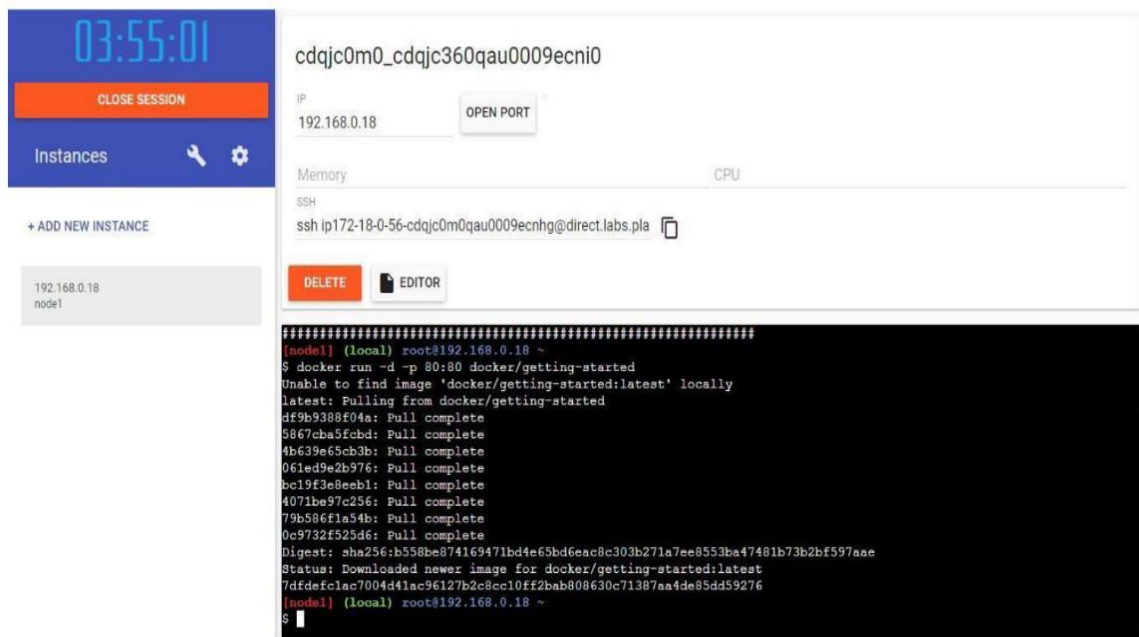


Assignment -4
Kubernetes/Docker

Assignment Date	09 September 2022
Student Name	Ms. Barghana Nisha S
Student Roll Number	111519205003
Maximum Marks	2 Marks

Question 1:

Pull an Image from docker hub and run it in docker playground.



The screenshot displays a Docker Playground interface. On the left, a sidebar shows a timer at 03:55:01, a 'CLOSE SESSION' button, and a list of instances with one instance named '192.168.0.18 node1'. The main panel shows the instance details for 'cdqjc0m0_cdqjc360qau0009ecni0' with IP '192.168.0.18'. Below this, there are buttons for 'DELETE' and 'EDITOR'. The terminal window shows the following commands and output:

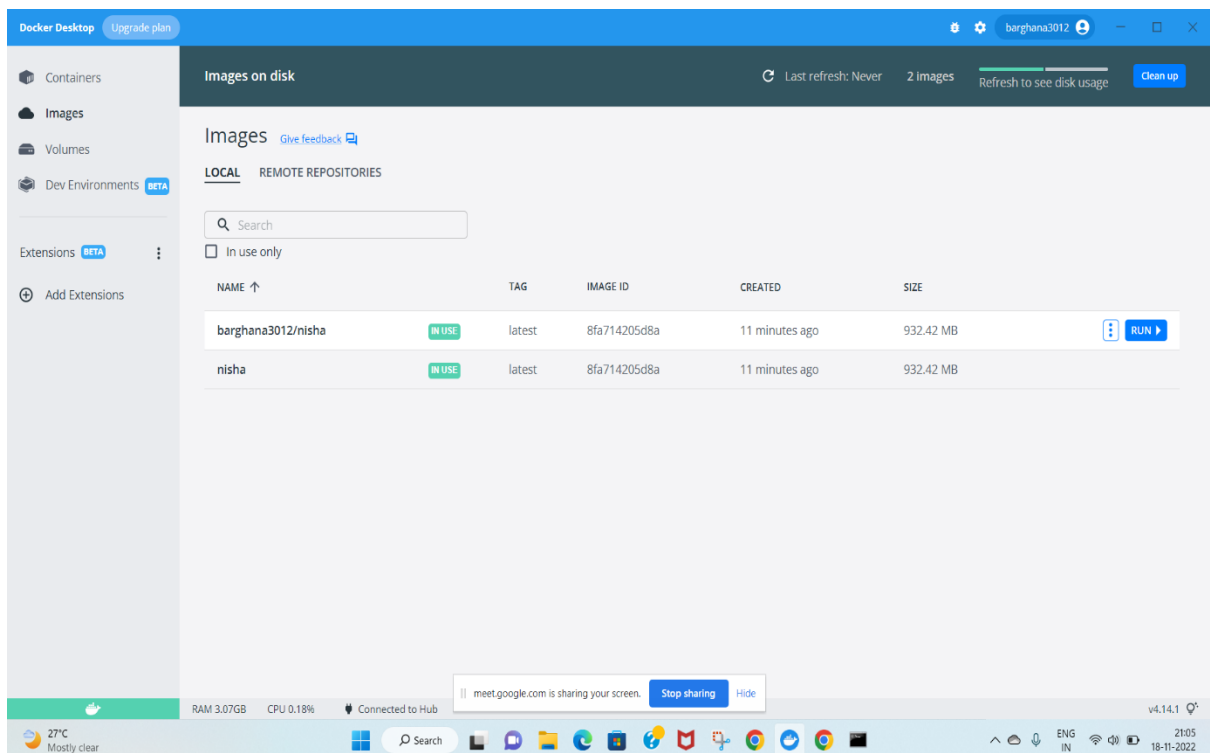
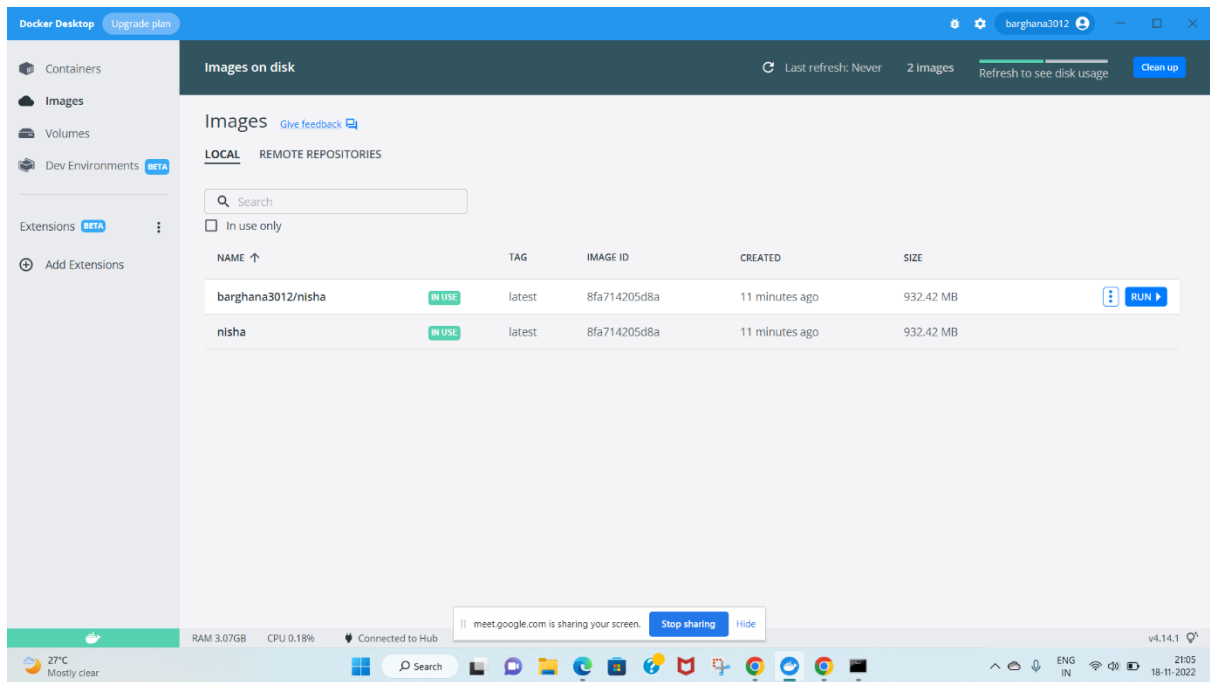
```
#####  
[node1] (local) root@192.168.0.18 ~  
$ docker run -d -p 80:80 docker/getting-started  
Unable to find image 'docker/getting-started:latest' locally  
latest: Pulling from docker/getting-started  
df9b9388f04a: Pull complete  
5867cba5fcbd: Pull complete  
4b639e65cb3b: Pull complete  
061ed9e2b976: Pull complete  
bc19f3e8eeb1: Pull complete  
4071be97c256: Pull complete  
79b586f1a54b: Pull complete  
0c9732f525d6: Pull complete  
Digest: sha256:b558be874169471bd4e65bd6eac8c303b271a7ee8553ba47481b73b2bf597aae  
Status: Downloaded newer image for docker/getting-started:latest  
7dfdefclac7004d41ac96127b2c8cc10ff2bab808630c71387aa4de85dd59276  
[node1] (local) root@192.168.0.18 ~  
$
```

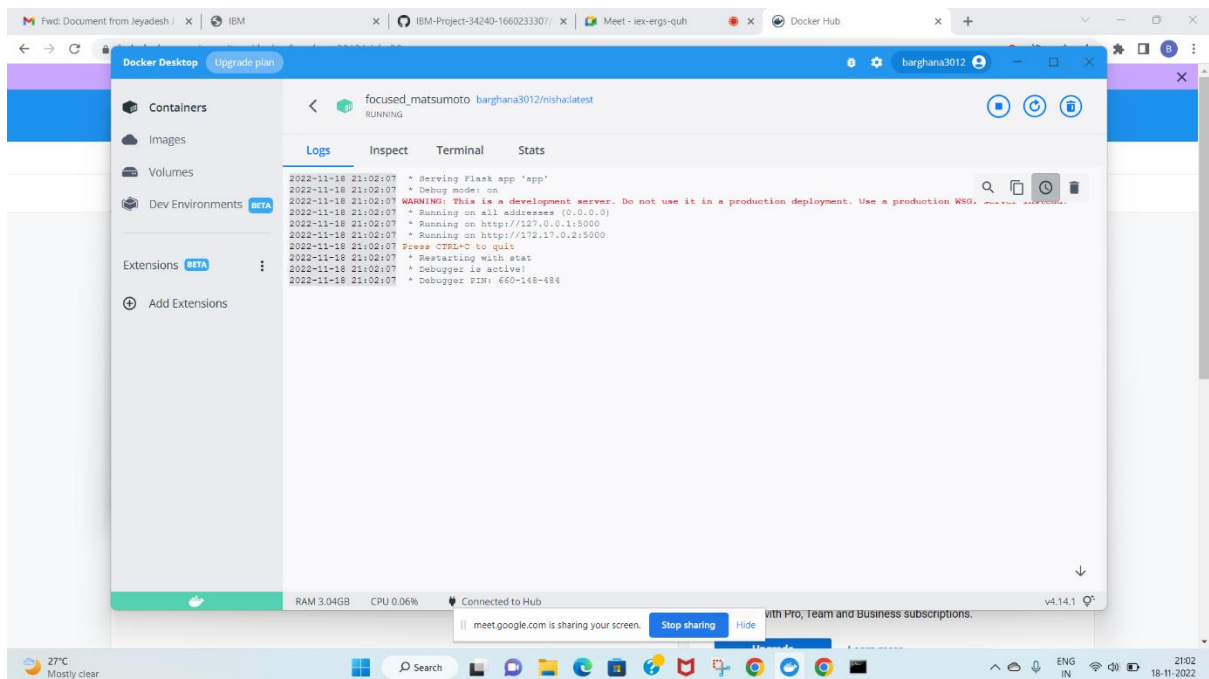
```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.22000.1098]
(c) Microsoft Corporation. All rights reserved.

C:\Users\farha\Desktop\Nisha 30\flask_with_form_and_docker-main>docker build -t nisha .
[+] Building 86.0s (11/11) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 170B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/python:3.10.6
=> [auth] library/python:pull token for registry-1.docker.io
=> [internal] load build context
=> => transferring context: 10.20MB
=> [1/5] FROM docker.io/library/python:3.10.6@sha256:745efdfb7e4aac9a84226d8c62d8bc35a693e8979a240d29677cb01e5a
=> => resolve docker.io/library/python:3.10.6@sha256:745efdfb7e4aac9a84226d8c62d8bc35a693e8979a240d29677cb01e5a
=> => sha256:c275a6388b302830d3f1696d7770ba3cd8189126f68be7d118b957488bcbf84 8.53kB / 8.53kB
=> => sha256:1d7150c184f8c35c9b6d1d13bc15a727081118438c179308420199d6d4e 55.41kB / 55.41kB
=> => sha256:f4bc7528c185210129a8e6bf9362a7802a7616a351a8855dca4108838057a6 18.89kB / 18.89kB
=> => sha256:745efdfb7e4aac9a84226d8c62d8bc35a693e8979a240d29677cb01e5a91052 2.35kB / 2.35kB
=> => sha256:8d1f943caaf3c0d5df5a02be7958030d0485700175b7bc56d8f77ac13ca 2.22kB / 2.22kB
=> => sha256:3e8dd11e55e7ade117f2117b7577b95c7e1f06d11f8704a0926096681f6e4 5.10kB / 5.10kB
=> => sha256:53a0972fbcdfc8a93b1d2b20e758e11ac0ef68b0aef6f1043c86da1981a 54.50kB / 54.50kB
=> => sha256:0d808111751367187341701e4593ddaf0d613c7980c6553a8e2a150e448a 196.79kB / 196.79kB
=> => sha256:08092d6dd5476f7c302256d4dcff493aebf4dd428aa1d8cb7581e24a6c 6.29kB / 6.29kB
=> => extracting sha256:1671505cc80f8c36539661d3fbc16dc73091f1b6438c4179588428f9a9a2e
=> => extracting sha256:3e8dd11e55e7ade117f2117b7577b95c7e1f06d11f8704a0926096681f6e4 8.33s
=> => extracting sha256:f4bc7528c185210129a8e6bf9362a7802a7616a351a8855dca4108838057a6 0.33s
=> => extracting sha256:53a0972fbcdfc8a93b1d2b20e758e11ac0ef68b0aef6f1043c86da1981a 2.77s
=> => sha256:c71afc63709ad64c5fd3c348586df82b358bb204f0057ea22c6a8a1d203a5 20.02kB / 20.02kB
=> => sha256:80da18b3c704553e0bc5fc42f1aee1c07048f6305f0fa75eb8a285411da40 234B / 234B
=> => sha256:4314471f422d318d4c1c359903aee8d210812c75c311c0a93bc40fbedc3 3.00kB / 3.00kB
=> => extracting sha256:0d808111751367187341701e4593ddaf0d613c7980c6553a8e2a150e448a 8.43s
=> => extracting sha256:08092d6dd5476f7c302256d4dcff493aebf4dd428aa1d8cb7581e24a6c 0.33s
=> => extracting sha256:c71afc63709ad64c5fd3c348586df82b358bb204f0057ea22c6a8a1d203a5 8.86s
=> => extracting sha256:80da18b3c704553e0bc5fc42f1aee1c07048f6305f0fa75eb8a285411da40 0.06s
=> => extracting sha256:4314471f422d318d4c1c359903aee8d210812c75c311c0a93bc40fbedc3 0.36s
=> [2/5] WORKDIR /app
=> [3/5] COPY requirements.txt ./
=> [4/5] RUN pip install -r requirements.txt
=> [5/5] COPY
=> => exporting image
=> => exporting layers
=> => writing image sha256:8fa71a205dbadecfbd0dd5ed2f14ad254809c0ef879f5867e5a17c739f8e
=> => naming to docker.io/library/nisha
Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
C:\Users\farha\Desktop\Nisha 30\flask_with_form_and_docker-> Start
```

Question 2:

Create a docker file for the job portal application and deploy it in Docker desktop application.





[FLASKAPP](#) [CREATE](#) [ABOUT](#)

Blog Page

Messages

Message One

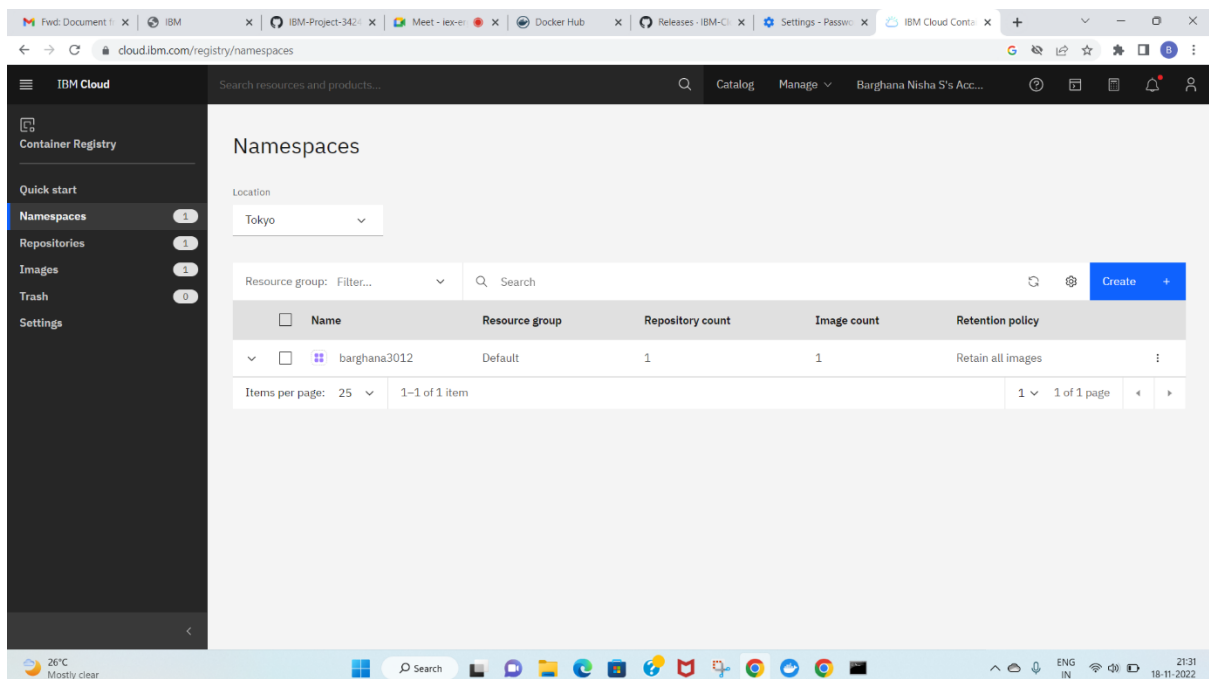
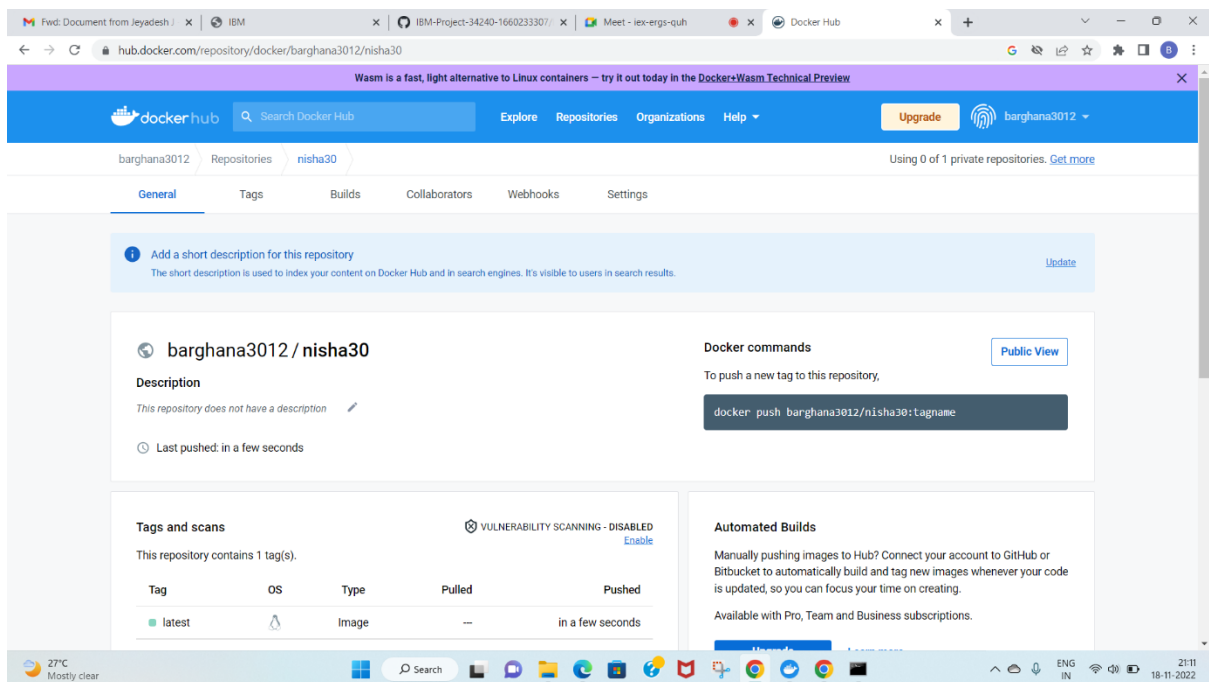
Message One Content

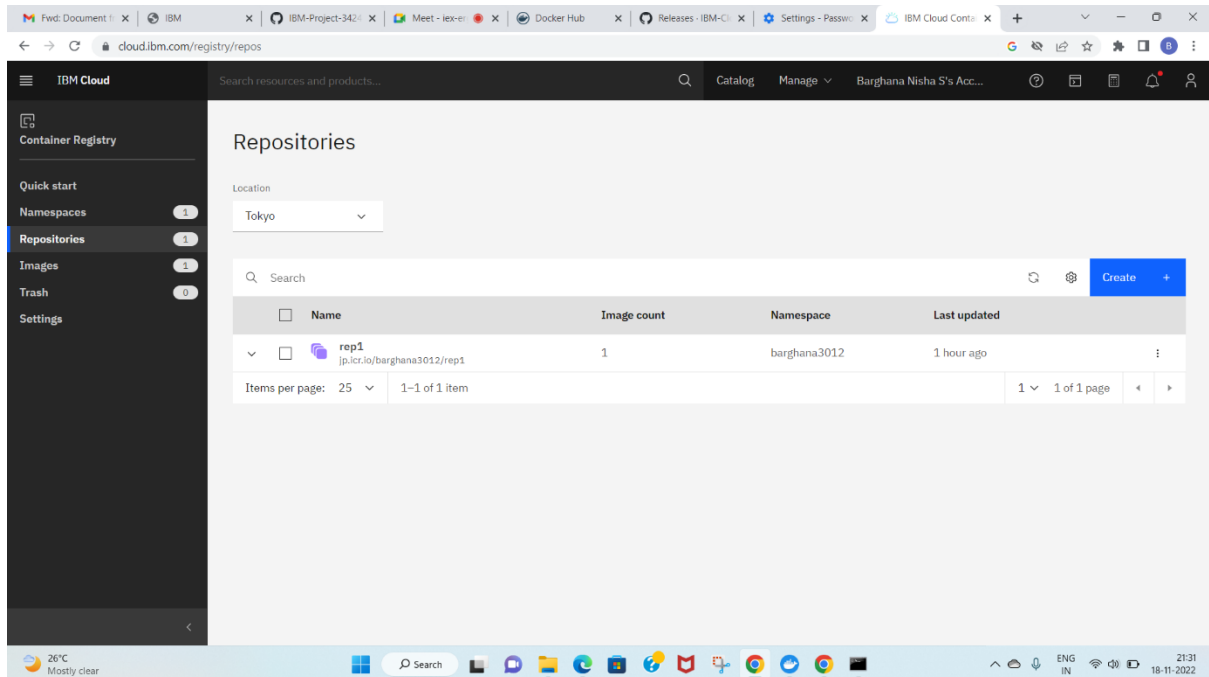
Message Two

Message Two Content

Question 3:

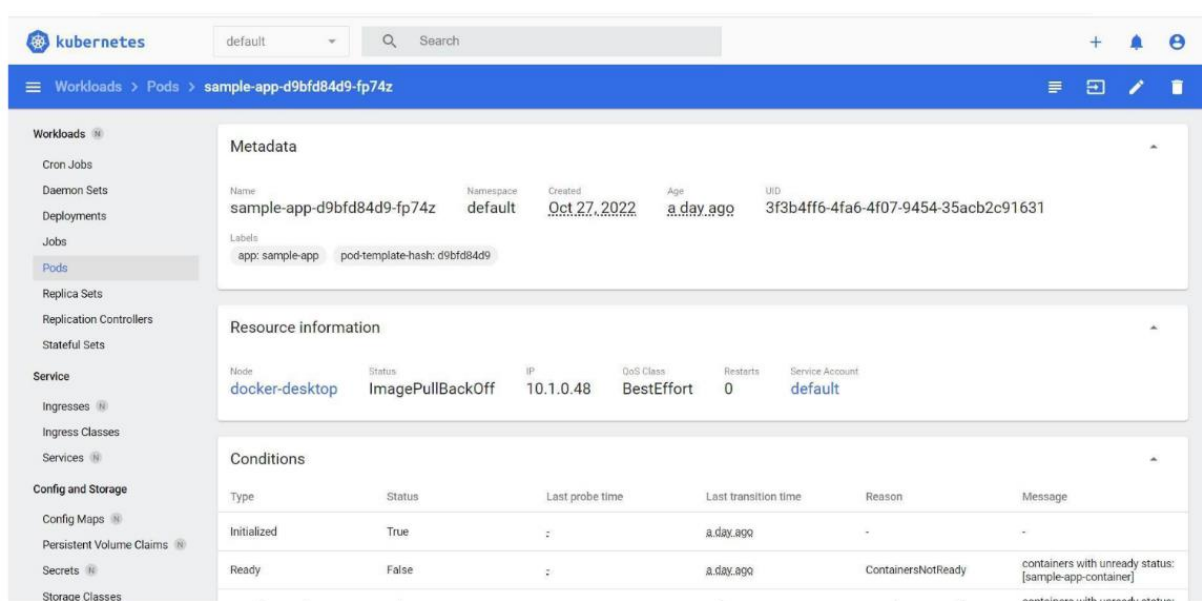
Create a IBM container registry and deploy a hello world app or job portal app.





Question 4:

Create a Kubernetes cluster in IBM cloud and deploy hello world image or job portal image and also expose the same app to run in nodeport.



kubernetes

default

Search

Workloads > Deployments > sample-app

Workloads

Cron Jobs

Daemon Sets

Deployments

Jobs

Pods

Replica Sets

Replication Controllers

Stateful Sets

Service

Ingresses

Ingress Classes

Services

Config and Storage

Config Maps

Persistent Volume Claims

Secrets

Storage Classes

Metadata

Name	Namespace	Created	Age	UID
sample-app	default	Oct 27, 2022	a day ago	9699564b-f097-4168-be80-31f40116a0fc

Annotations

deployment.kubernetes.io/revision: 1

kubectl.kubernetes.io/fast-applied-configuration

Resource information

Strategy	Min ready seconds	Revision history limit
RollingUpdate	0	10

Selector

app: sample-app

Rolling update strategy

Max surge	Max unavailable
25%	25%