# ASSIGNMENT 4

**NAME:** D.SINDHU

**REGISTER NUMBER:** 311519104056

**PROJECT TOPIC:** CUSTOMER CARE REGISTRY

**COLLEGE NAME:** MEENAKSHI SUNDARARAJAN ENGINEERING COLLEGE

**TEAM NUMBER:** PNT2022TMID27825

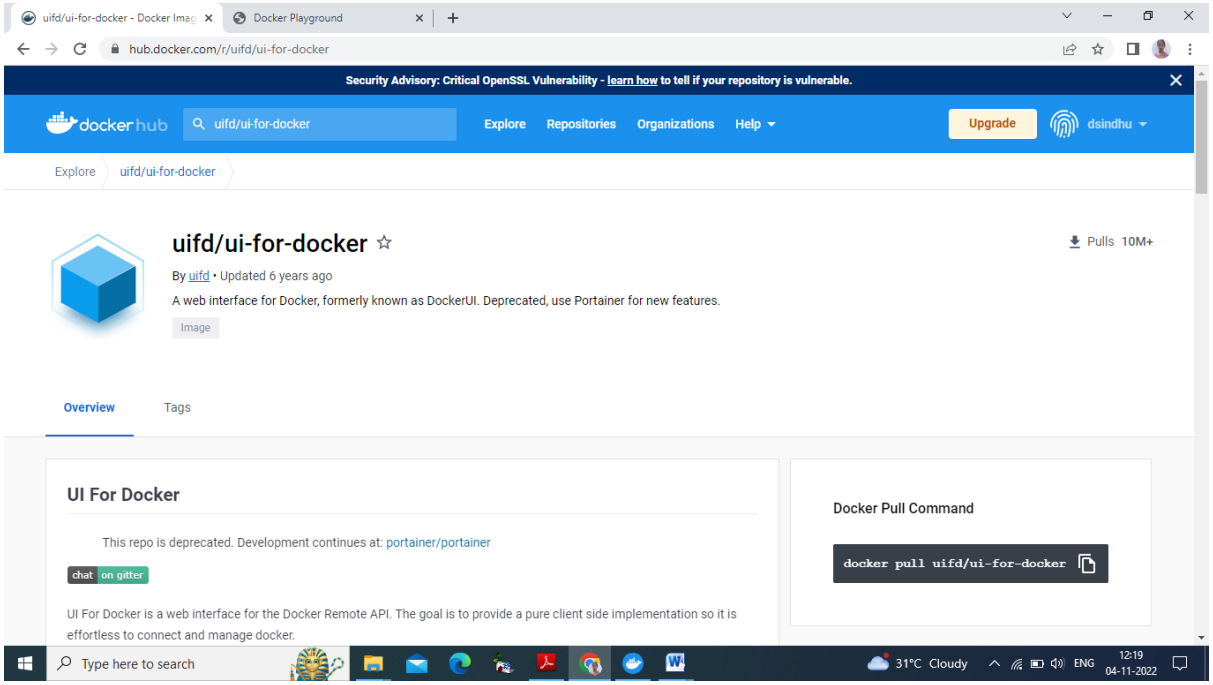
## 1)PULL AN IMAGE FROM DOCKER HUB AND RUN IT IN DOCKER BACKGROUND

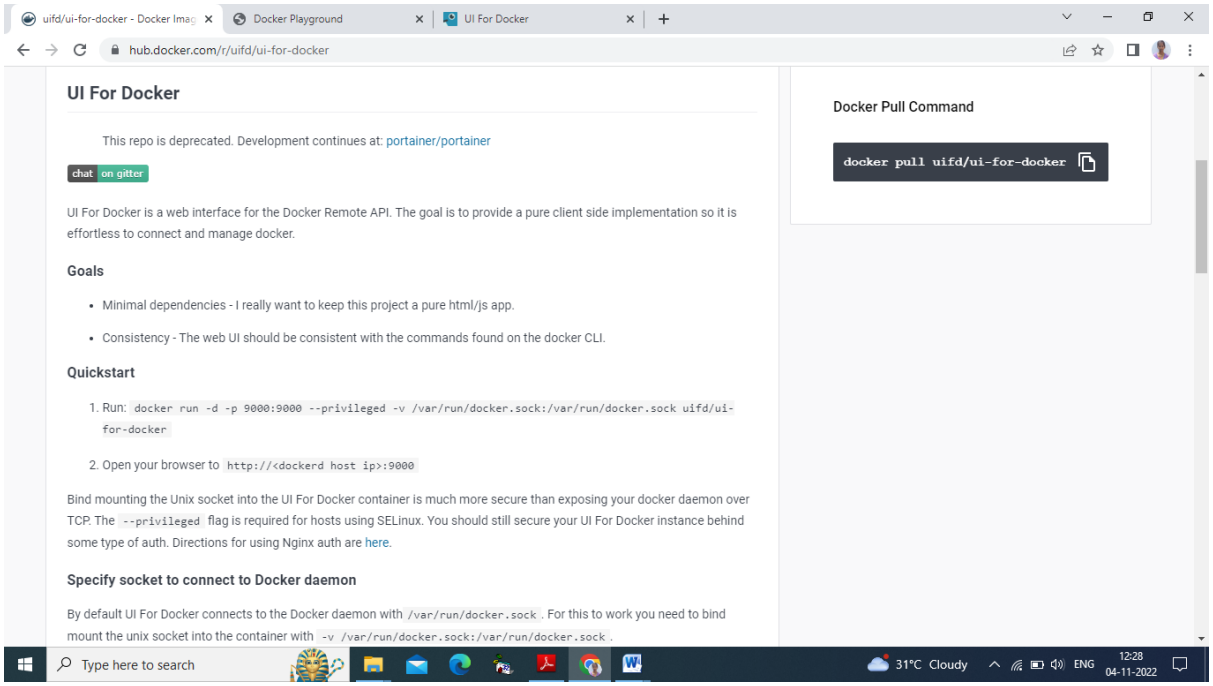
**COMMANDS:**

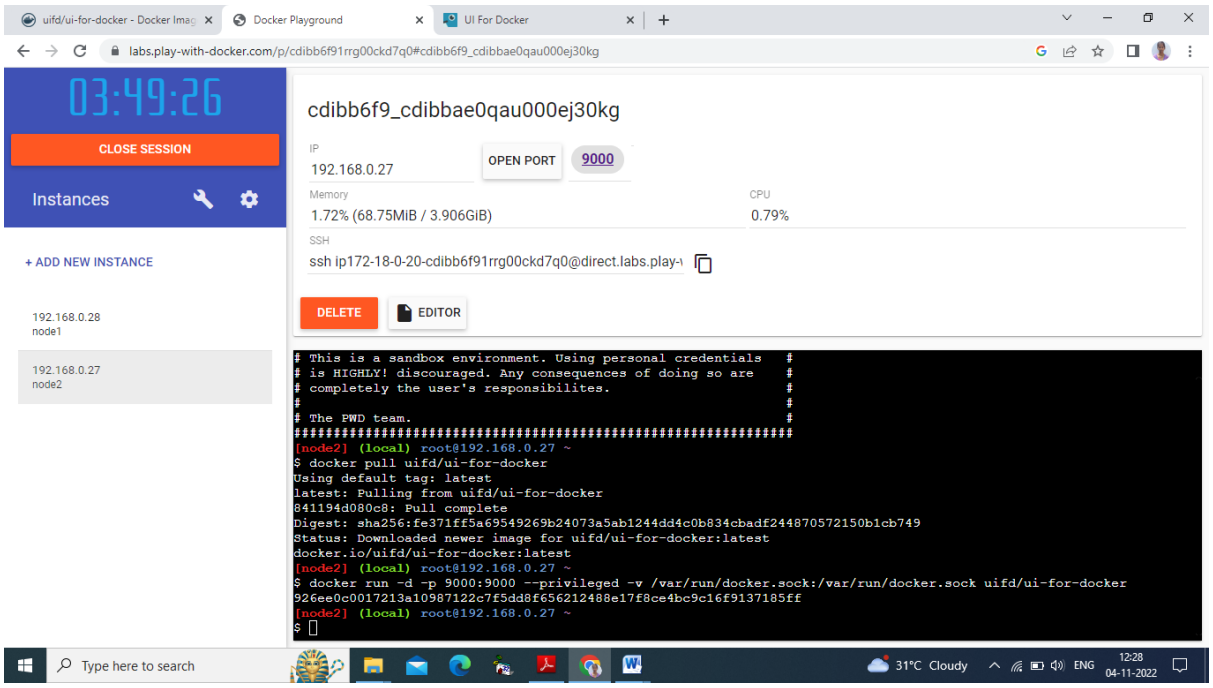
docker pull uifd/ui-for-docker

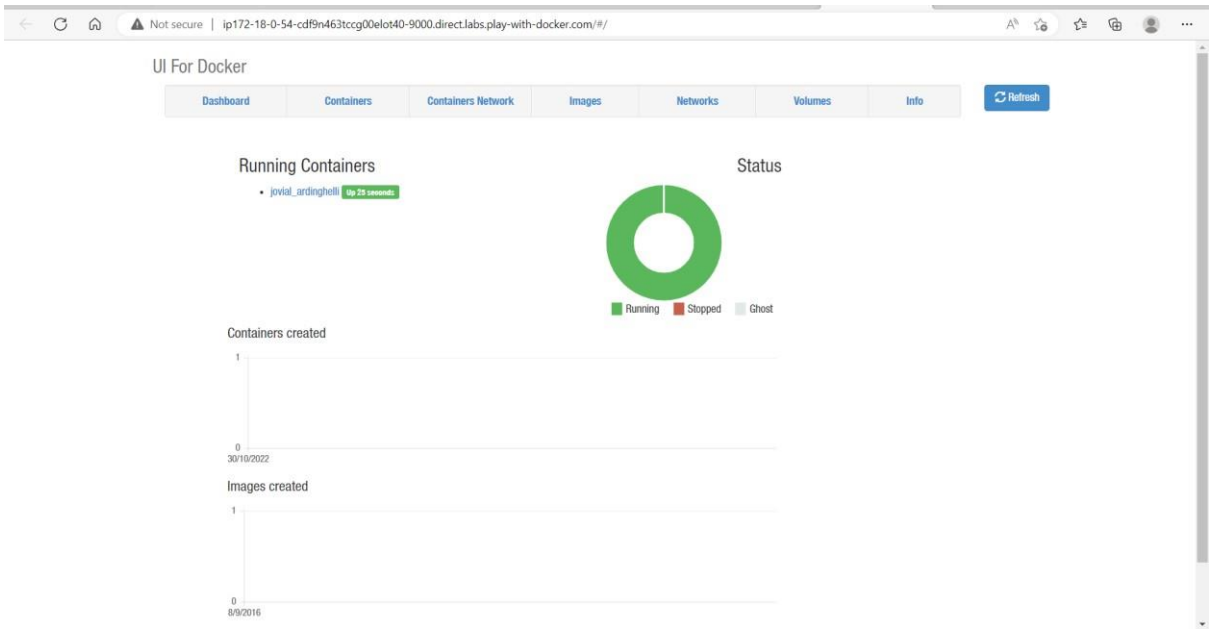
docker run -d -p 9000:9000 --privileged -v /var/run/docker.sock:/var/run/docker.sock uifd/ui-for-docker

**OUTPUT:**




uifd/ui-for-docker - Docker Imag
Docker Playground
UI For Docker
hub.docker.com/r/uifd/ui-for-docker
UI For Docker
This repo is deprecated. Development continues at: portainer/portainer
chat on gitter
UI For Docker is a web interface for the Docker Remote API. The goal is to provide a pure client side implementation so it is effortless to connect and manage docker.
Goals
Minimal dependencies - I really want to keep this project a pure html/js app.
Consistency - The web UI should be consistent with the commands found on the docker CLI.
Quickstart
1. Run: docker run -d -p 9000:9000 --privileged -v /var/run/docker.sock:/var/run/docker.sock uifd/ui-for-docker
2. Open your browser to http://<dockerd host ip>:9000
Bind mounting the Unix socket into the UI For Docker container is much more secure than exposing your docker daemon over TCP. The --privileged flag is required for hosts using SELinux. You should still secure your UI For Docker instance behind some type of auth. Directions for using Nginx auth are here.
Specify socket to connect to Docker daemon
By default UI For Docker connects to the Docker daemon with /var/run/docker.sock . For this to work you need to bind mount the unix socket into the container with -v /var/run/docker.sock:/var/run/docker.sock .
Docker Pull Command
docker pull uifd/ui-for-docker
Type here to search
31°C Cloudy
ENG
12:28
04-11-2022


uifd/ui-for-docker - Docker Imag
Docker Playground
UI For Docker
labs.play-with-docker.com/p/cdibb6f91rrg00ckd7q0#cdibb6f9_cdibbae0qau000ej30kg
03:49:26
CLOSE SESSION
Instances
+ ADD NEW INSTANCE
192.168.0.28
node1
192.168.0.27
node2
cdibb6f9_cdibbae0qau000ej30kg
IP
192.168.0.27
OPEN PORT
9000
Memory
1.72% (68.75MiB / 3.906GiB)
CPU
0.79%
SSH
ssh ip172-18-0-20-cdibb6f91rrg00ckd7q0@direct.labs.play-
DELETE
EDITOR
# This is a sandbox environment. Using personal credentials #
# is HIGHLY! discouraged. Any consequences of doing so are #
# completely the user's responsibilites. #
# #
# The PWD team. #
[node2] (local) root@192.168.0.27 ~
$ docker pull uifd/ui-for-docker
Using default tag: latest
latest: Pulling from uifd/ui-for-docker
841194d080c8: Pull complete
Digest: sha256:fe371ff5a69549269b24073a5ab1244dd4c0b834cbadf244870572150b1cb749
Status: Downloaded newer image for uifd/ui-for-docker:latest
docker.io/uifd/ui-for-docker:latest
[node2] (local) root@192.168.0.27 ~
$ docker run -d -p 9000:9000 --privileged -v /var/run/docker.sock:/var/run/docker.sock uifd/ui-for-docker
926ee0c0017213a10987122c7f5dd8f656212488e17f8ce4bc9c16f9137185ff
[node2] (local) root@192.168.0.27 ~
$
Type here to search
31°C Cloudy
ENG
12:28
04-11-2022



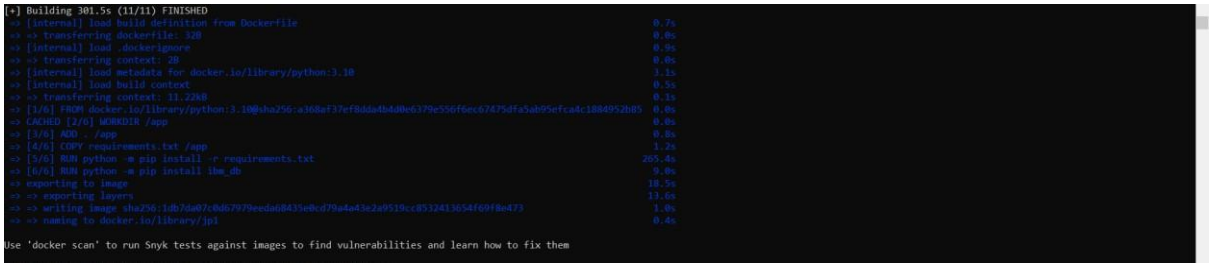

## 2)CREATE A DOCKER FILE FOR THE JOB PORTAL APPLICATION AND DEPLOY IT IN DOCKER DESKTOP APPLICATION.

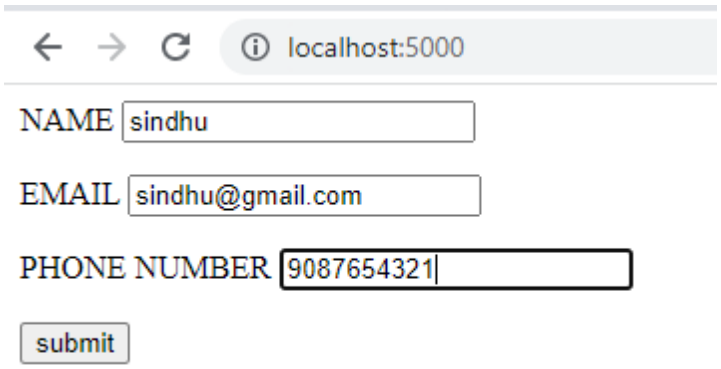
**COMMANDS:**

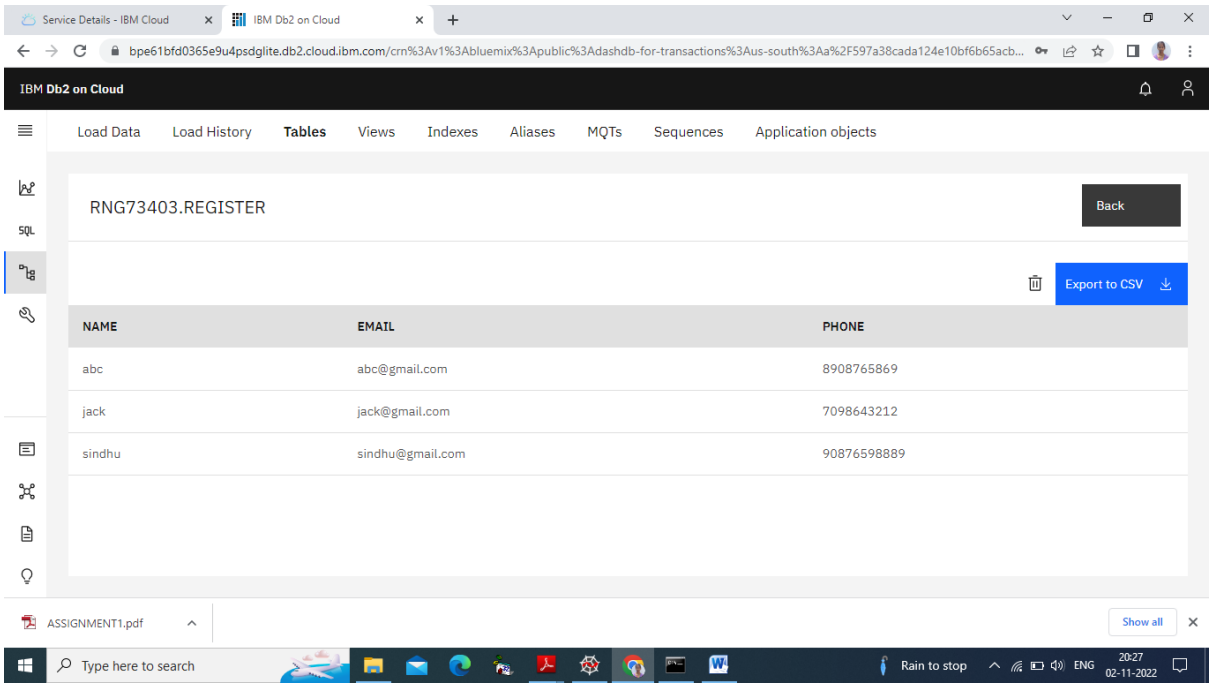
docker build -t jp1 .

docker run -p 5000:5000 jp1

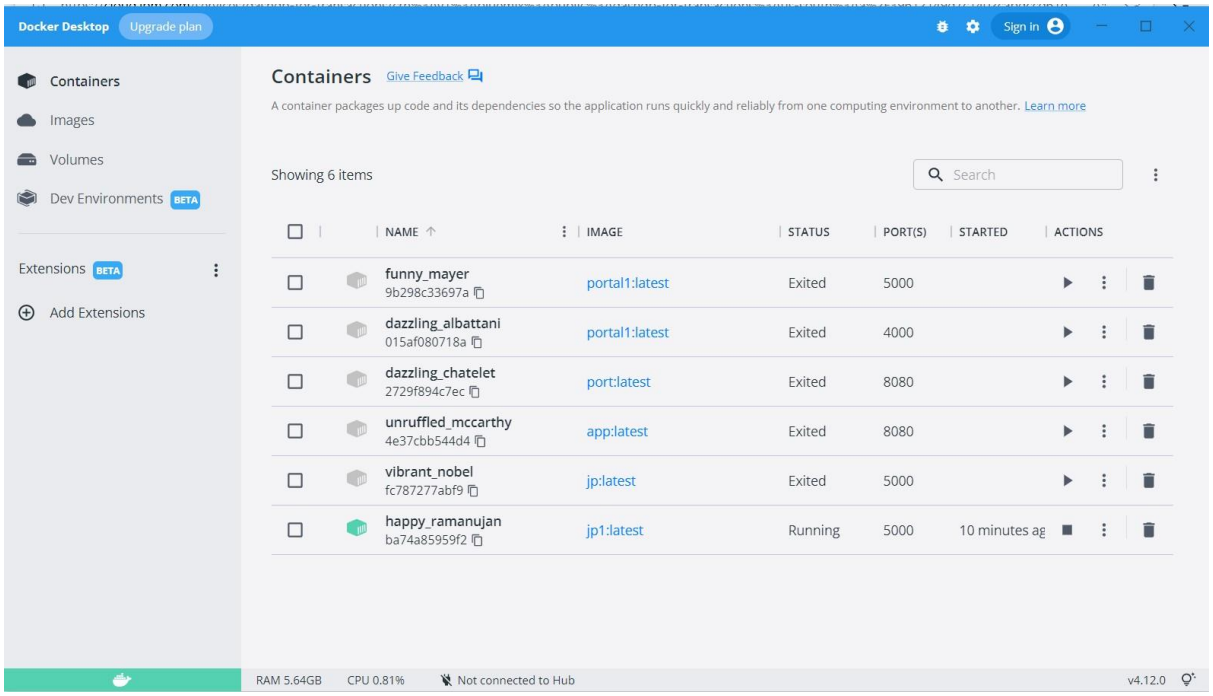
**OUTPUT:**



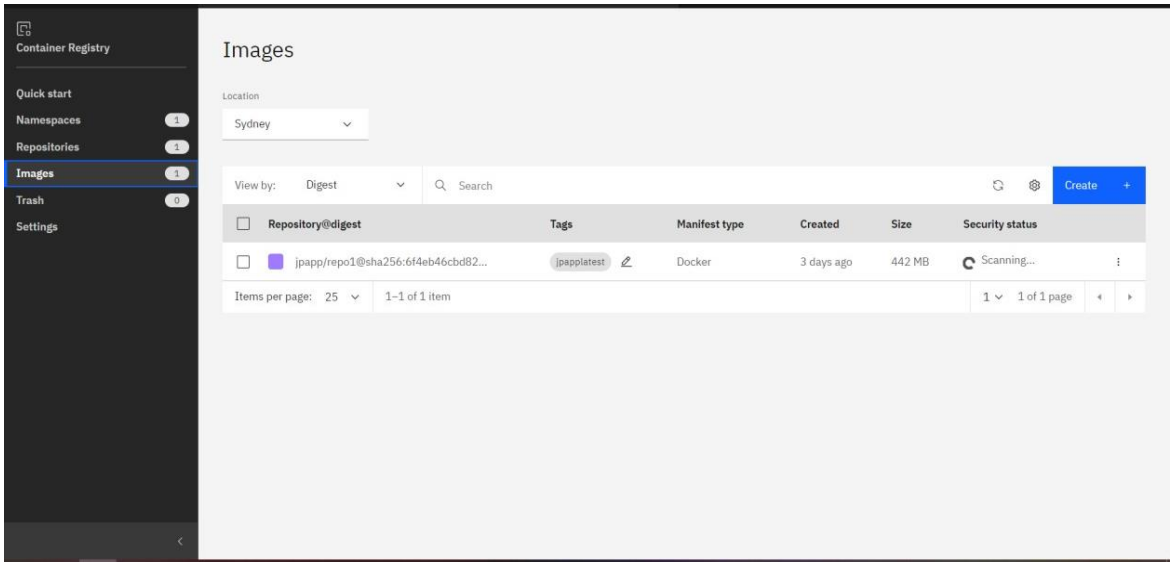

localhost:5000
NAME sindhu
EMAIL sindhu@gmail.com
PHONE NUMBER 9087654321
submit

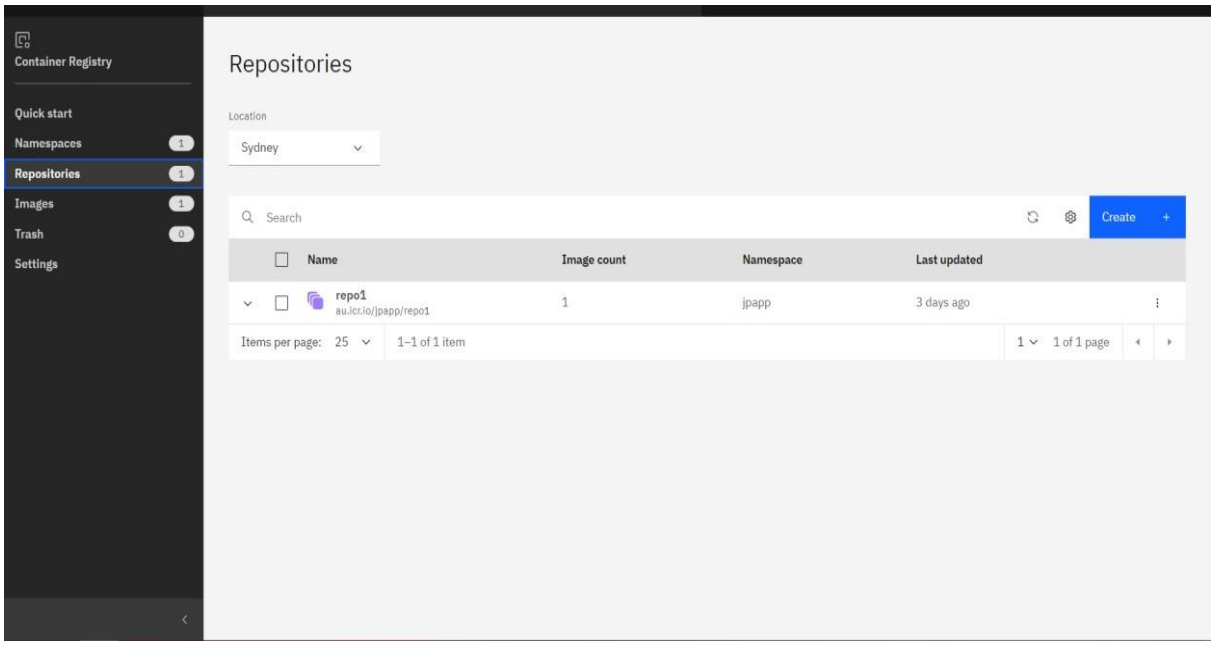

Service Details - IBM Cloud
IBM Db2 on Cloud
IBM Db2 on Cloud
Load Data
Load History
Tables
Views
Indexes
Aliases
MQTs
Sequences
Application objects
RNG73403.REGISTER
Back
Export to CSV
NAME
EMAIL
PHONE
abc
abc@gmail.com
8908765869
jack
jack@gmail.com
7098643212
sindhu
sindhu@gmail.com
90876598889
ASSIGNMENT1.pdf
Show all
Type here to search
Rain to stop
ENG
20:27
02-11-2022

localhost:5000/
Hello

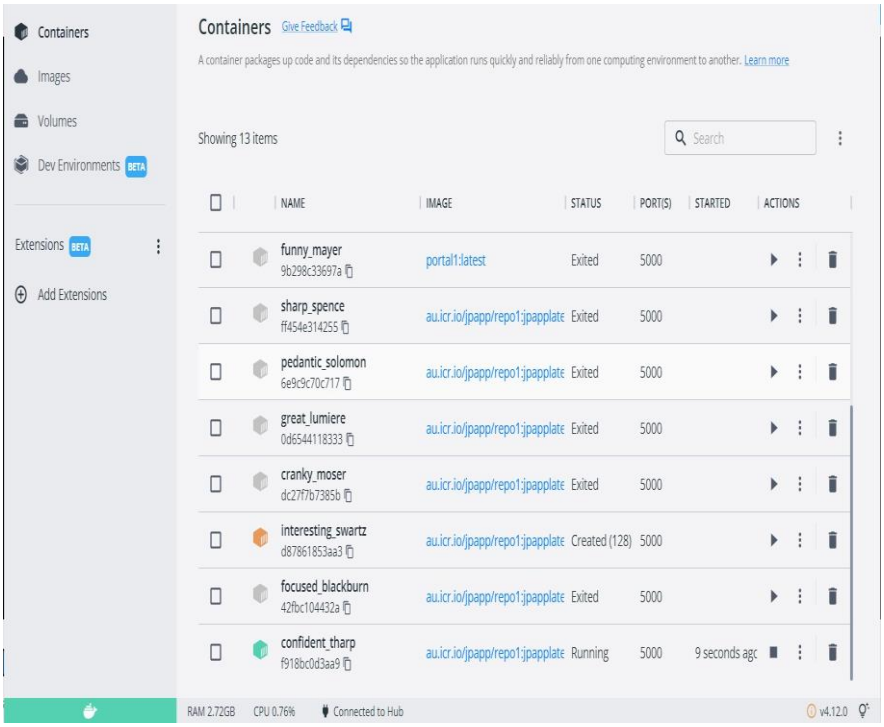

Docker Desktop
Upgrade plan
Sign in
Containers
Images
Volumes
Dev Environments
BETA
Extensions
BETA
Add Extensions
Containers
Give Feedback
A container packages up code and its dependencies so the application runs quickly and reliably from one computing environment to another. Learn more
Showing 6 items
Search
NAME
IMAGE
STATUS
PORT(S)
STARTED
ACTIONS
funny_mayer
9b298c33697a
portal1:latest
Exited
5000
dazzling_albattani
015af080718a
portal1:latest
Exited
4000
dazzling_chatelet
2729f894c7ec
port:latest
Exited
8080
unruffled_mccarthy
4e37cbb544d4
app:latest
Exited
8080
vibrant_nobel
fc787277abf9
jp:latest
Exited
5000
happy_ramanujan
ba74a85959f2
jp1:latest
Running
5000
10 minutes ag
RAM 5.64GB
CPU 0.81%
Not connected to Hub
v4.12.0

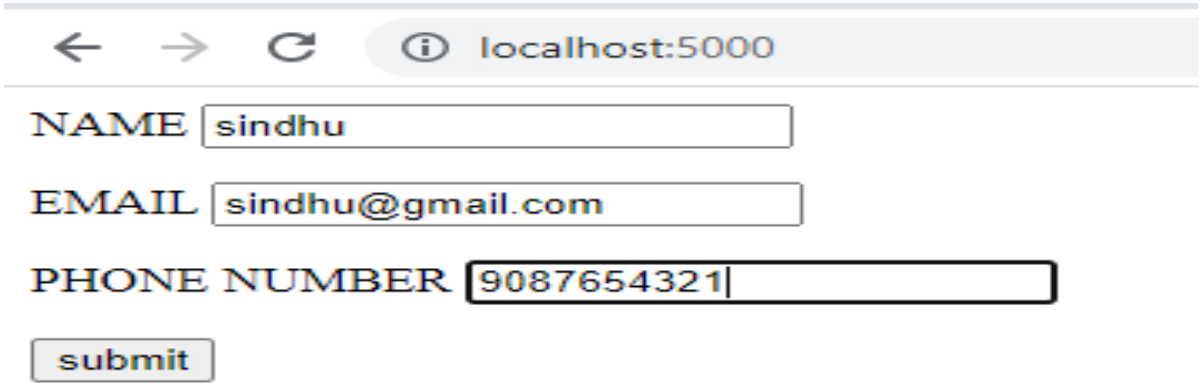
## 3)CREATE AN IBM CONTAINER REGISTRY AND DEPLOY HELLOWORLD APP OR JOBPORTAL APP



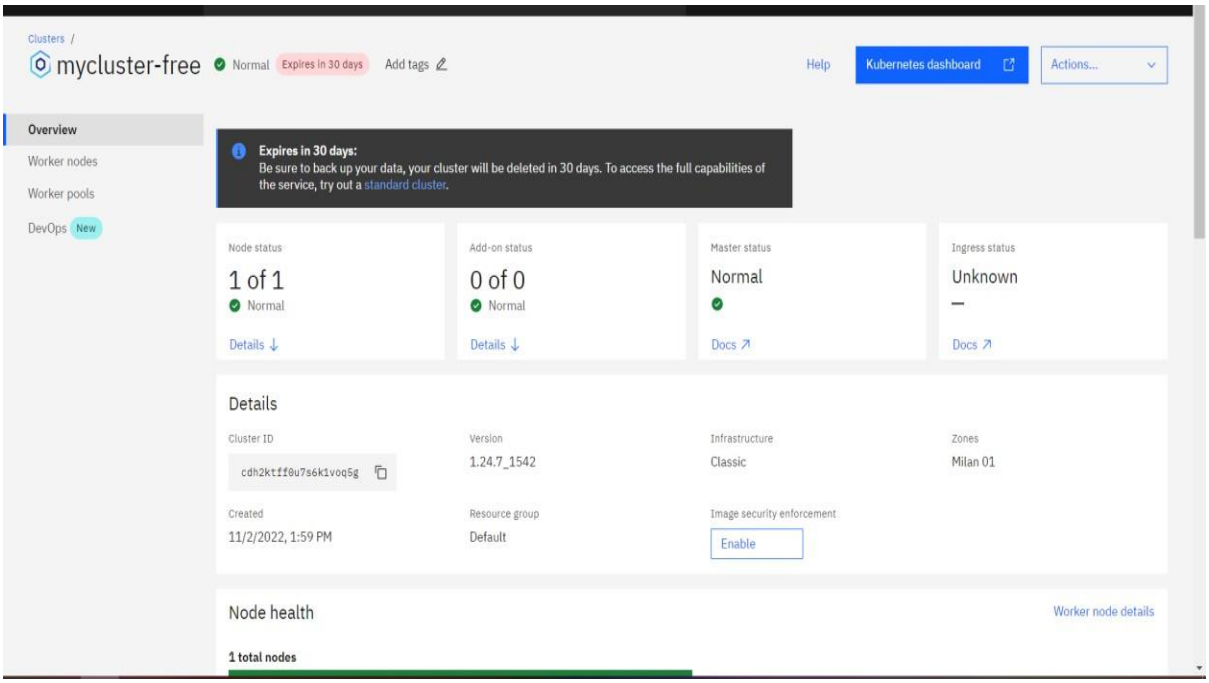

```
Using default tag: latest
The push refers to repository [au.icr.io/jpapp/repo1]
848973eaa72c: Layer already exists
875c3fdf2253: Layer already exists
76025884cfd0: Layer already exists
f2939f19badd: Layer already exists
aec753ddd83a: Layer already exists
f5d161bba139: Layer already exists
1569e0d95ce6: Layer already exists
d9e08da15d0c: Layer already exists
6b183c62e3d7: Layer already exists
382fd36bfd35: Layer already exists
d1dec9917839: Layer already exists
d38adf39e1dd: Layer already exists
4ed121b04368: Layer already exists
d9d07d703dd5: Layer already exists
latest: digest: sha256:6f4eb46cbd82b126771051a7915c8f41324c611a97857a6a82a9f450c25ec7f0 size: 3259
```



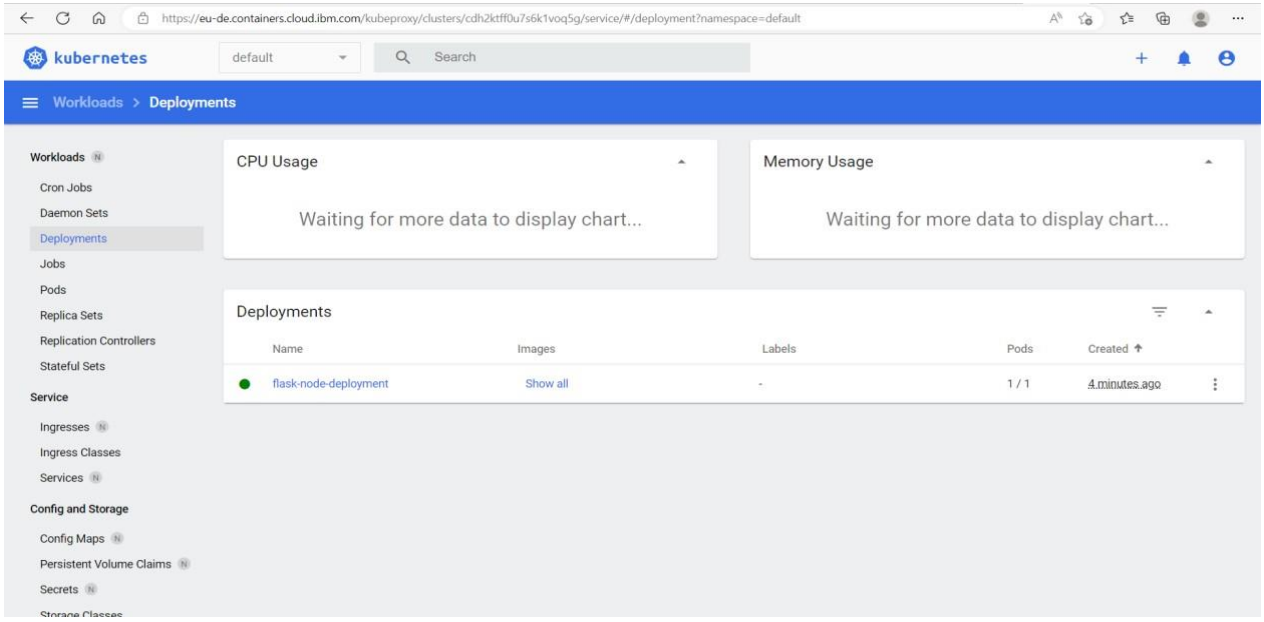



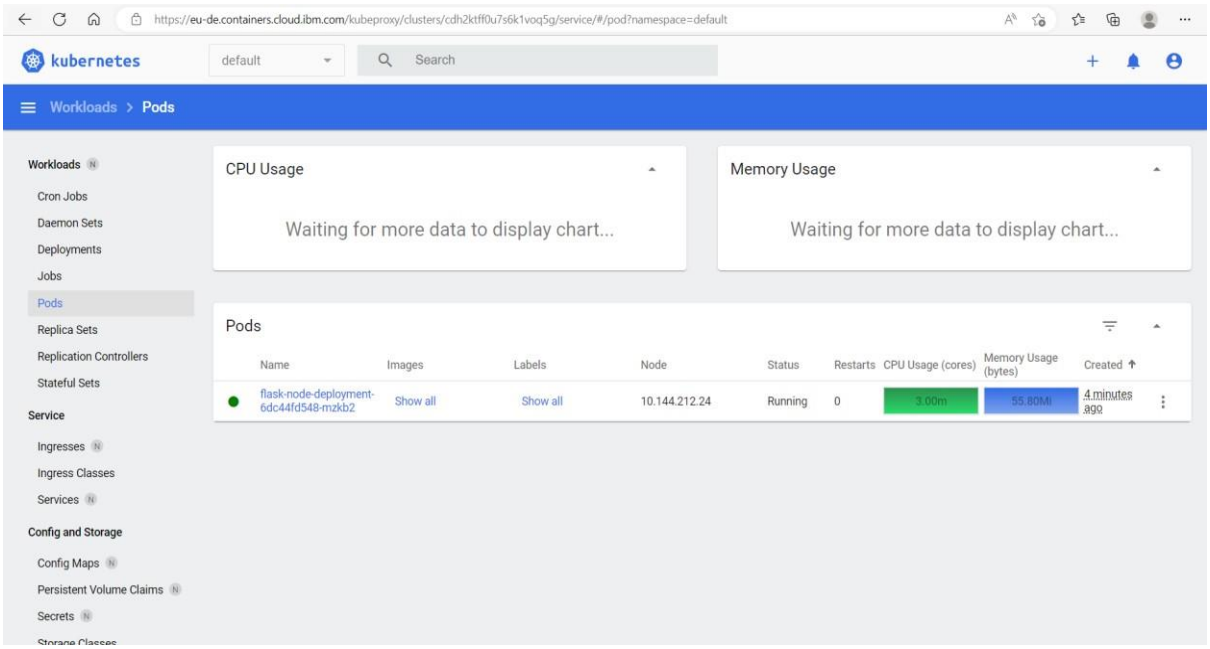

**4) CREATE A KUBERNETES CLUSTER IN IBM CLOUD AND DEPLOY HELLO WORLD IMAGE OR JOB PORTAL IMAGE AND ALSO EXPOSE THE SAME APP TO RUN IN A NODEPORT.**

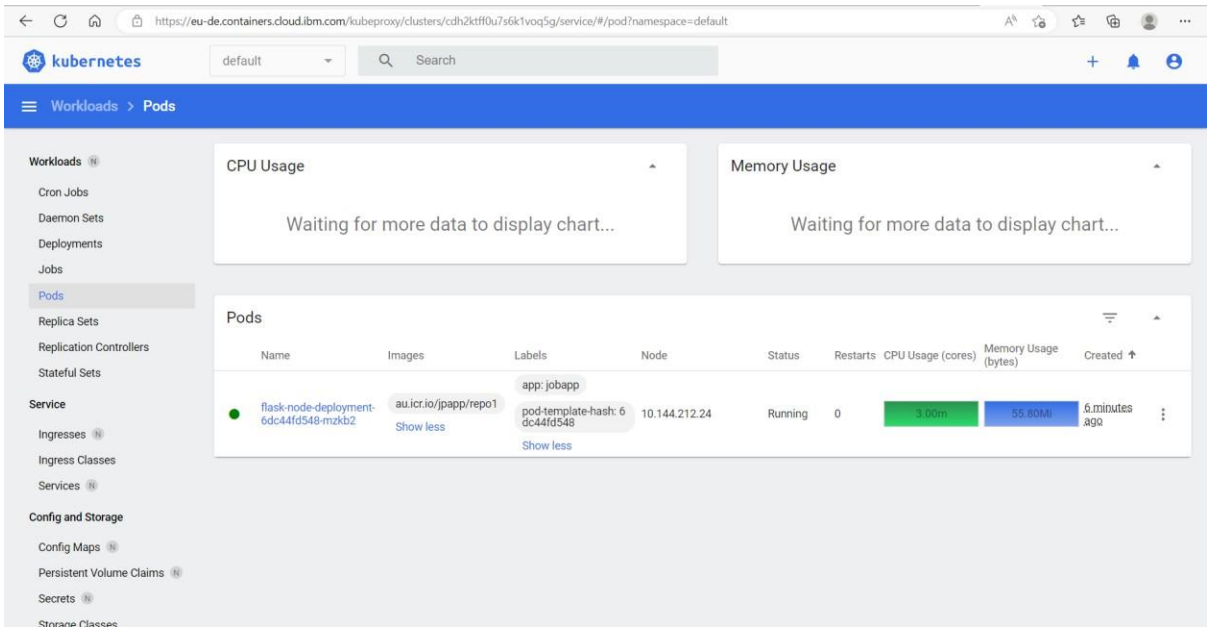
**KUBERNETES CLUSTER CREATION:**

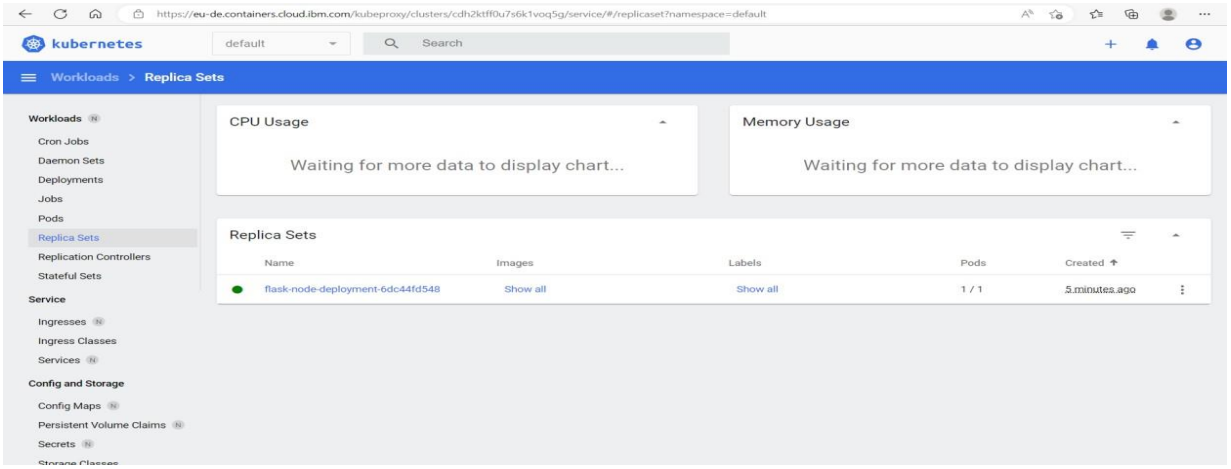


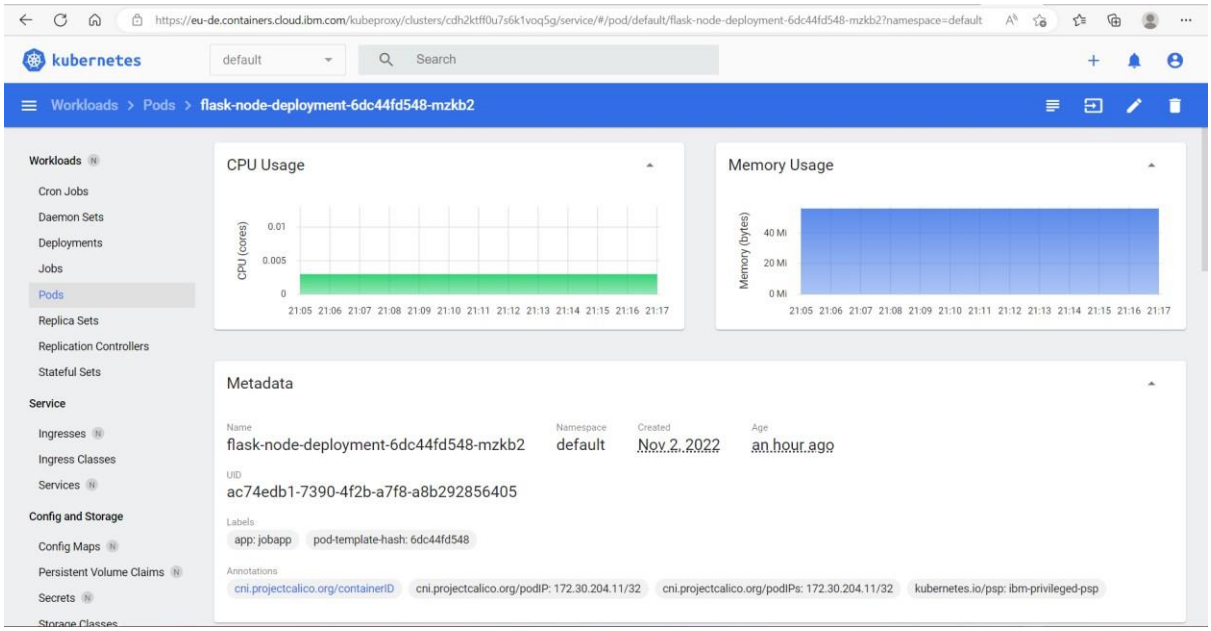

**DEPLOYMENT OF JOB PORTAL APP IN CLUSTER**




https://eu-de.containers.cloud.ibm.com/kubeproxy/clusters/cdh2ktff0u7s6k1voq5g/service/#/pod?namespace=default
kubernetes
default
Search
Workloads > Pods
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
CPU Usage
Waiting for more data to display chart...
Memory Usage
Waiting for more data to display chart...
Pods
Name
Images
Labels
Node
Status
Restarts
CPU Usage (cores)
Memory Usage (bytes)
Created
flask-node-deployment-6dc44fd548-mzkb2
Show all
Show all
10.144.212.24
Running
0
3.00m
55.80Mi
4 minutes ago

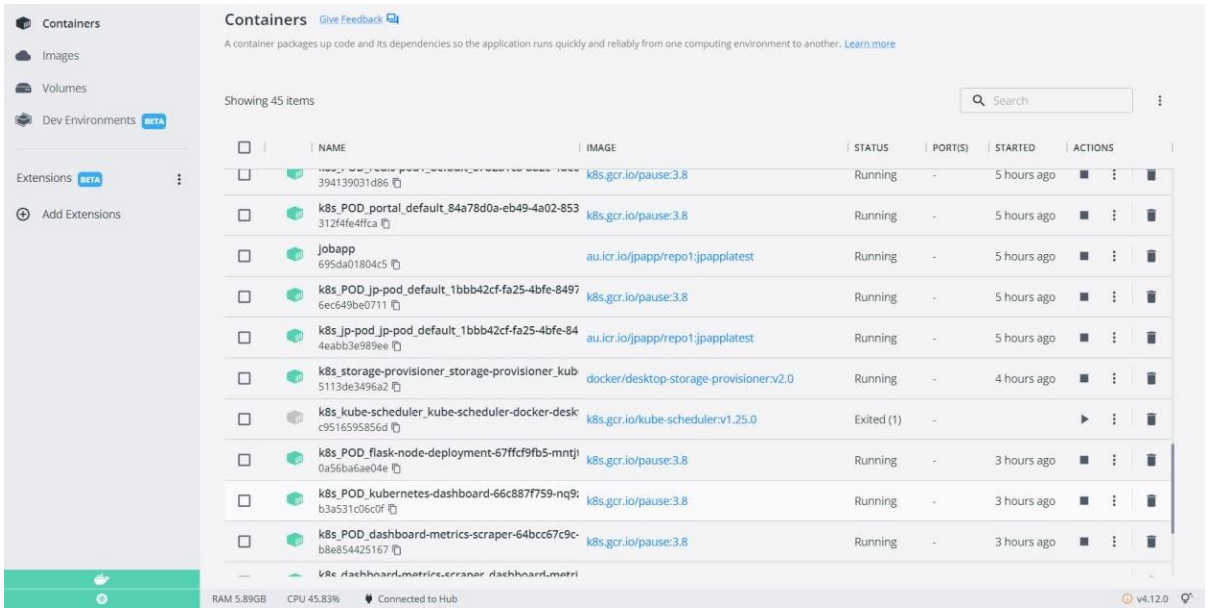

https://eu-de.containers.cloud.ibm.com/kubeproxy/clusters/cdh2ktff0u7s6k1voq5g/service/#/pod?namespace=default
kubernetes
default
Search
Workloads > Pods
Workloads
Cron Jobs
Daemon Sets
Deployments
Jobs
Pods
Replica Sets
Replication Controllers
Stateful Sets
Service
Ingresses
Ingress Classes
Services
Config and Storage
Config Maps
Persistent Volume Claims
Secrets
CPU Usage
Waiting for more data to display chart...
Memory Usage
Waiting for more data to display chart...
Pods
Name
Images
Labels
Node
Status
Restarts
CPU Usage (cores)
Memory Usage (bytes)
Created
flask-node-deployment-6dc44fd548-mzkb2
au.icr.io/jpapp/repo1
Show less
app: jobapp
pod-template-hash: 6dc44fd548
Show less
10.144.212.24
Running
0
3.00m
55.80Mi
6 minutes ago


https://eu-de.containers.cloud.ibm.com/kubeproxy/clusters/cdh2ktff0u7s6k1voq5g/service/#/replicaset?namespace=default
kubernetes
default
Search
Workloads > Replica Sets
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
CPU Usage
Waiting for more data to display chart...
Memory Usage
Waiting for more data to display chart...
Replica Sets
Name
Images
Labels
Pods
Created
flask-node-deployment-6dc44fd548
Show all
Show all
1 / 1
5 minutes ago




```
Name:              flask-node-deployment
Namespace:         default
Labels:            <none>
Annotations:       <none>
Selector:          app=flasknode
Type:              ClusterIP
IP Family Policy:  SingleStack
IP Families:       IPv4
IP:                172.21.233.153
IPs:               172.21.233.153
Port:              <unset>  5000/TCP
TargetPort:        5000/TCP
Endpoints:         <none>
Session Affinity:  None
Events:            <none>
```

```
NAME                  TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)    AGE
flask-node-deployment   ClusterIP   172.21.233.153   <none>        5000/TCP   68m
kubernetes              ClusterIP   172.21.0.1       <none>        443/TCP    6h58m
Name:              flask-node
Namespace:         default
Labels:            <none>
Annotations:       <none>
Selector:          app=jobapp
Type:              ClusterIP
IP Family Policy:  SingleStack
IP Families:       IPv4
IP:                172.21.30.184
IPs:               172.21.30.184
Port:              <unset>  80/TCP
TargetPort:        8008/TCP
Endpoints:         172.30.204.11:8008
Session Affinity:  None
Events:            <none>
```

## EXPOSE APP TO NODEPORT:

```
Name:                     flask
Namespace:                default
Labels:                   <none>
Annotations:              <none>
Selector:                 app=jobapp
Type:                     NodePort
IP Family Policy:         SingleStack
IP Families:              IPv4
IP:                       172.21.15.156
IPs:                      172.21.15.156
Port:                     <unset>  80/TCP
TargetPort:               8008/TCP
NodePort:                 <unset>  32153/TCP
Endpoints:                172.30.204.11:8008
Session Affinity:         None
External Traffic Policy:  Cluster
Events:                   <none>
```