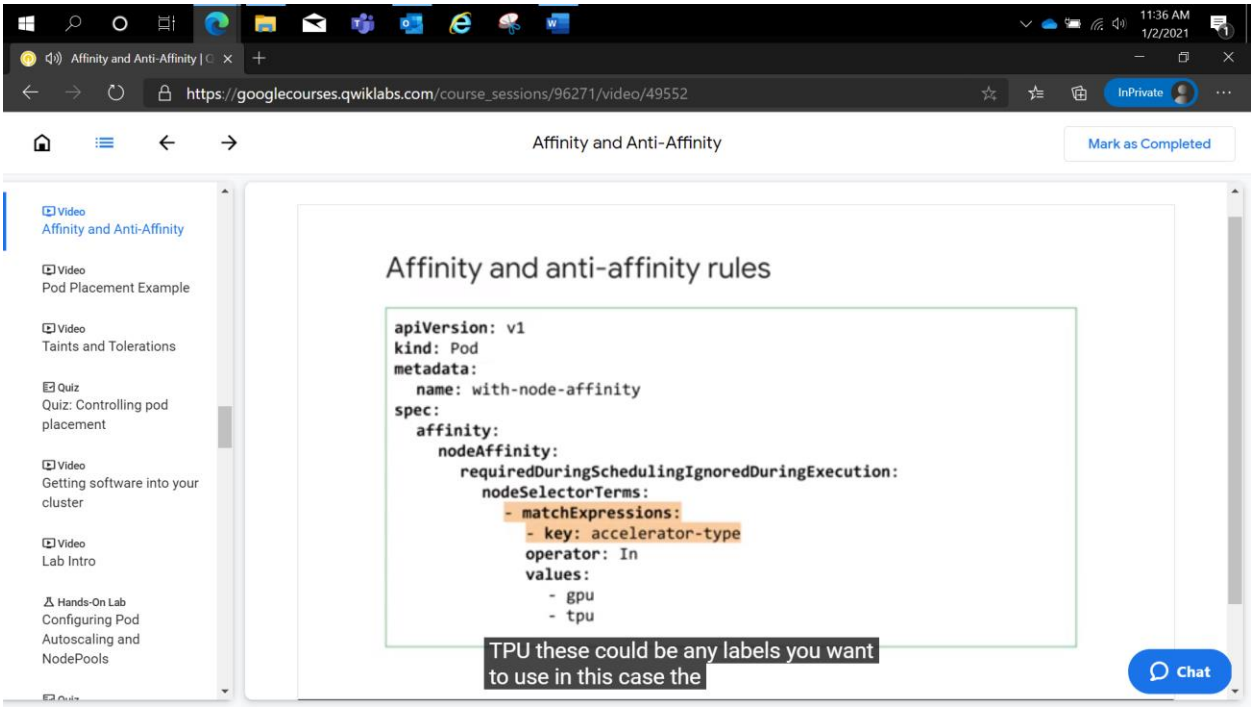

Controlling Pod Placement
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49551
InPrivate
11:27 AM
1/2/2021
Controlling Pod Placement
Mark as Completed
Quiz
Quiz: Cluster scaling
Video
Controlling Pod Placement
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Controlled scheduling
Cluster
Master
Scheduler
Node
Pod
Container
CPU limit
Memory limit
CPU request
Memory request
when nodes are started the cubelet automatically
Chat

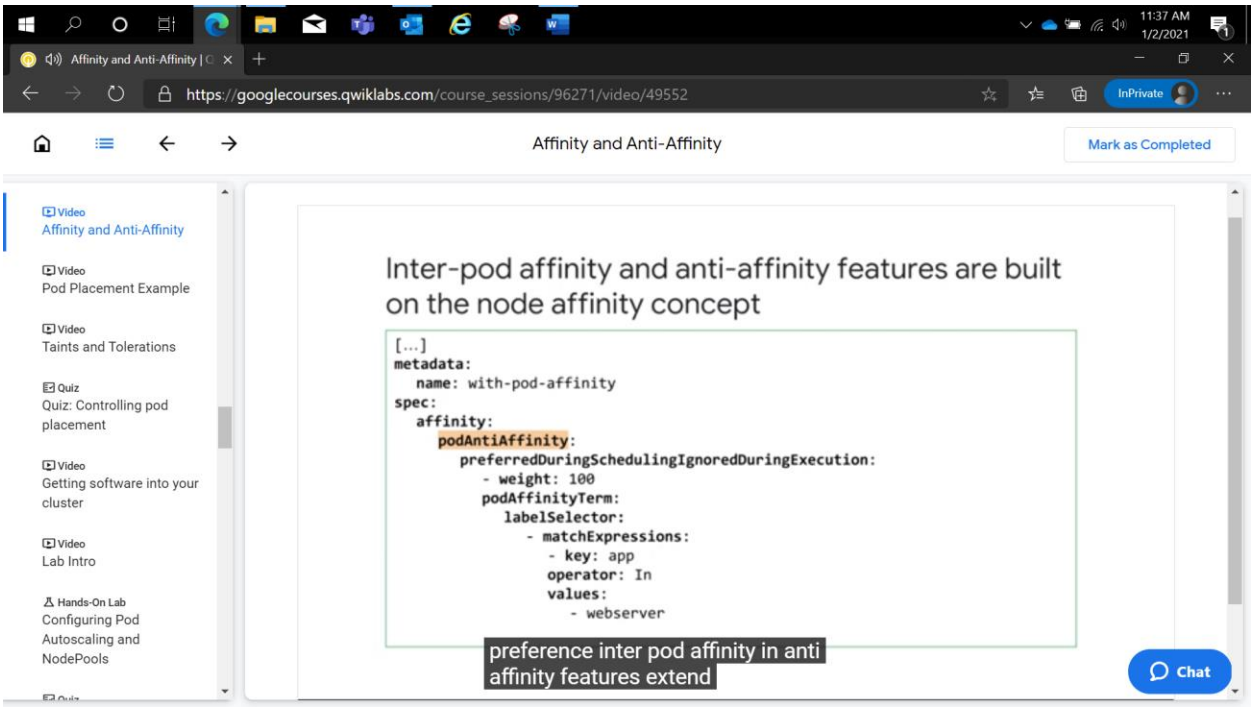

Controlling Pod Placement
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49551
InPrivate
11:28 AM
1/2/2021
Controlling Pod Placement
Mark as Completed
Quiz
Quiz: Cluster scaling
Video
Controlling Pod Placement
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Nodes must match all the labels present under the nodeSelector field
apiVersion: v1
kind: Pod
metadata:
name: nginx
labels:
env: test
spec:
containers:
- name: nginx
image: nginx
imagePullPolicy: IfNotPresent
nodeSelector:
disktype: ssd
[...]
apiVersion: v1
kind: Node
metadata:
name: node1
labels:
disktype: ssd
[...]
assigned for example the label kubernetes
Chat

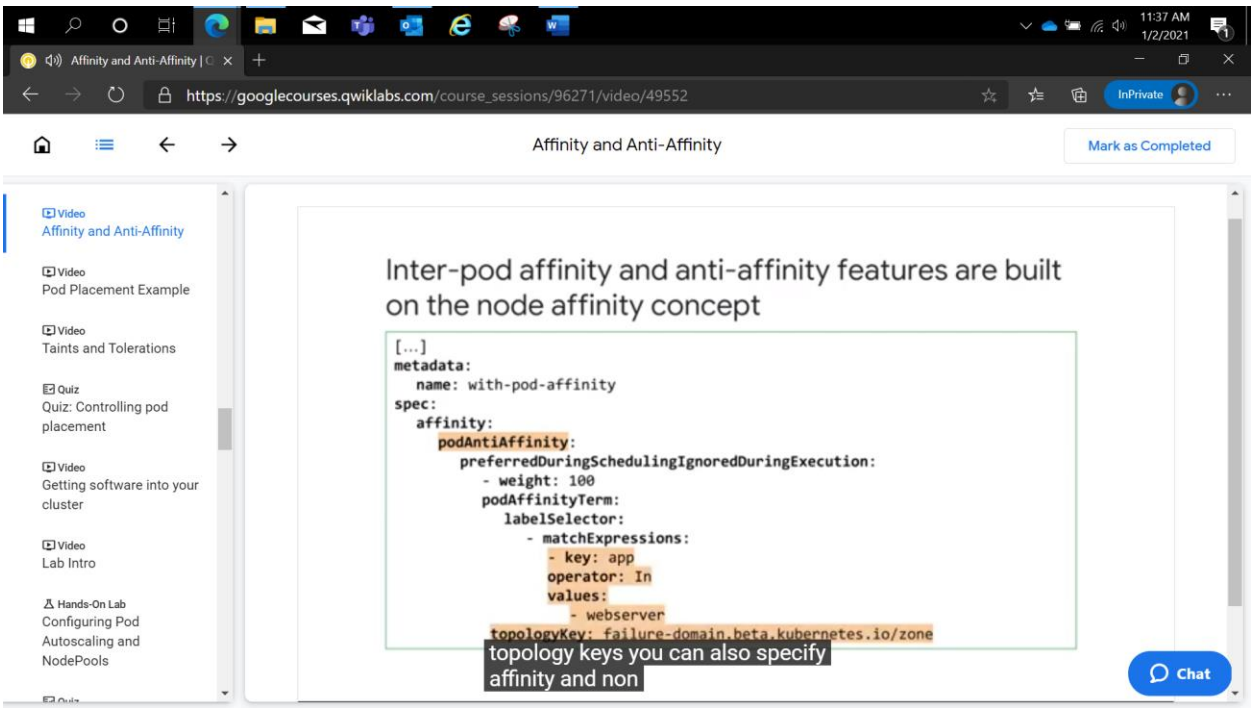

11:28 AM
1/2/2021
Controlling Pod Placement
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49551
InPrivate
Controlling Pod Placement
Mark as Completed
Quiz
Quiz: Cluster scaling
Video
Controlling Pod Placement
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Nodes must match all the labels present under the nodeSelector field
apiVersion: v1
kind: Pod
metadata:
name: mysql
labels:
env: test
spec:
containers:
- name: mysql
image: mysql
imagePullPolicy: IfNotPresent
nodeSelector:
cloud.google.com/gke-nodepool=ssd
[...]
for example this pod will only run on a node that is labeled with a gke label
Chat

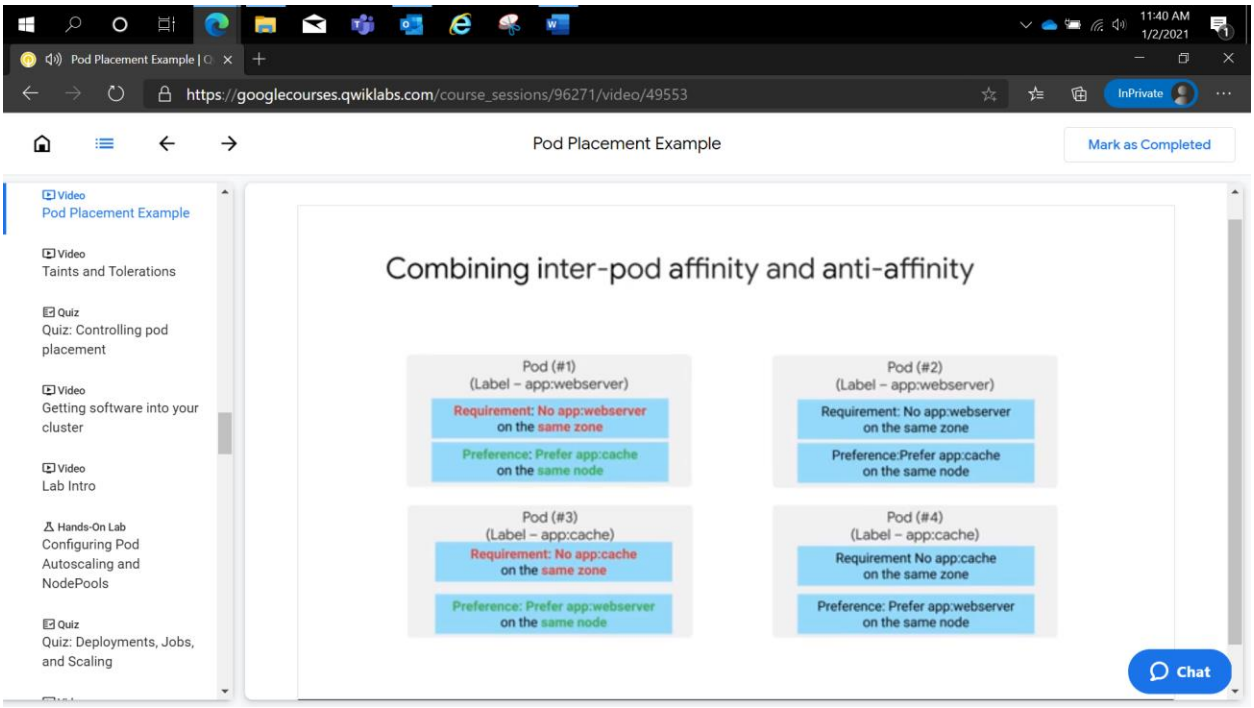

11:32 AM
1/2/2021
Affinity and Anti-Affinity
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49552
InPrivate
Affinity and Anti-Affinity
Mark as Completed
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Node affinity is conceptually similar to nodeSelector
apiVersion: v1
kind: Pod
metadata:
name: with-node-affinity
spec:
affinity:
nodeAffinity:
requiredDuringSchedulingIgnoredDuringExecution:
nodeSelectorTerms:
- matchExpressions:
- key: beta.kubernetes.io/instance-type
operator: In
values:
- n1-highmem-4
- n1-highmem-8
[...]
next let's look at no Definity in ante
Chat

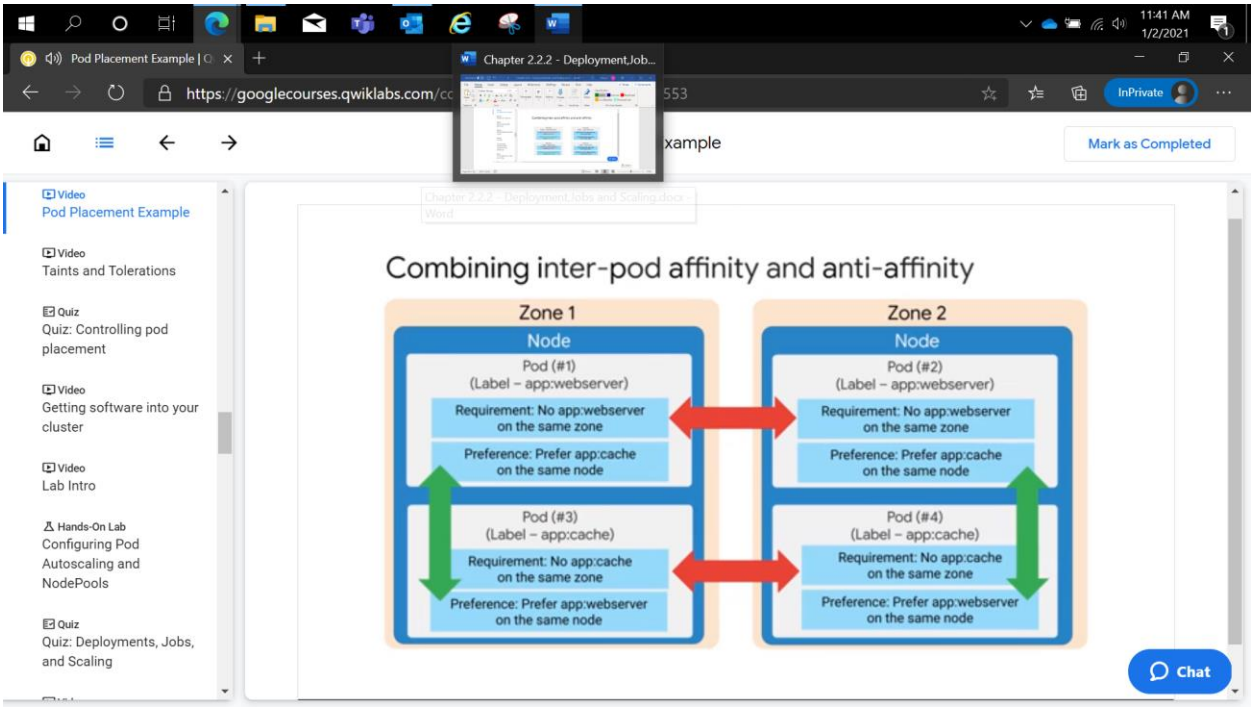

11:35 AM
1/2/2021
Affinity and Anti-Affinity | C
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49552
InPrivate
Affinity and Anti-Affinity
Mark as Completed
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Defining the intensity of preference
apiVersion: v1
kind: Pod
metadata:
name: with-node-affinity
spec:
affinity:
nodeAffinity:
preferredDuringSchedulingIgnoredDuringExecution:
- weight: 1
preference:
- matchExpressions:
- key: accelerator-type
operator: In
values:
- gpu
1 well it makes it a soft preference
rather than the hard requirement is our
Chat

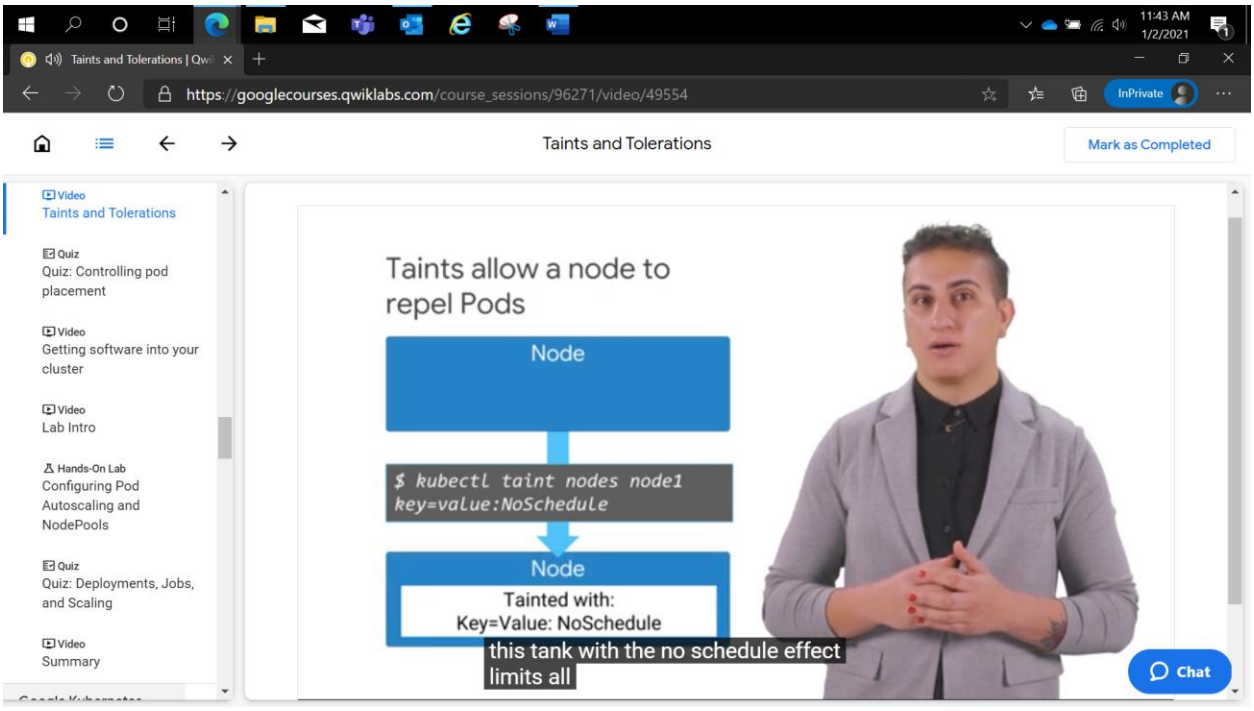

11:36 AM
1/2/2021
Affinity and Anti-Affinity | C
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49552
InPrivate
Affinity and Anti-Affinity
Mark as Completed
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Affinity and anti-affinity rules
apiVersion: v1
kind: Pod
metadata:
name: with-node-affinity
spec:
affinity:
nodeAffinity:
requiredDuringSchedulingIgnoredDuringExecution:
nodeSelectorTerms:
- matchExpressions:
- key: accelerator-type
operator: In
values:
- gpu
- tpu
TPU these could be any labels you want
to use in this case the
Chat

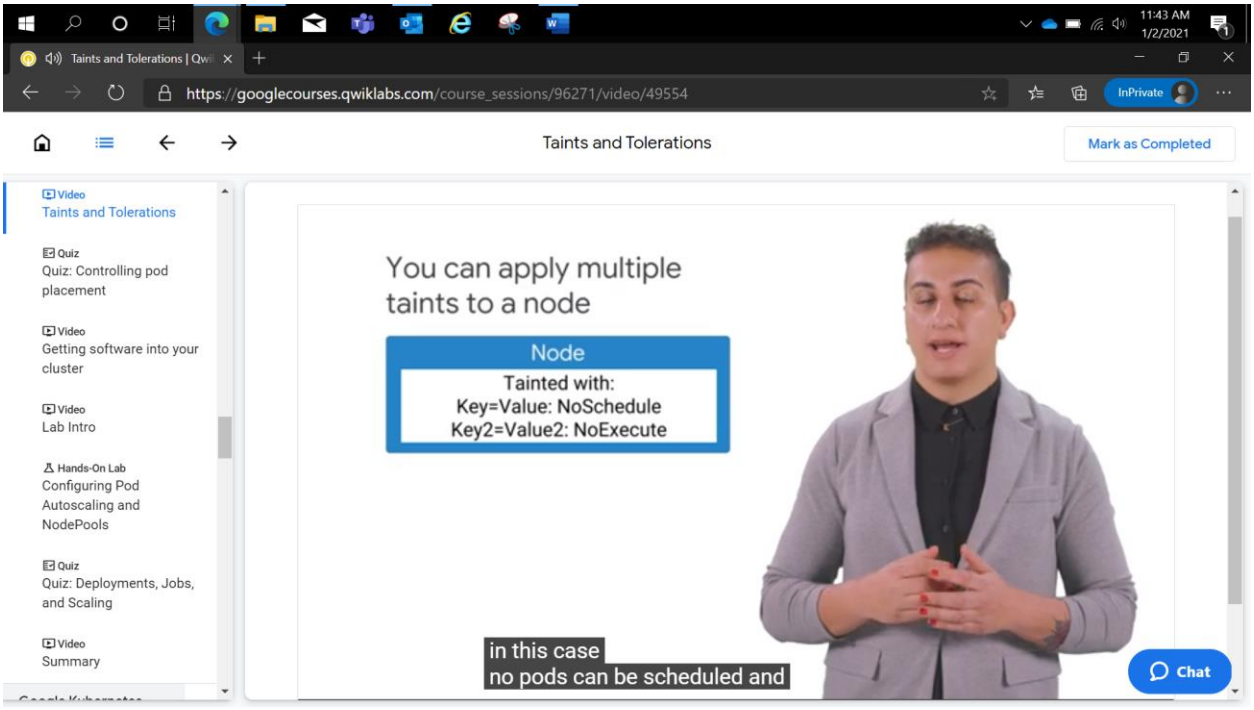

Affinity and Anti-Affinity | C
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49552
InPrivate
11:37 AM
1/2/2021
Affinity and Anti-Affinity
Mark as Completed
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Inter-pod affinity and anti-affinity features are built on the node affinity concept
[...]
metadata:
name: with-pod-affinity
spec:
affinity:
podAntiAffinity:
preferredDuringSchedulingIgnoredDuringExecution:
- weight: 100
podAffinityTerm:
labelSelector:
- matchExpressions:
- key: app
operator: In
values:
- webserver
preference inter pod affinity in anti affinity features extend
Chat

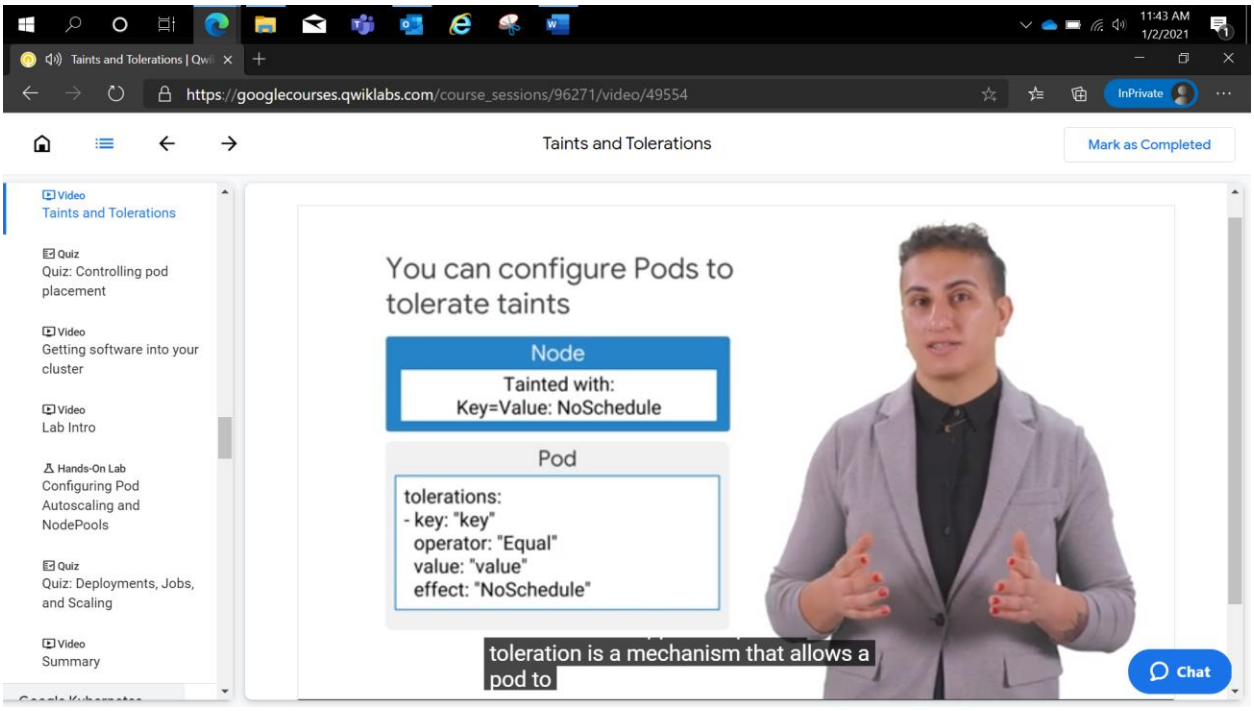

Affinity and Anti-Affinity | C
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49552
InPrivate
11:37 AM
1/2/2021
Affinity and Anti-Affinity
Mark as Completed
Video
Affinity and Anti-Affinity
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Inter-pod affinity and anti-affinity features are built on the node affinity concept
[...]
metadata:
name: with-pod-affinity
spec:
affinity:
podAntiAffinity:
preferredDuringSchedulingIgnoredDuringExecution:
- weight: 100
podAffinityTerm:
labelSelector:
- matchExpressions:
- key: app
operator: In
values:
- webserver
topologyKey: failure-domain.beta.kubernetes.io/zone
topology keys you can also specify affinity and non
Chat

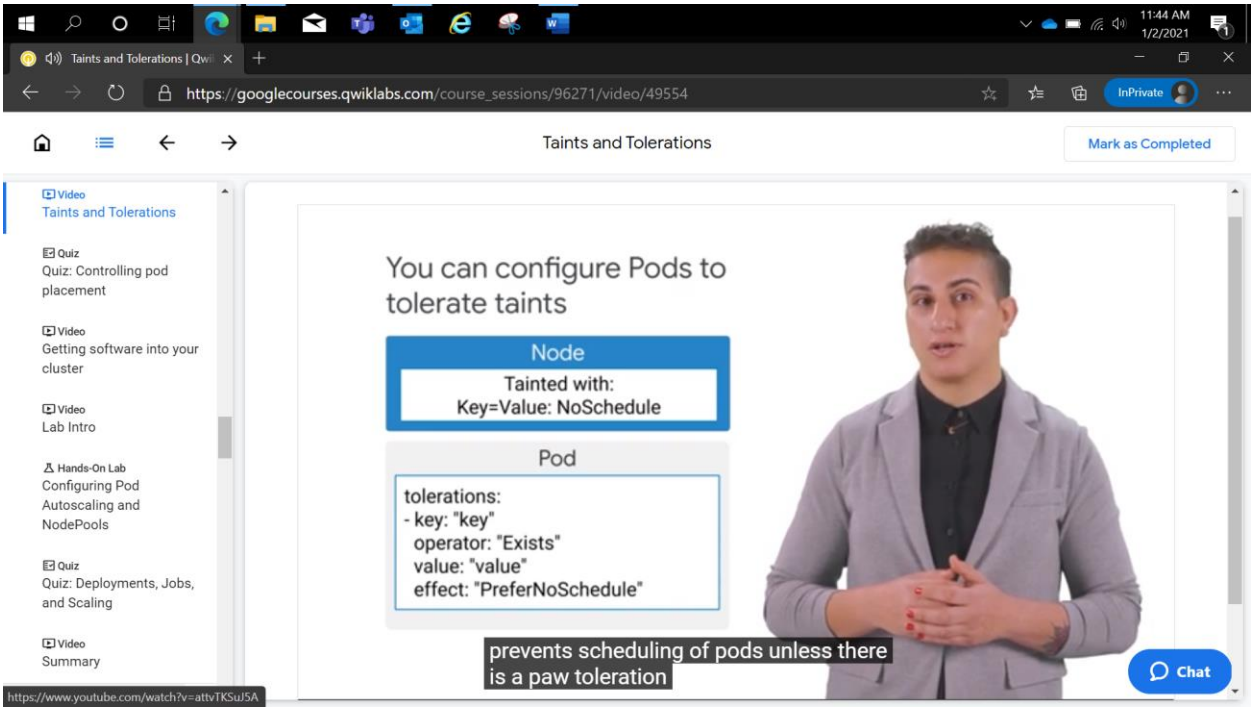

11:40 AM
1/2/2021
Pod Placement Example | C
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49553
InPrivate
Pod Placement Example
Mark as Completed
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Quiz: Deployments, Jobs, and Scaling
Combining inter-pod affinity and anti-affinity
Pod (#1)
(Label – app:webserver)
Requirement: No app:webserver on the same zone
Preference: Prefer app:cache on the same node
Pod (#2)
(Label – app:webserver)
Requirement: No app:webserver on the same zone
Preference:Prefer app:cache on the same node
Pod (#3)
(Label – app:cache)
Requirement: No app:cache on the same zone
Preference: Prefer app:webserver on the same node
Pod (#4)
(Label – app:cache)
Requirement No app:cache on the same zone
Preference: Prefer app:webserver on the same node
Chat

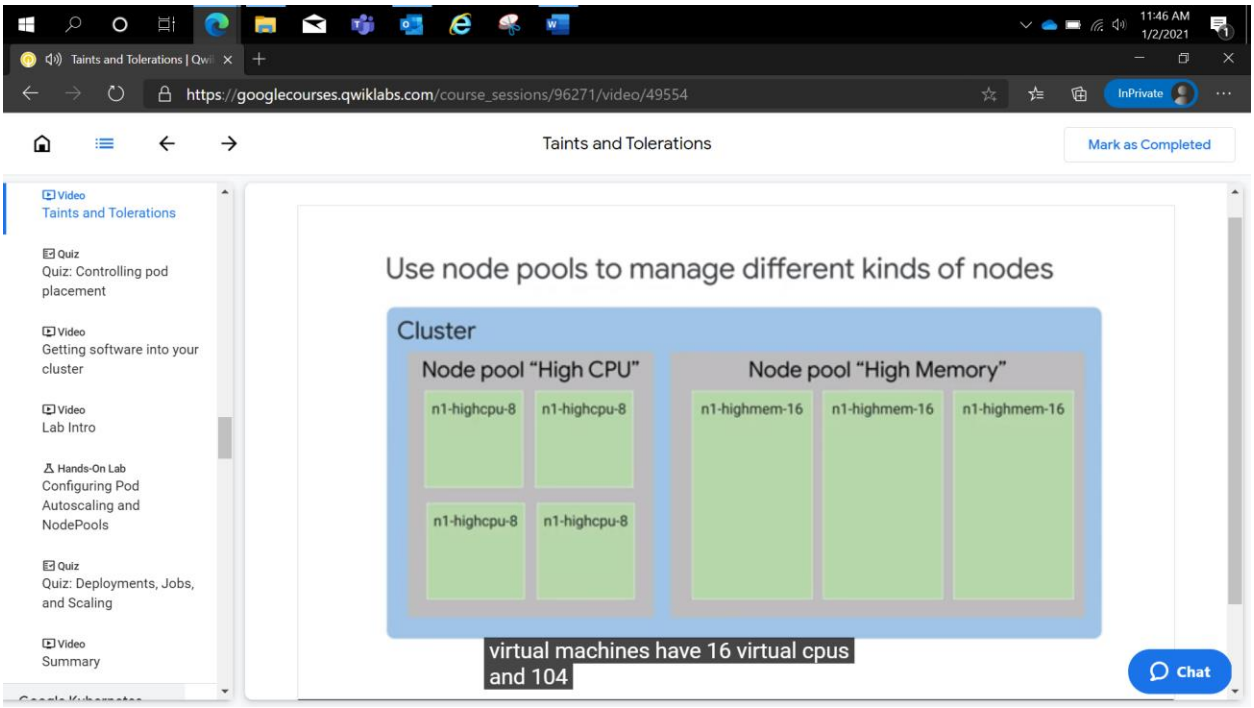

11:41 AM
1/2/2021
Pod Placement Example | C
Chapter 2.2.2 - Deployment,Job...
https://googlecourses.qwiklabs.com/cc
553
InPrivate
xample
Mark as Completed
Video
Pod Placement Example
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Quiz: Deployments, Jobs, and Scaling
Combining inter-pod affinity and anti-affinity
Zone 1
Node
Pod (#1)
(Label – app:webserver)
Requirement: No app:webserver on the same zone
Preference: Prefer app:cache on the same node
Pod (#3)
(Label – app:cache)
Requirement: No app:cache on the same zone
Preference: Prefer app:webserver on the same node
Zone 2
Node
Pod (#2)
(Label – app:webserver)
Requirement: No app:webserver on the same zone
Preference: Prefer app:cache on the same node
Pod (#4)
(Label – app:cache)
Requirement: No app:cache on the same zone
Preference: Prefer app:webserver on the same node
Chat

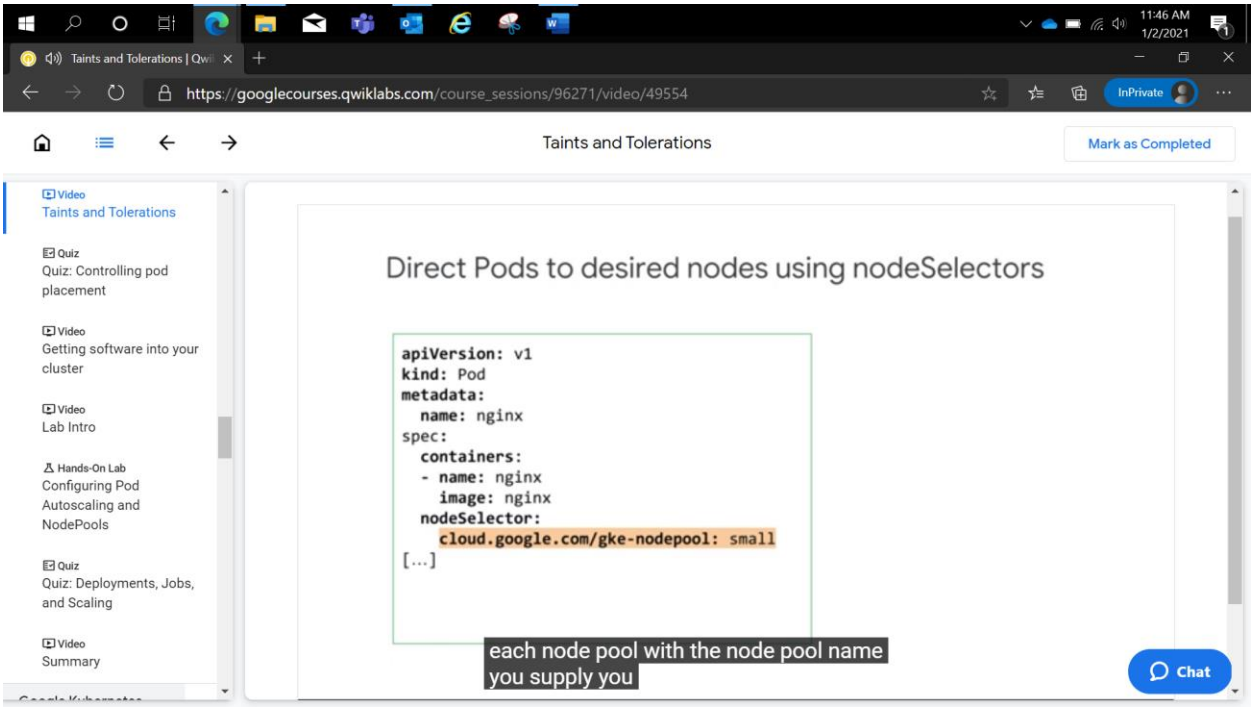

Taints and Tolerations | Qwi
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49554
InPrivate
11:43 AM
1/2/2021
Taints and Tolerations
Mark as Completed
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Taints allow a node to repel Pods
Node
$ kubectl taint nodes node1 key=value:NoSchedule
Node
Tainted with:
Key=Value: NoSchedule
this tank with the no schedule effect
limits all
Chat

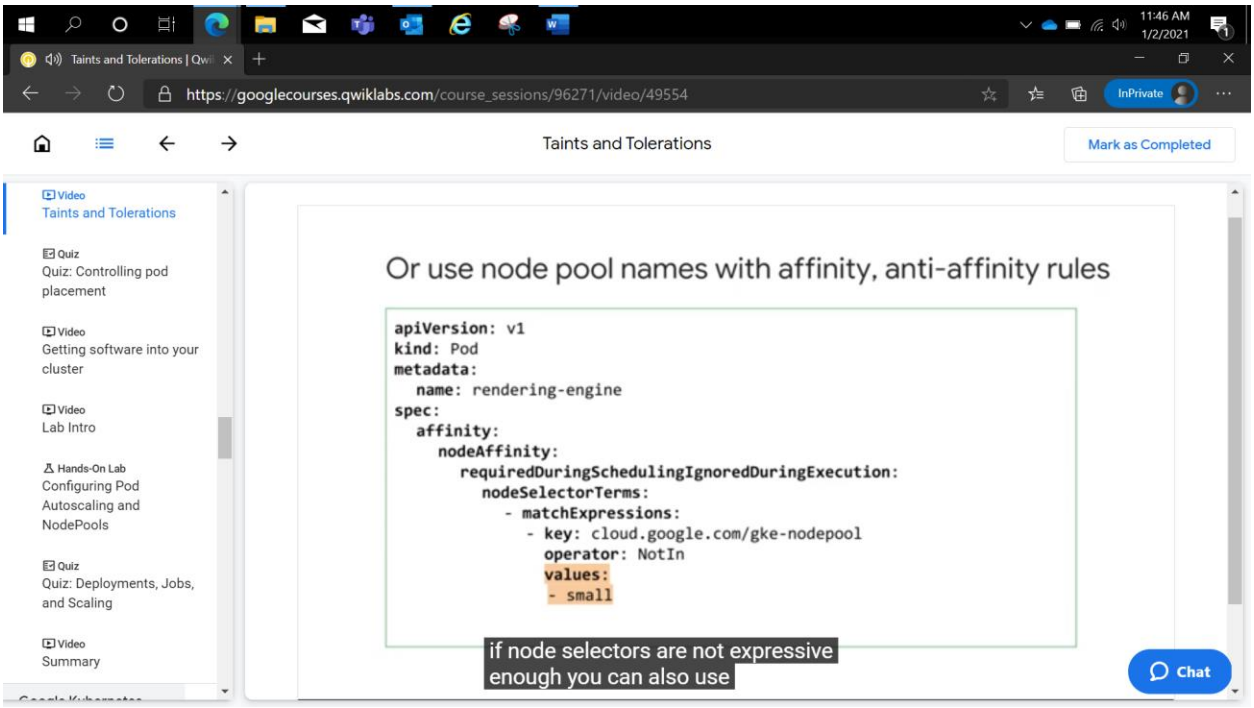

Taints and Tolerations | Qwi
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49554
InPrivate
11:43 AM
1/2/2021
Taints and Tolerations
Mark as Completed
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
You can apply multiple taints to a node
Node
Tainted with:
Key=Value: NoSchedule
Key2=Value2: NoExecute
in this case
no pods can be scheduled and
Chat

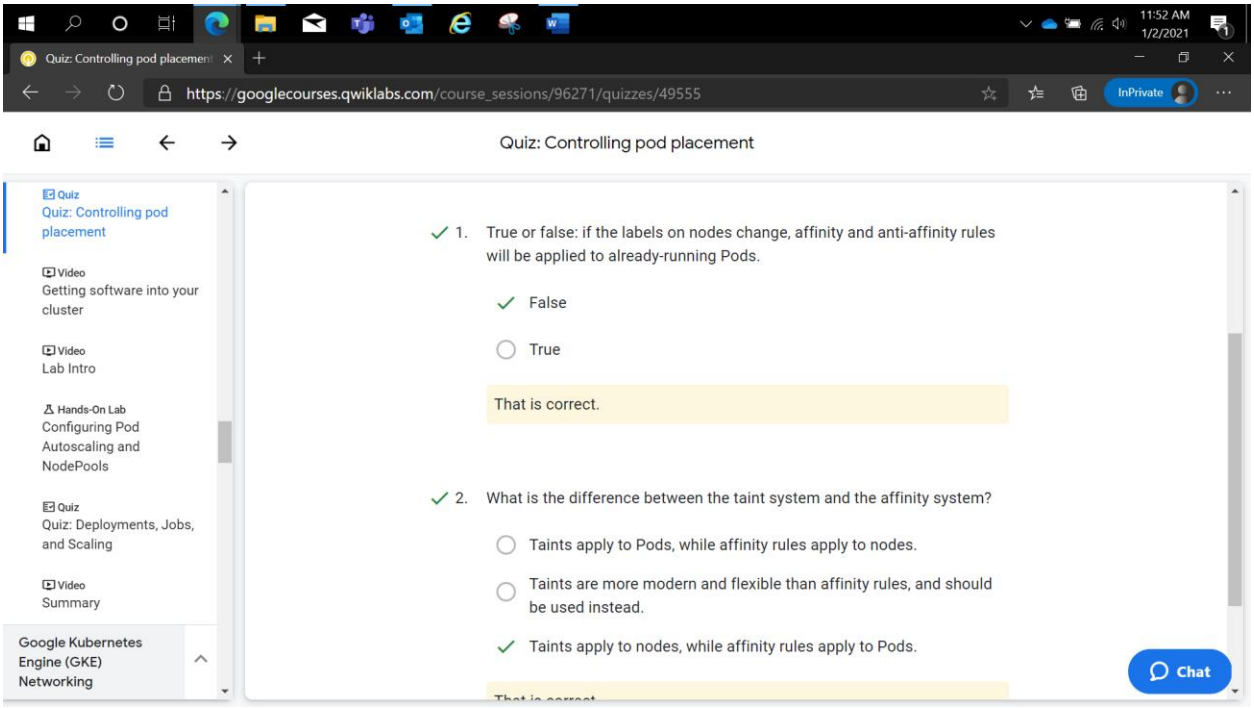

11:43 AM
1/2/2021
Taints and Tolerations | Qwi
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49554
InPrivate
Taints and Tolerations
Mark as Completed
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
You can configure Pods to tolerate taints
Node
Tainted with:
Key=Value: NoSchedule
Pod
tolerations:
- key: "key"
operator: "Equal"
value: "value"
effect: "NoSchedule"
toleration is a mechanism that allows a pod to
Chat

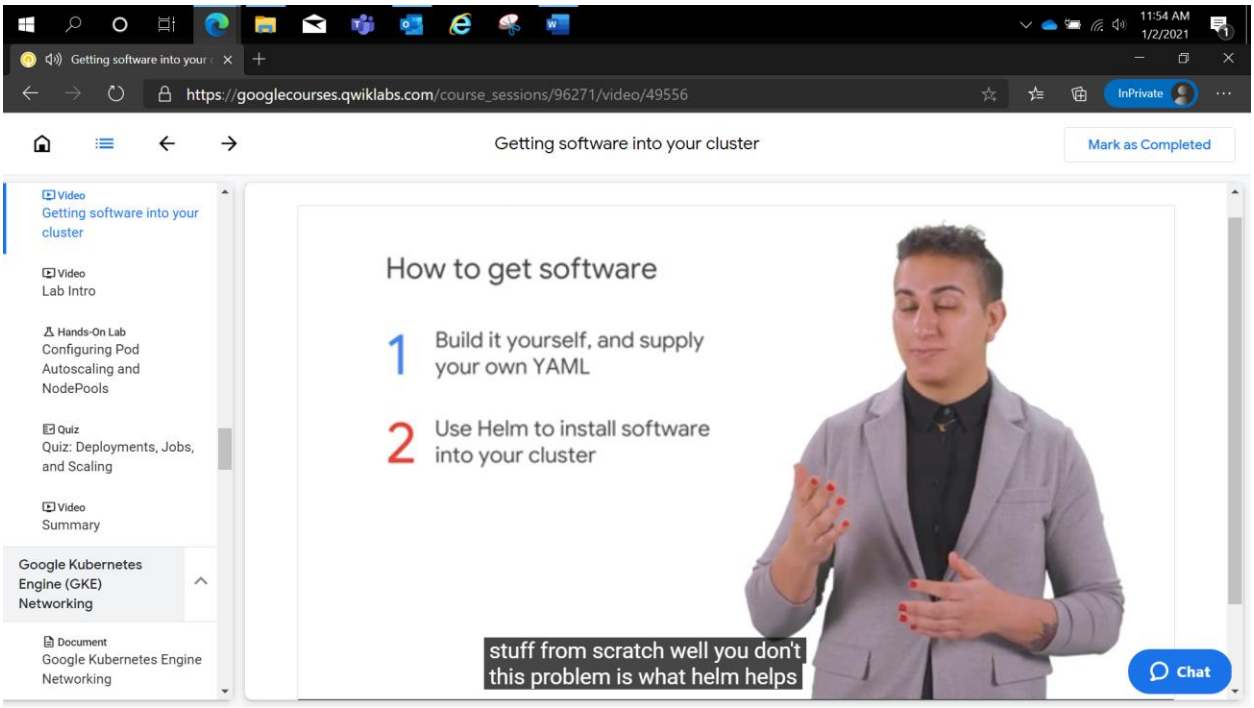

11:44 AM
1/2/2021
Taints and Tolerations | Qwi
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49554
InPrivate
Taints and Tolerations
Mark as Completed
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
You can configure Pods to tolerate taints
Node
Tainted with:
Key=Value: NoSchedule
Pod
tolerations:
- key: "key"
operator: "Exists"
value: "value"
effect: "PreferNoSchedule"
prevents scheduling of pods unless there is a paw toleration
Chat
https://www.youtube.com/watch?v=attvTKSuJ5A

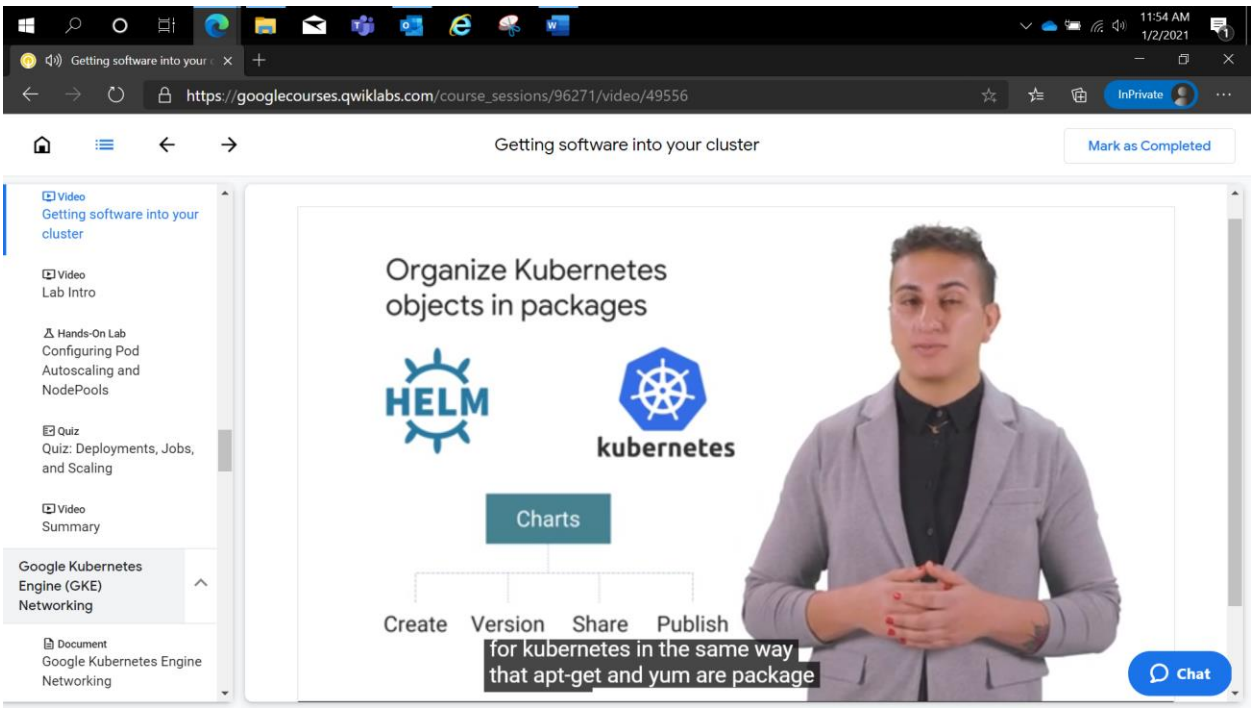

Taints and Tolerations | Qwi
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49554
InPrivate
11:46 AM
1/2/2021
Taints and Tolerations
Mark as Completed
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Use node pools to manage different kinds of nodes
Cluster
Node pool "High CPU"
n1-highcpu-8
n1-highcpu-8
n1-highcpu-8
n1-highcpu-8
Node pool "High Memory"
n1-highmem-16
n1-highmem-16
n1-highmem-16
virtual machines have 16 virtual cpus and 104
Chat

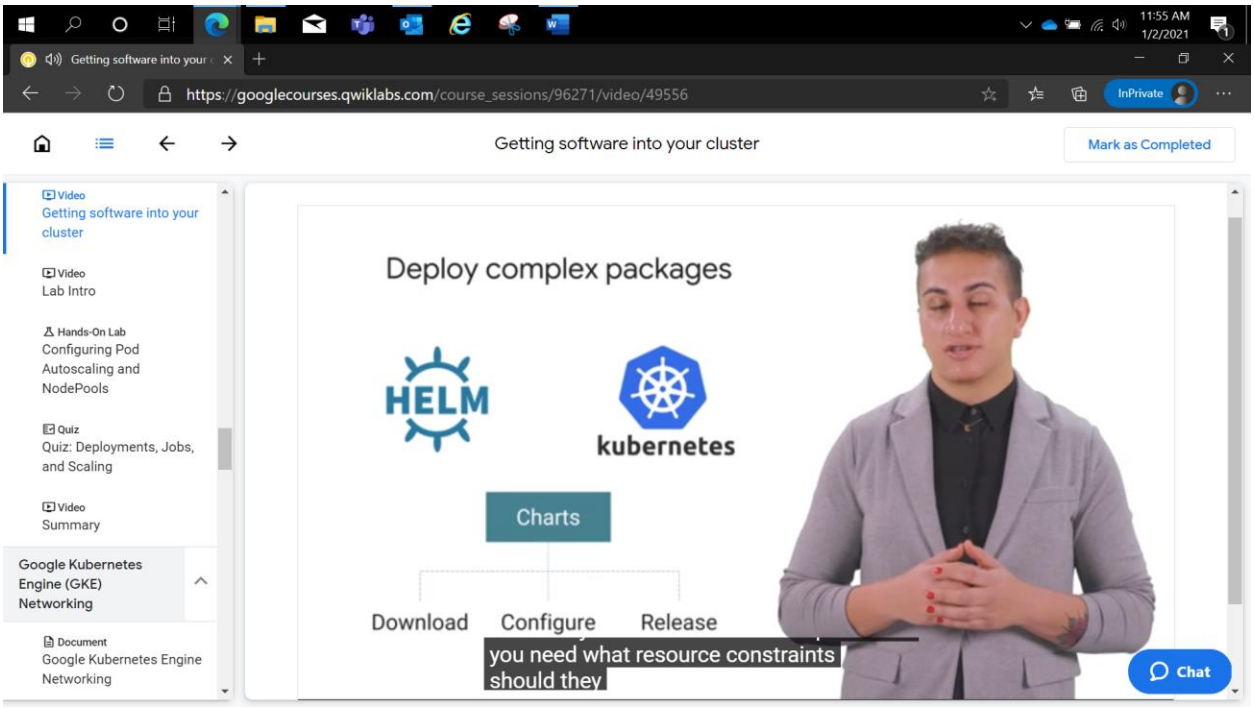

Taints and Tolerations | Qwi
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49554
InPrivate
11:46 AM
1/2/2021
Taints and Tolerations
Mark as Completed
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Direct Pods to desired nodes using nodeSelectors
apiVersion: v1
kind: Pod
metadata:
name: nginx
spec:
containers:
- name: nginx
image: nginx
nodeSelector:
cloud.google.com/gke-nodepool: small
[...]
each node pool with the node pool name you supply you
Chat

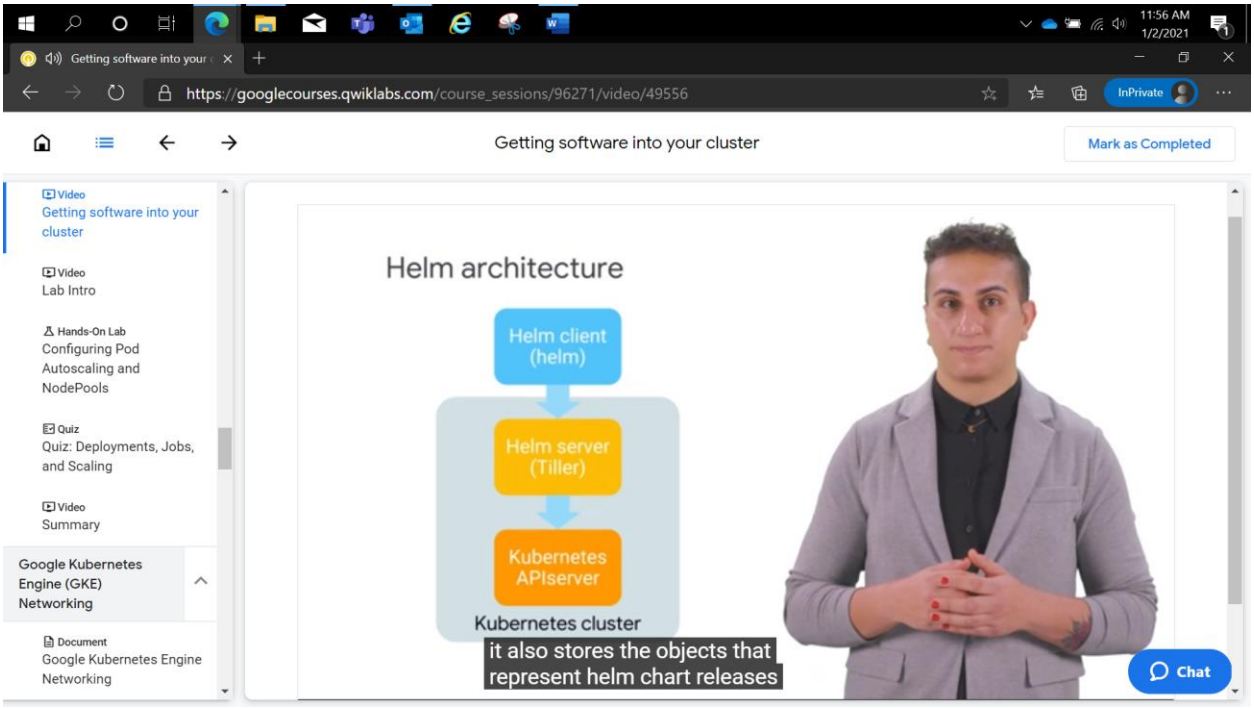

Taints and Tolerations | Qwi
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49554
InPrivate
11:46 AM
1/2/2021
Taints and Tolerations
Mark as Completed
Video
Taints and Tolerations
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Or use node pool names with affinity, anti-affinity rules
apiVersion: v1
kind: Pod
metadata:
name: rendering-engine
spec:
affinity:
nodeAffinity:
requiredDuringSchedulingIgnoredDuringExecution:
nodeSelectorTerms:
- matchExpressions:
- key: cloud.google.com/gke-nodepool
operator: NotIn
values:
- small
if node selectors are not expressive enough you can also use
Chat

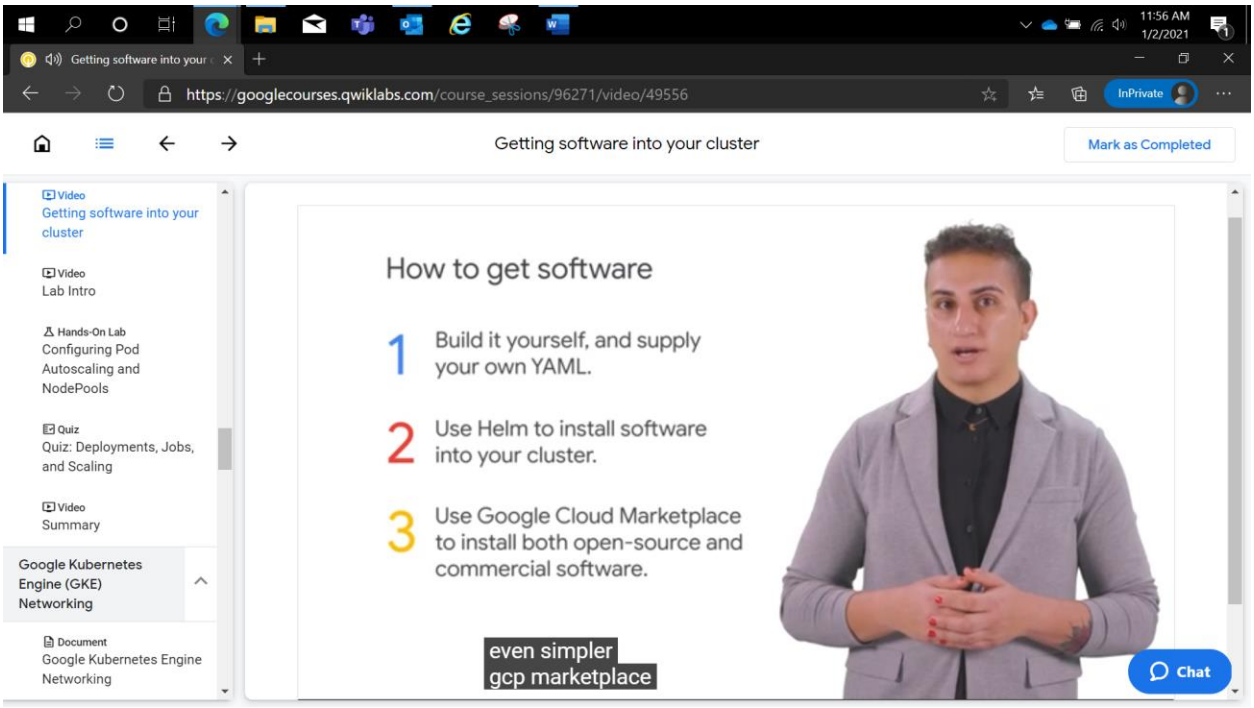

Quiz: Controlling pod placement
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49555
InPrivate
11:52 AM
1/2/2021
Quiz: Controlling pod placement
Quiz
Quiz: Controlling pod placement
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
1. True or false: if the labels on nodes change, affinity and anti-affinity rules will be applied to already-running Pods.
False
True
That is correct.
2. What is the difference between the taint system and the affinity system?
Taints apply to Pods, while affinity rules apply to nodes.
Taints are more modern and flexible than affinity rules, and should be used instead.
Taints apply to nodes, while affinity rules apply to Pods.
Chat


Getting software into your
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49556
InPrivate
11:54 AM
1/2/2021
Getting software into your cluster
Mark as Completed
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
How to get software
1 Build it yourself, and supply your own YAML
2 Use Helm to install software into your cluster
stuff from scratch well you don't
this problem is what helm helps
Chat


Getting software into your
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49556
InPrivate
11:54 AM
1/2/2021
Getting software into your cluster
Mark as Completed
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Organize Kubernetes objects in packages
HELM
kubernetes
Charts
Create
Version
Share
Publish
for kubernetes in the same way
that apt-get and yum are package
Chat


11:55 AM
1/2/2021
Getting software into your
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49556
InPrivate
Getting software into your cluster
Mark as Completed
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Deploy complex packages
HELM
kubernetes
Charts
Download
Configure
Release
you need what resource constraints should they
Chat


11:56 AM
1/2/2021
Getting software into your
https://googlecourses.qwiklabs.com/course_sessions/96271/video/49556
InPrivate
Getting software into your cluster
Mark as Completed
Video
Getting software into your cluster
Video
Lab Intro
Hands-On Lab
Configuring Pod Autoscaling and NodePools
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Helm architecture
Helm client (helm)
Helm server (Tiller)
Kubernetes APIserver
Kubernetes cluster
it also stores the objects that represent helm chart releases
Chat



Configuring Pod Autoscaling and NodePools

1 hourFree

Rate Lab

Overview

In this lab, you set up an application in Google Kubernetes Engine (GKE), and then use a HorizontalPodAutoscaler to autoscale the web application. You then work with multiple node pools of different types, and you apply taints and tolerations to control the scheduling of Pods with respect to the underlying node pool.

Objectives

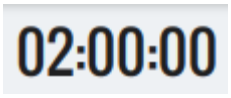
In this lab, you learn how to perform the following tasks:

- Configure autoscaling and HorizontalPodAutoscaler
- Add a node pool and configure taints on the nodes for Pod anti-affinity
- Configure an exception for the node taint by adding a toleration to a Pod's manifest

Task 0. Lab Setup

Access Qwiklabs

For each lab, you get a new GCP project and set of resources for a fixed time at no cost.

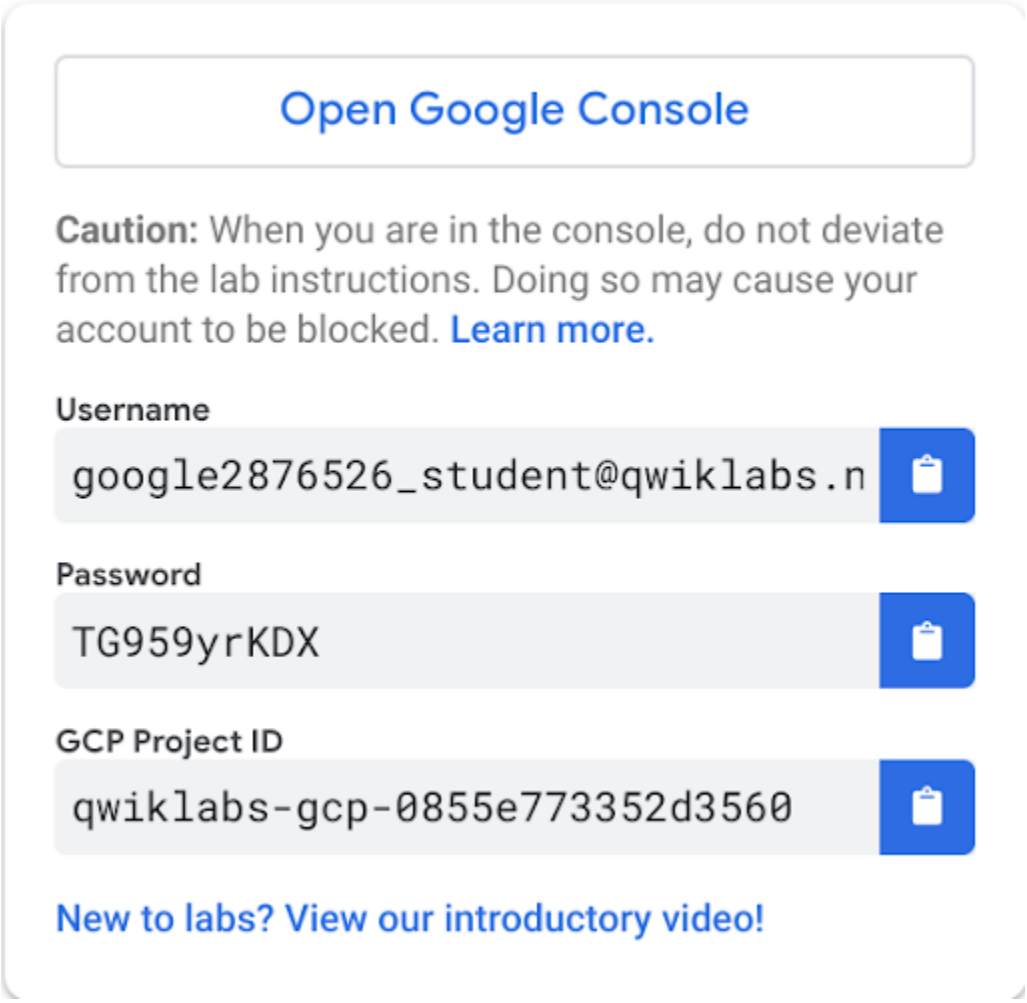
1. Make sure you signed into Qwiklabs using an **incognito window**.
2. Note the lab's access time (for example, 

and make sure you can finish in that time block.

There is no pause feature. You can restart if needed, but you have to start at the beginning.

3. When ready, click 

.
4. Note your lab credentials. You will use them to sign in to Cloud Platform Console.



5. Click **Open Google Console**.

6. Click **Use another account** and copy/paste credentials for **this** lab into the prompts.

If you use other credentials, you'll get errors or **incur charges**.

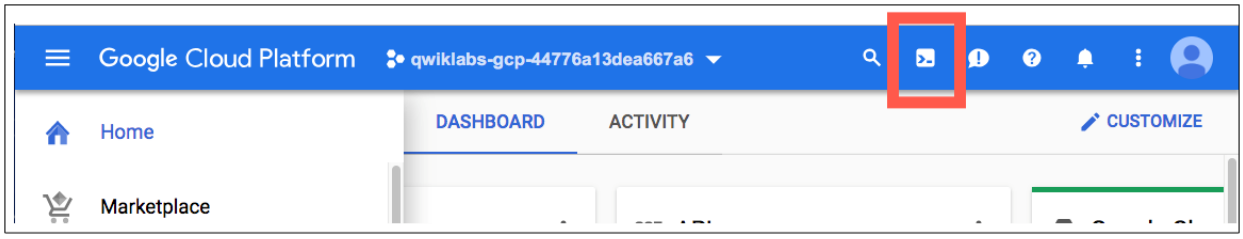
7. Accept the terms and skip the recovery resource page.

Do not click **End Lab** unless you are finished with the lab or want to restart it. This clears your work and removes the project.
After you complete the initial sign-in steps, the project dashboard appears.

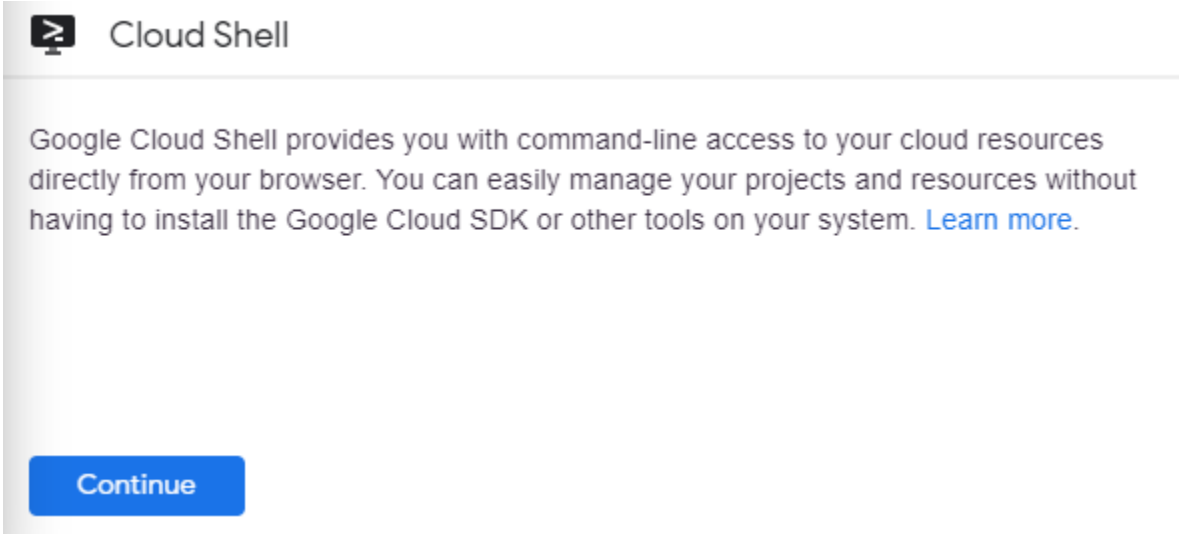
Activate Google Cloud Shell

Google Cloud Shell is a virtual machine that is loaded with development tools. It offers a persistent 5GB home directory and runs on the Google Cloud. Google Cloud Shell provides command-line access to your GCP resources.

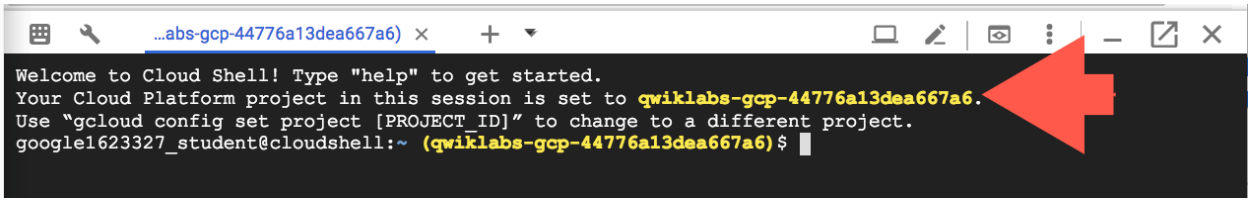
1. In GCP console, on the top right toolbar, click the Open Cloud Shell button.



2. Click **Continue**.



It takes a few moments to provision and connect to the environment. When you are connected, you are already authenticated, and the project is set to your *PROJECT_ID*. For example:



gcloud is the command-line tool for Google Cloud Platform. It comes pre-installed on Cloud Shell and supports tab-completion.

You can list the active account name with this command:

```
gcloud auth list
```
content_copy

Output:

```
Credentialed accounts:
 - <myaccount>@<mydomain>.com (active)
```
content_copy

Example output:

```
Credentialed accounts:
 - google1623327_student@qwiklabs.net
```
content_copy

You can list the project ID with this command:

```
gcloud config list project
```

```
content_copy
```

Output:

```
[core]
project = <project_ID>content_copy
```

Example output:

```
[core]
project = qwiklabs-gcp-44776a13dea667a6content_copy
```

Full documentation of **gcloud** is available on Google Cloud gcloud Overview .

Task 1. Connect to the lab GKE cluster and deploy a sample workload

In this task, you connect to the lab GKE cluster and create a deployment manifest for a set of Pods within the cluster.

Connect to the lab GKE cluster

1. In Cloud Shell, type the following command to set the environment variable for the zone and cluster name.

```
export my_zone=us-central1-a
export my_cluster=standard-cluster-1
content_copy
```

2. Configure tab completion for the kubectl command-line tool.

```
source <(kubectl completion bash)
content_copy
```

3. Configure access to your cluster for kubectl:

```
gcloud container clusters get-credentials $my_cluster --zone $my_zone
content_copy
```

Deploy a sample web application to your GKE cluster

You will deploy a sample application to your cluster using the `web.yaml` deployment file that has been created for you:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: web
spec:
  replicas: 1
  selector:
    matchLabels:
      run: web
  template:
    metadata:
      labels:
        run: web
    spec:
      containers:
      - image: gcr.io/google-samples/hello-app:1.0
        name: web
        ports:
        - containerPort: 8080
          protocol: TCP
```
content_copy

This manifest creates a deployment using a sample web application container image that listens on an HTTP server on port 8080.

1. In Cloud Shell enter the following command to clone the repository to the lab Cloud Shell.

```
git clone https://github.com/GoogleCloudPlatform/training-data-analyst
```
content_copy

2. Create a soft link as a shortcut to the working directory.

```
ln -s ~/training-data-analyst/courses/ak8s/v1.1 ~/ak8s
```
content_copy

3. Change to the directory that contains the sample files for this lab.

```
cd ~/ak8s/Autoscaling/
```
content_copy

4. To create a deployment from this file, execute the following command:

```
kubectl create -f web.yaml --save-config
```
content_copy

5. Create a service resource of type NodePort on port 8080 for the web deployment.

```
kubectl expose deployment web --target-port=8080 --type=NodePort
```
content_copy

6. Verify that the service was created and that a node port was allocated:

```
kubectl get service web
```
content_copy

Your IP address and port number might be different from the example output.

Output (do not copy)

```
NAME      TYPE       CLUSTER-IP      EXTERNAL-IP   PORT(S)          AGE
web       NodePort   10.11.246.185   <none>        8080:32056/TCP   12s
```
content_copy

Click *Check my progress* to verify the objective.

Deploy a sample web application to GKE cluster

Check my progress

Task 2. Configure autoscaling on the cluster

In this task, you configure the cluster to automatically scale the sample application that you deployed earlier.

Configure autoscaling

1. Get the list of deployments to determine whether your sample web application is still running.

```
kubectl get deployment
content_copy
```

The output should look like the example.

Output (do not copy)

```
NAME    READY   UP-TO-DATE   AVAILABLE   AGE
web     1/1     1            1           5m48s
content_copy
```

If the web deployment of your application is not displayed, return to task 1 and redeploy it to the cluster.

2. To configure your sample application for autoscaling (and to set the maximum number of replicas to four and the minimum to one, with a CPU utilization target of 1%), execute the following command:

```
kubectl autoscale deployment web --max 4 --min 1 --cpu-percent 1
content_copy
```

When you use `kubectl autoscale`, you specify a maximum and minimum number of replicas for your application, as well as a CPU utilization target.

3. Get the list of deployments to verify that there is still only one deployment of the web application.

```
kubectl get deployment
content_copy
```

Output (do not copy)

```
NAME    READY   UP-TO-DATE   AVAILABLE   AGE
web     1/1     1            1           8m21s
content_copy
```

Inspect the HorizontalPodAutoscaler object

The `kubectl autoscale` command you used in the previous task creates a `HorizontalPodAutoscaler` object that targets a specified resource, called the]scale target, and scales it as needed. The autoscaler periodically adjusts the number of replicas of the scale target to match the average CPU utilization that you specify when creating the autoscaler.

1. To get the list of HorizontalPodAutoscaler resources, execute the following command:

```
kubectl get hpa
```
content_copy

The output should look like the example.

Output (do not copy)

```
NAME      REFERENCE         TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
web       Deployment/web    0%/1%     1         4         1          1m
```
content_copy

2. To inspect the configuration of `HorizontalPodAutoscaler` in YAML form, execute the following command:

```
kubectl describe horizontalpodautoscaler web
```
content_copy

The output should look like the example.

Output (do not copy)

```
Name:                                                  web
Namespace:                                             default
Labels:                                                <none>
Annotations:                                           <none>
CreationTimestamp:                                     Tue, 08 Sep 2020...
Reference:                                             Deployment/web
Metrics:                                               ( current / target )
  resource cpu on pods  (as a percentage of request):  0% (0) / 1%
Min replicas:                                          1
Max replicas:                                          4
Deployment pods:                                       1 current / 1 desired
Conditions:
  Type            Status  Reason               Message
  ----            ------  ------               -------
  AbleToScale     True    ScaleDownStabilized  recent recommendations [...]
  ScalingActive   True    ValidMetricFound     the HPA was able to [...]
  ScalingLimited  False   DesiredWithinRange   the desired count [...]
Events:           <none>
```
content_copy

3. To view the configuration of HorizontalPodAutoscaler in YAML form, execute the following command:

```
kubectl get horizontalpodautoscaler web -o yaml
```
content_copy

The output should look like the example.

Output (do not copy)

```
apiVersion: autoscaling/v1
kind: HorizontalPodAutoscaler
metadata:
  annotations:
    autoscaling.alpha.kubernetes.io/conditions:  [...]
autoscaling.alpha.kubernetes.io/current-metrics: [...]
  creationTimestamp: 2018-11-14T02:59:28Z
  name: web
  namespace: default
  resourceVersion: "14588"
  selfLink: /apis/autoscaling/v1/namespaces/[...]
spec:
  maxReplicas: 4
  minReplicas: 1
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: web
  targetCPUUtilizationPercentage: 1
status:
  currentCPUUtilizationPercentage: 0
  currentReplicas: 1
  desiredReplicas: 1
content_copy
```

Test the autoscale configuration

You need to create a heavy load on the web application to force it to scale out. You create a configuration file that defines a deployment of four containers that run an infinite loop of HTTP queries against the sample application web server.

You create the load on your web application by deploying the loadgen application using the `loadgen.yaml` file that has been provided for you.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: loadgen
spec:
  replicas: 4
  selector:
    matchLabels:
      app: loadgen
  template:
    metadata:
      labels:
        app: loadgen
    spec:
      containers:
```

```
      - name: loadgen
        image: k8s.gcr.io/busybox
        args:
        - /bin/sh
        - -c
        - while true; do wget -q -O- http://web:8080; done
content_copy
```

1. To deploy this container, execute the following command:

```
kubectl apply -f loadgen.yaml
content_copy
```

After you deploy this manifest, the web Pod should begin to scale.

Click *Check my progress* to verify the objective.

Deploying the loadgen application

Check my progress

2. Get the list of deployments to verify that the load generator is running.

```
kubectl get deployment
content_copy
```

The output should look like the example.

Output (do not copy)

```
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
loadgen   4/4     4            4           26s
web       1/1     1            1           14m
content_copy
```

3. Inspect HorizontalPodAutoscaler.

```
kubectl get hpa
content_copy
```

Once the loadgen Pod starts to generate traffic, the web deployment CPU utilization begins to increase. In the example output, the targets are now at 35% CPU utilization compared to the 1% CPU threshold.

Output (do not copy)

```
NAME      REFERENCE         TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
web       Deployment/web    35%/1%    1         4         1          8m
content_copy
```

4. After a few minutes, inspect the HorizontalPodAutoscaler again.

```
kubectl get hpa
content_copy
```

The autoscaler has increased the web deployment to four replicas.

Output (do not copy)

```
NAME      REFERENCE         TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
web       Deployment/web    88%/1%    1         4         4          9m
content_copy
```

5. To stop the load on the web application, scale the loadgen deployment to zero replicas.

```
kubectl scale deployment loadgen --replicas 0
content_copy
```

6. Get the list of deployments to verify that loadgen has scaled down.

```
kubectl get deployment
content_copy
```

The loadgen deployment should have zero replicas.

Output (do not copy)

```
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
loadgen   0/0     0            0           3m40s
web       2/4     4            2           18m
web       2/4     4            2           18m
content_copy
```

Note: You need to wait 2 to 3 minutes before you list the deployments again.

7. Get the list of deployments to verify that the web application has scaled down to the minimum value of 1 replica that you configured when you deployed the autoscaler.

```
kubectl get deployment
content_copy
```

You should now have one deployment of the web application.

Output (do not copy)

```
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
loadgen   0/0     0            0           6m28s
web       4/4     4            4           20m
content_copy
```

Task 3. Manage node pools

In this task, you create a new pool of nodes using preemptible instances, and then you constrain the web deployment to run only on the preemptible nodes.

Add a node pool

1. To deploy a new node pool with three preemptible VM instances, execute the following command:

```
gcloud container node-pools create "temp-pool-1" \
--cluster=$my_cluster --zone=$my_zone \
--num-nodes "2" --node-labels=temp=true --preemptible
content_copy
```

If you receive an error that no preemptible instances are available you can remove the `--preemptible` option to proceed with the lab.

2. Get the list of nodes to verify that the new nodes are ready.

```
kubectl get nodes
content_copy
```

You should now have `4` nodes.

Your names will be different from the example output.

Output (do not copy)

```
NAME                                          STATUS   ROLES    AGE   VERSION
gke-standard-cluster-1-default-pool...xc   Ready    <none>   33m   v1.15.12-
gke.2
gke-standard-cluster-1-default-pool...q8   Ready    <none>   33m   v1.15.12-
gke.2
gke-standard-cluster-1-temp-pool-1-...vj   Ready    <none>   32s   v1.15.12-
gke.2
gke-standard-cluster-1-temp-pool-1-...xj   Ready    <none>   37s   v1.15.12-
gke.2
content_copy
```

All the nodes that you added have the `temp=true` label because you set that label when you created the node-pool. This label makes it easier to locate and configure these nodes.

3. To list only the nodes with the `temp=true` label, execute the following command:

```
kubectl get nodes -l temp=true
content_copy
```

You should see only the two nodes that you added.

Your names will be different from the example output.

Output (do not copy)

```
NAME                                          STATUS   ROLES    AGE     VERSION
gke-standard-cluster-1-temp-pool-1-...vj   Ready    <none>   3m26s
v1.15.12-gke.2
gke-standard-cluster-1-temp-pool-1-...xj   Ready    <none>   3m31s
v1.15.12-gke.2
content_copy
```

Control scheduling with taints and tolerations

To prevent the scheduler from running a Pod on the temporary nodes, you add a taint to each of the nodes in the temp pool. Taints are implemented as a key-value pair with an effect (such as NoExecute) that determines whether Pods can run on a certain node. Only nodes that are configured to tolerate the key-value of the taint are scheduled to run on these nodes.

1. To add a taint to each of the newly created nodes, execute the following command.

You can use the `temp=true` label to apply this change across all the new nodes simultaneously.

```
kubectl taint node -l temp=true nodetype=preemptible:NoExecute
content_copy
```

To allow application Pods to execute on these tainted nodes, you must add a tolerations key to the deployment configuration.

2. Edit the `web.yaml` file to add the following key in the template's `spec` section:

```
tolerations:
- key: "nodetype"
```

```
  operator: Equal
  value: "preemptible"
```
content_copy

The spec section of the file should look like the example.

```
...
    spec:
      tolerations:
      - key: "nodetype"
        operator: Equal
        value: "preemptible"
      containers:
      - image: gcr.io/google-samples/hello-app:1.0
        name: web
        ports:
        - containerPort: 8080
          protocol: TCP
```
content_copy

3. To force the web deployment to use the new node-pool add a nodeSelector key in the template's `spec` section. This is parallel to the tolerations key you just added.

```
     nodeSelector:
       temp: "true"
```
content_copy

Note: GKE adds a custom label to each node called `cloud.google.com/gke-nodepool` that contains the name of the node-pool that the node belongs to. This key can also be used as part of a nodeSelector to ensure Pods are only deployed to suitable nodes.
The full web.yaml deployment should now look as follows.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: web
spec:
  replicas: 1
  selector:
    matchLabels:
      run: web
  template:
    metadata:
      labels:
        run: web
    spec:
      tolerations:
      - key: "nodetype"
        operator: Equal
        value: "preemptible"
      nodeSelector:
        temp: "true"
      containers:
```

```
      - image: gcr.io/google-samples/hello-app:1.0
        name: web
        ports:
        - containerPort: 8080
          protocol: TCP
content_copy
```

4. To apply this change, execute the following command:

```
kubectl apply -f web.yaml
content_copy
```

If you have problems editing this file successfully you can use the pre-prepared sample file called `web-tolerations.yaml` instead.

Click *Check my progress* to verify the objective.

Manage node pools

Check my progress

5. Get the list of Pods.

```
kubectl get pods
content_copy
```

Your names might be different from the example output.

Output (do not copy)

```
NAME                   READY     STATUS    RESTARTS   AGE
web-7cb566bccd-pkfst   1/1       Running   0          1m
content_copy
```

6. To confirm the change, inspect the running web Pod(s) using the following command

```
kubectl describe pods -l run=web
content_copy
```

A `Tolerations` section with `nodetype=preemptible` in the list should appear near the bottom of the (truncated) output.

Output (do not copy)

```
<SNIP>

Node-Selectors:  temp=true
Tolerations:     node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                 node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
                 nodetype=preemptible
Events:
```

```
<SNIP>
content_copy
```

The output confirms that the Pods will tolerate the taint value on the new preemptible nodes, and thus that they can be scheduled to execute on those nodes.

7. To force the web application to scale out again scale the loadgen deployment back to four replicas.

```
kubectl scale deployment loadgen --replicas 4
content_copy
```

You could scale just the web application directly but using the loadgen app will allow you to see how the different taint, toleration and nodeSelector settings that apply to the web and loadgen applications affect which nodes they are scheduled on.

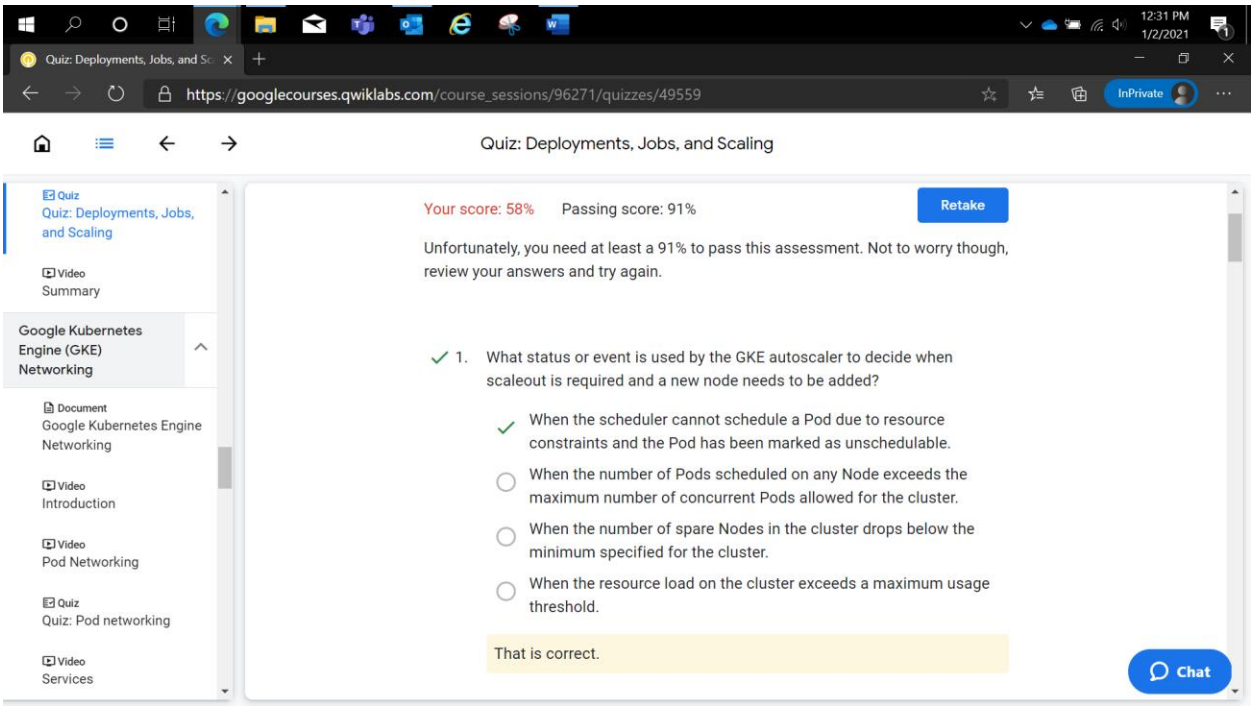
8. Get the list of Pods using the wide output format to show the nodes running the Pods

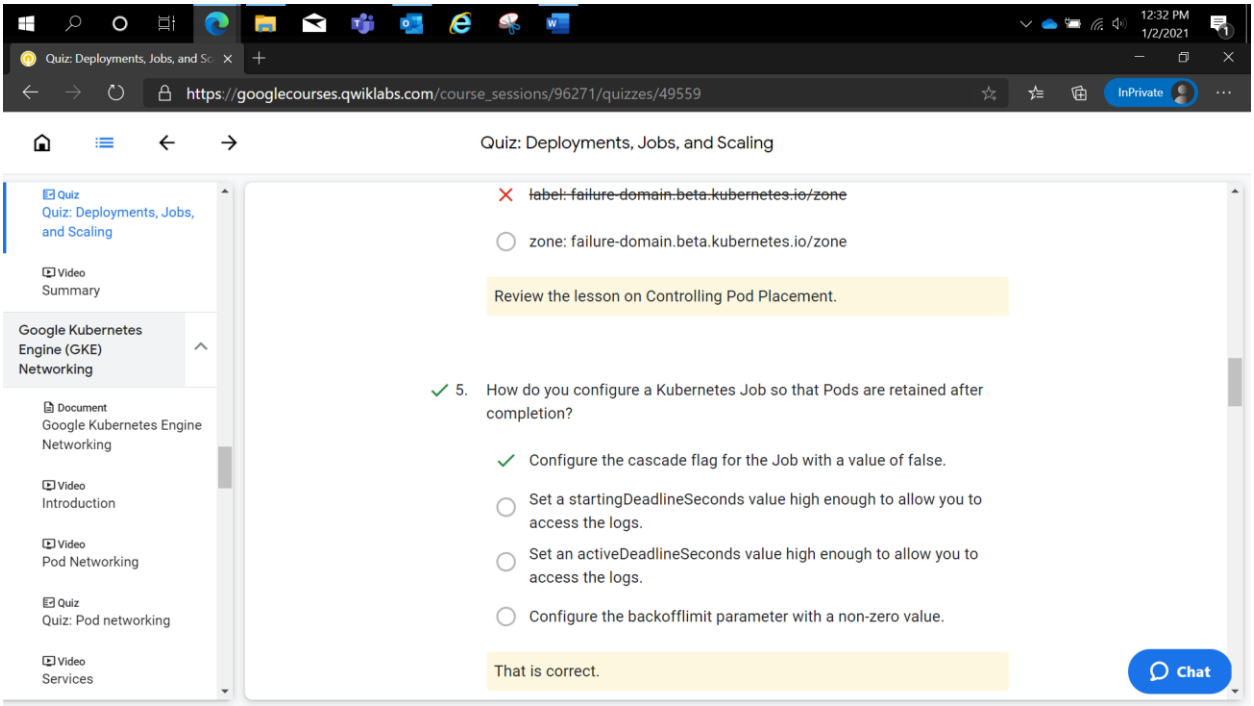
```
kubectl get pods -o wide
content_copy
```

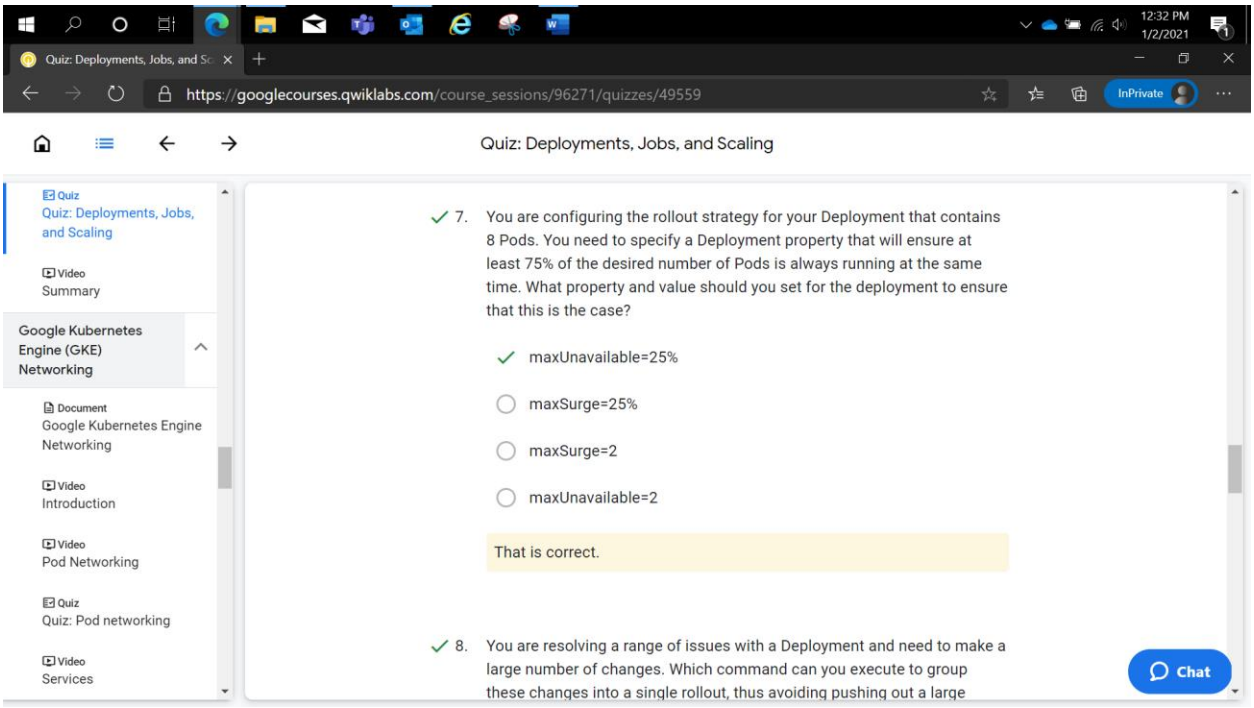
This shows that the loadgen app is running only on `default-pool` nodes while the web app is running only the preemptible nodes in `temp-pool-1`.

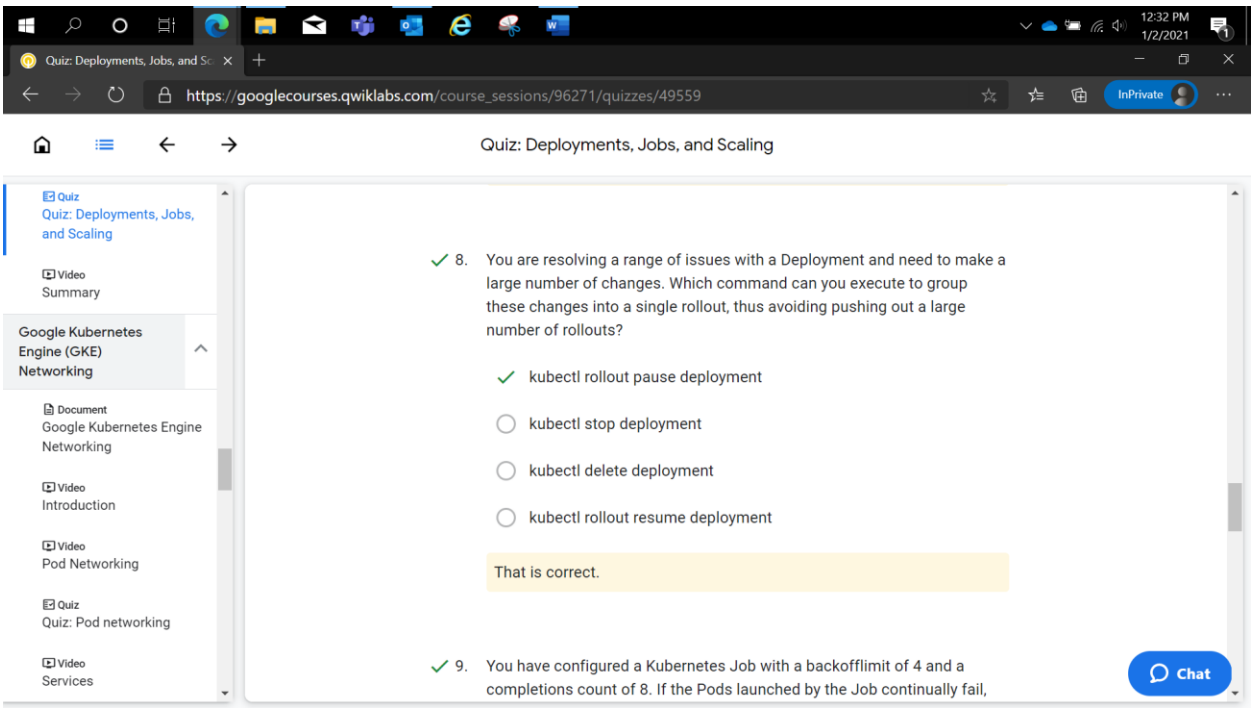
The taint setting prevents Pods from running on the preemptible nodes so the loadgen application only runs on the default pool. The toleration setting allows the web application to run on the preemptible nodes and the nodeSelector forces the web application Pods to run on those nodes.

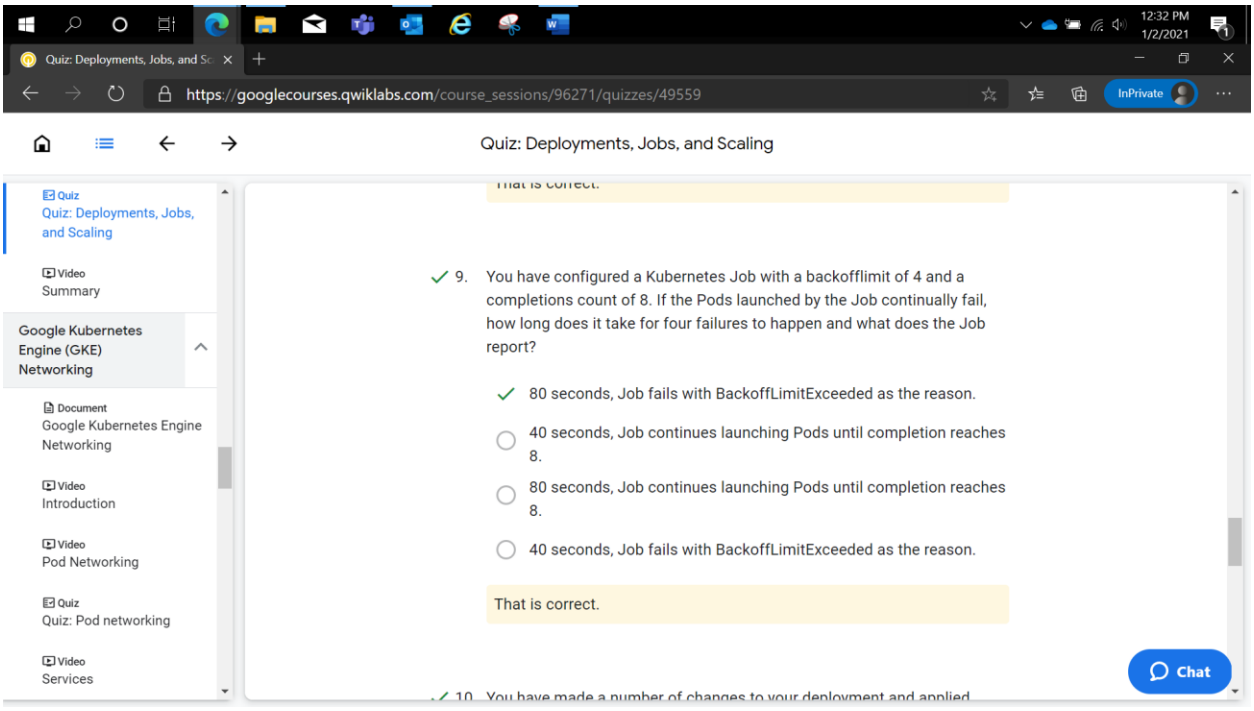
```
NAME         READY  STATUS    [...]          NODE
Loadgen-x0   1/1    Running   [...]          gke-xx-default-pool-y0
loadgen-x1   1/1    Running   [...]          gke-xx-default-pool-y2
loadgen-x3   1/1    Running   [...]          gke-xx-default-pool-y3
loadgen-x4   1/1    Running   [...]          gke-xx-default-pool-y4
web-x1       1/1    Running   [...]          gke-xx-temp-pool-1-z1
web-x2       1/1    Running   [...]          gke-xx-temp-pool-1-z2
web-x3       1/1    Running   [...]          gke-xx-temp-pool-1-z3
web-x4       1/1    Running   [...]          gke-xx-temp-pool-1-z4
```

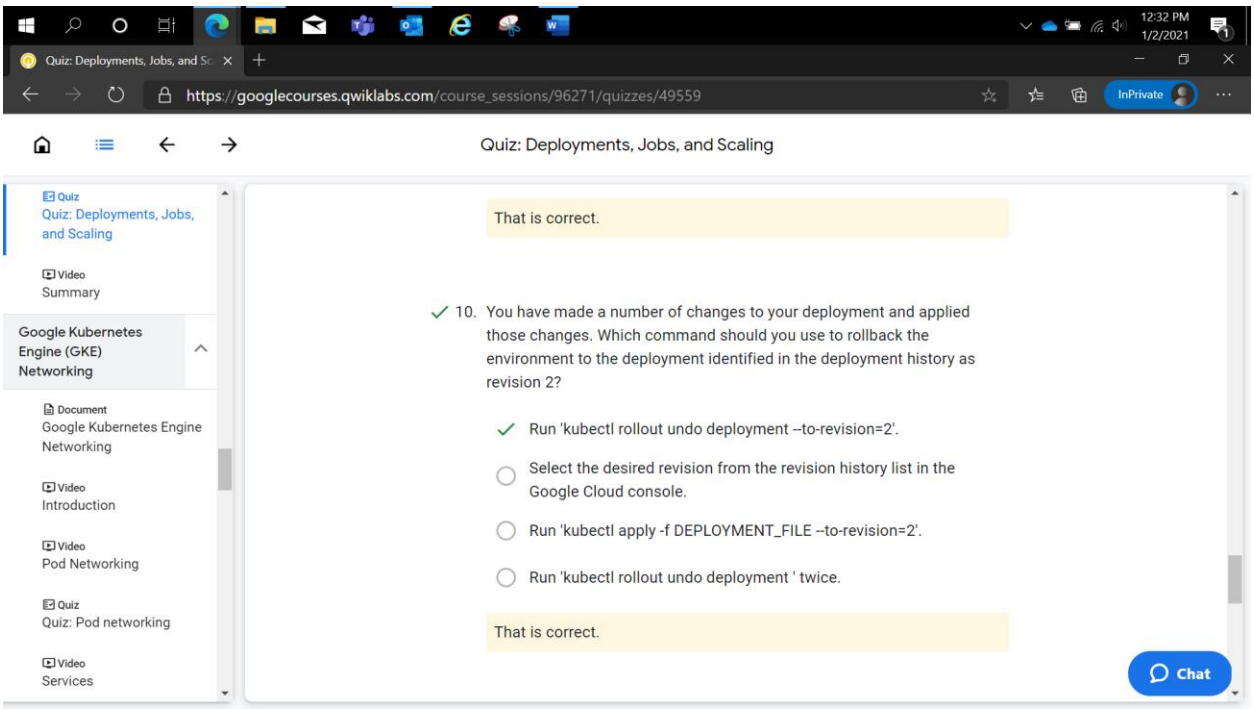

12:31 PM
1/2/2021
Quiz: Deployments, Jobs, and Sc
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49559
InPrivate
Quiz: Deployments, Jobs, and Scaling
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Video
Introduction
Video
Pod Networking
Quiz
Quiz: Pod networking
Video
Services
Your score: 58%
Passing score: 91%
Retake
Unfortunately, you need at least a 91% to pass this assessment. Not to worry though, review your answers and try again.
1. What status or event is used by the GKE autoscaler to decide when scaleout is required and a new node needs to be added?
When the scheduler cannot schedule a Pod due to resource constraints and the Pod has been marked as unschedulable.
When the number of Pods scheduled on any Node exceeds the maximum number of concurrent Pods allowed for the cluster.
When the number of spare Nodes in the cluster drops below the minimum specified for the cluster.
When the resource load on the cluster exceeds a maximum usage threshold.
That is correct.
Chat

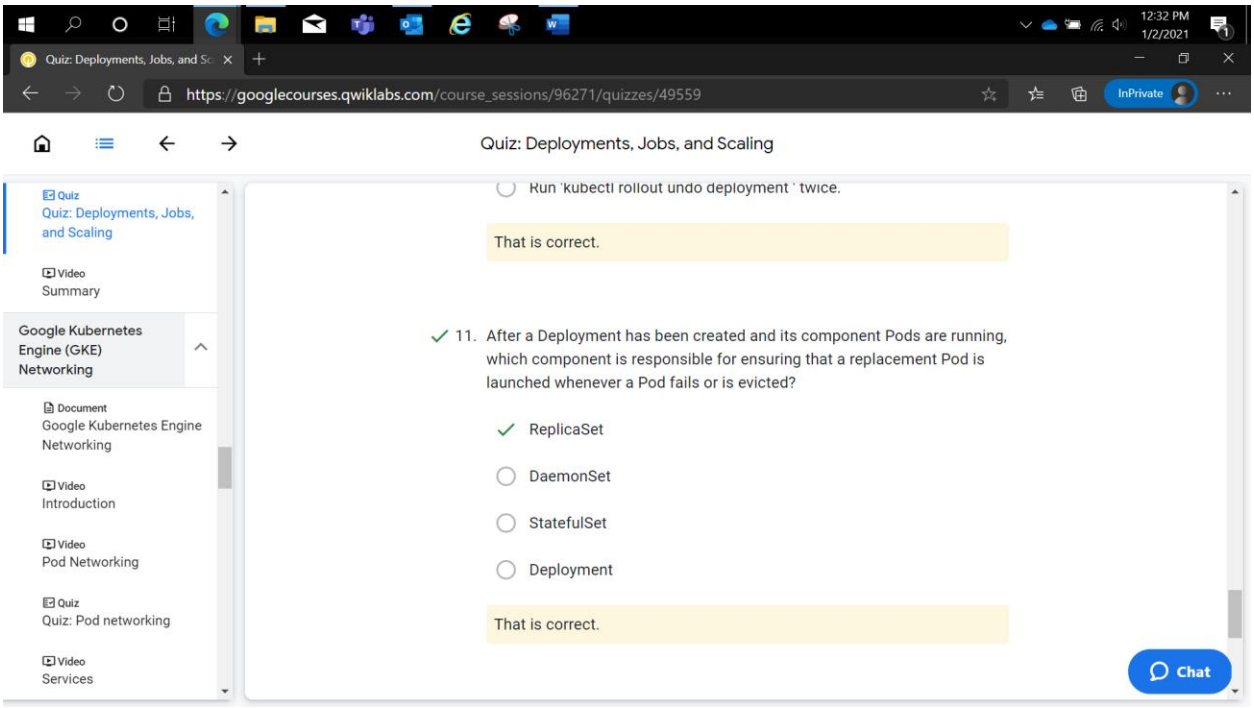

12:32 PM
1/2/2021
Quiz: Deployments, Jobs, and Sc
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49559
InPrivate
Quiz: Deployments, Jobs, and Scaling
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Video
Introduction
Video
Pod Networking
Quiz
Quiz: Pod networking
Video
Services
label: failure-domain.beta.kubernetes.io/zone
zone: failure-domain.beta.kubernetes.io/zone
Review the lesson on Controlling Pod Placement.
5. How do you configure a Kubernetes Job so that Pods are retained after completion?
Configure the cascade flag for the Job with a value of false.
Set a startingDeadlineSeconds value high enough to allow you to access the logs.
Set an activeDeadlineSeconds value high enough to allow you to access the logs.
Configure the backofflimit parameter with a non-zero value.
That is correct.
Chat

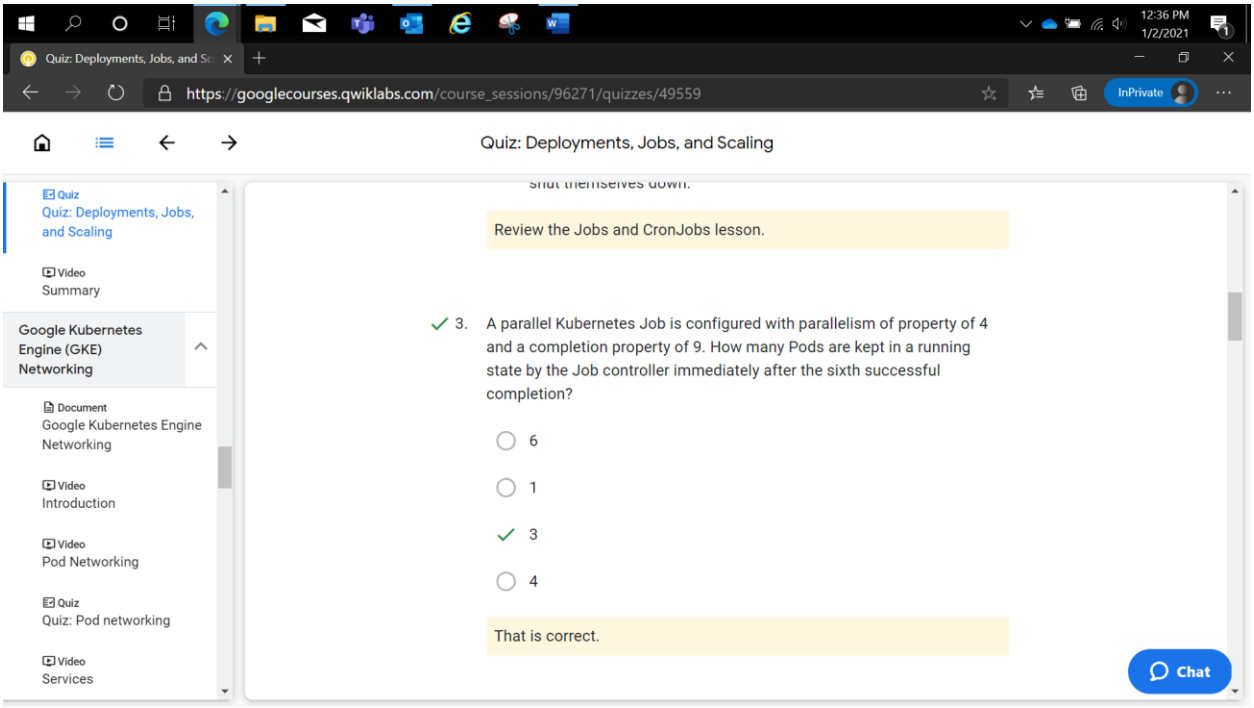

Quiz: Deployments, Jobs, and Sc
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49559
InPrivate
12:32 PM
1/2/2021
Quiz: Deployments, Jobs, and Scaling
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Video
Introduction
Video
Pod Networking
Quiz
Quiz: Pod networking
Video
Services
7. You are configuring the rollout strategy for your Deployment that contains 8 Pods. You need to specify a Deployment property that will ensure at least 75% of the desired number of Pods is always running at the same time. What property and value should you set for the deployment to ensure that this is the case?
maxUnavailable=25%
maxSurge=25%
maxSurge=2
maxUnavailable=2
That is correct.
8. You are resolving a range of issues with a Deployment and need to make a large number of changes. Which command can you execute to group these changes into a single rollout, thus avoiding pushing out a large
Chat

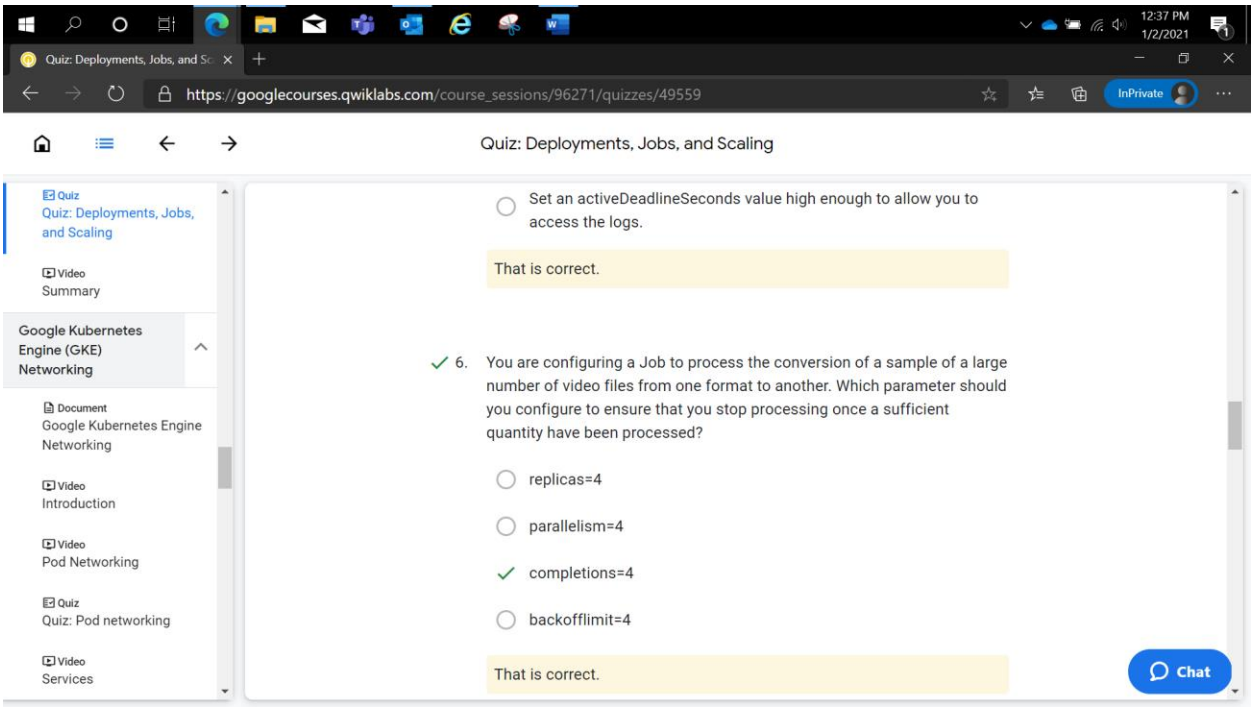

Quiz: Deployments, Jobs, and Sc
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49559
InPrivate
12:32 PM
1/2/2021
Quiz: Deployments, Jobs, and Scaling
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Video
Introduction
Video
Pod Networking
Quiz
Quiz: Pod networking
Video
Services
8. You are resolving a range of issues with a Deployment and need to make a large number of changes. Which command can you execute to group these changes into a single rollout, thus avoiding pushing out a large number of rollouts?
kubectl rollout pause deployment
kubectl stop deployment
kubectl delete deployment
kubectl rollout resume deployment
That is correct.
9. You have configured a Kubernetes Job with a backofflimit of 4 and a completions count of 8. If the Pods launched by the Job continually fail,
Chat

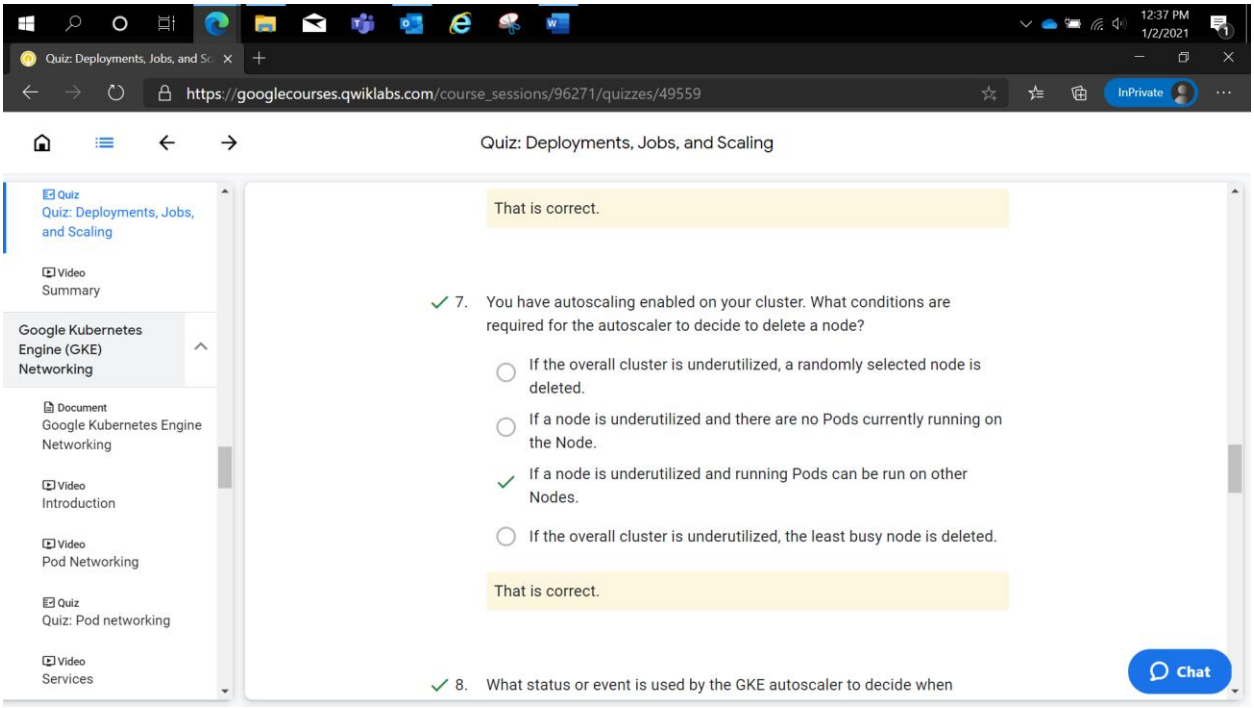

Quiz: Deployments, Jobs, and Sc
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49559
InPrivate
12:32 PM
1/2/2021
Quiz: Deployments, Jobs, and Scaling
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Video
Introduction
Video
Pod Networking
Quiz
Quiz: Pod networking
Video
Services
That is correct.
9. You have configured a Kubernetes Job with a backofflimit of 4 and a completions count of 8. If the Pods launched by the Job continually fail, how long does it take for four failures to happen and what does the Job report?
80 seconds, Job fails with BackoffLimitExceeded as the reason.
40 seconds, Job continues launching Pods until completion reaches 8.
80 seconds, Job continues launching Pods until completion reaches 8.
40 seconds, Job fails with BackoffLimitExceeded as the reason.
That is correct.
Chat

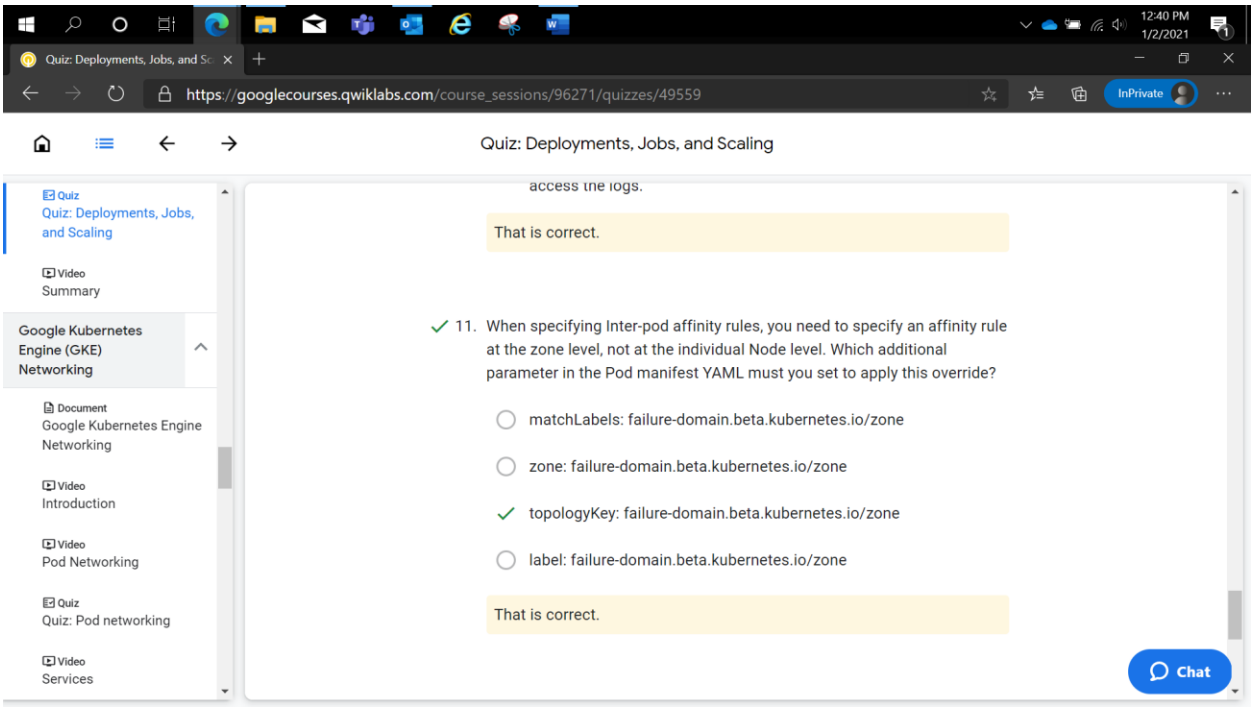

Quiz: Deployments, Jobs, and Sc
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49559
InPrivate
12:32 PM
1/2/2021
Quiz: Deployments, Jobs, and Scaling
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Video
Introduction
Video
Pod Networking
Quiz
Quiz: Pod networking
Video
Services
That is correct.
10. You have made a number of changes to your deployment and applied those changes. Which command should you use to rollback the environment to the deployment identified in the deployment history as revision 2?
Run 'kubectl rollout undo deployment --to-revision=2'.
Select the desired revision from the revision history list in the Google Cloud console.
Run 'kubectl apply -f DEPLOYMENT_FILE --to-revision=2'.
Run 'kubectl rollout undo deployment ' twice.
That is correct.
Chat

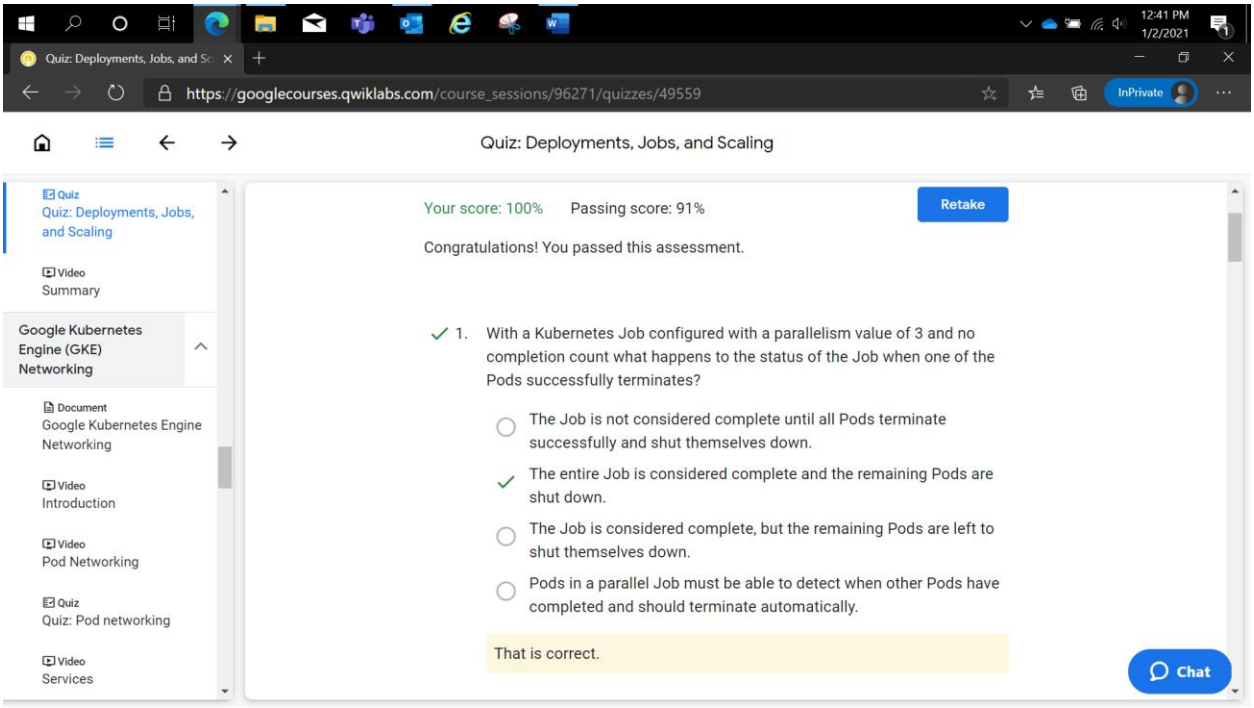

12:32 PM
1/2/2021
Quiz: Deployments, Jobs, and Sc
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49559
InPrivate
Quiz: Deployments, Jobs, and Scaling
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Video
Introduction
Video
Pod Networking
Quiz
Quiz: Pod networking
Video
Services
Run 'kubectl rollout undo deployment ' twice.
That is correct.
11. After a Deployment has been created and its component Pods are running, which component is responsible for ensuring that a replacement Pod is launched whenever a Pod fails or is evicted?
ReplicaSet
DaemonSet
StatefulSet
Deployment
That is correct.
Chat


12:36 PM
1/2/2021
Quiz: Deployments, Jobs, and Sc
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49559
InPrivate
Quiz: Deployments, Jobs, and Scaling
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Video
Introduction
Video
Pod Networking
Quiz
Quiz: Pod networking
Video
Services
shut themselves down.
Review the Jobs and CronJobs lesson.
3. A parallel Kubernetes Job is configured with parallelism of property of 4 and a completion property of 9. How many Pods are kept in a running state by the Job controller immediately after the sixth successful completion?
6
1
3
4
That is correct.
Chat


Quiz: Deployments, Jobs, and Sc
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49559
InPrivate
12:37 PM
1/2/2021
Quiz: Deployments, Jobs, and Scaling
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Video
Introduction
Video
Pod Networking
Quiz
Quiz: Pod networking
Video
Services
Set an activeDeadlineSeconds value high enough to allow you to access the logs.
That is correct.
6. You are configuring a Job to process the conversion of a sample of a large number of video files from one format to another. Which parameter should you configure to ensure that you stop processing once a sufficient quantity have been processed?
replicas=4
parallelism=4
completions=4
backofflimit=4
That is correct.
Chat


Quiz: Deployments, Jobs, and Sc
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49559
InPrivate
12:37 PM
1/2/2021
Quiz: Deployments, Jobs, and Scaling
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Video
Introduction
Video
Pod Networking
Quiz
Quiz: Pod networking
Video
Services
That is correct.
7. You have autoscaling enabled on your cluster. What conditions are required for the autoscaler to decide to delete a node?
If the overall cluster is underutilized, a randomly selected node is deleted.
If a node is underutilized and there are no Pods currently running on the Node.
If a node is underutilized and running Pods can be run on other Nodes.
If the overall cluster is underutilized, the least busy node is deleted.
That is correct.
8. What status or event is used by the GKE autoscaler to decide when
Chat


Quiz: Deployments, Jobs, and Sc
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49559
InPrivate
12:40 PM
1/2/2021
Quiz: Deployments, Jobs, and Scaling
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Video
Introduction
Video
Pod Networking
Quiz
Quiz: Pod networking
Video
Services
access the logs.
That is correct.
11. When specifying Inter-pod affinity rules, you need to specify an affinity rule at the zone level, not at the individual Node level. Which additional parameter in the Pod manifest YAML must you set to apply this override?
matchLabels: failure-domain.beta.kubernetes.io/zone
zone: failure-domain.beta.kubernetes.io/zone
topologyKey: failure-domain.beta.kubernetes.io/zone
label: failure-domain.beta.kubernetes.io/zone
That is correct.
Chat


Quiz: Deployments, Jobs, and Sc
https://googlecourses.qwiklabs.com/course_sessions/96271/quizzes/49559
InPrivate
12:41 PM
1/2/2021
Quiz: Deployments, Jobs, and Scaling
Quiz
Quiz: Deployments, Jobs, and Scaling
Video
Summary
Google Kubernetes Engine (GKE) Networking
Document
Google Kubernetes Engine Networking
Video
Introduction
Video
Pod Networking
Quiz
Quiz: Pod networking
Video
Services
Your score: 100%
Passing score: 91%
Retake
Congratulations! You passed this assessment.
1. With a Kubernetes Job configured with a parallelism value of 3 and no completion count what happens to the status of the Job when one of the Pods successfully terminates?
The Job is not considered complete until all Pods terminate successfully and shut themselves down.
The entire Job is considered complete and the remaining Pods are shut down.
The Job is considered complete, but the remaining Pods are left to shut themselves down.
Pods in a parallel Job must be able to detect when other Pods have completed and should terminate automatically.
That is correct.
Chat