

Assignment-4

Cloud App Development

| | |
|---------------------|------------------|
| Assignment Date | 7 September 2022 |
| Student Name | Deeparachana.K |
| Student Roll Number | 812619104009 |
| Maximum Marks | 2 Marks |

Questions:

- 1.Pull an Image from docker hub and run it in docker playground.
- 2.Create a docker file for the jobportal application and deploy it in Docker desktop application.
- 3.Create a IBM container registry and deploy helloworld app or jobportalapp.
- 4.Create a Kubernetes cluster in IBM cloud and deploy helloworld image or jobportal image and also expose the Image and also expose the same app to run in nodeport.

Dockerfile

FROM python:3.10-buster

WORKDIR /app

COPY ..

RUN pip install --no-cache-dir -r requirements.txt

CMD ["gunicorn", "--bind", "0.0.0.0:5000", "app:app"]

App.py

```
import uuid

from flask import Flask

instance_id = uuid.uuid4().hex

app = Flask(__name__)


@app.route("/")
def get_instance_id():

    return f"<b style='font-size:30px;color:red;'>Instance ID: {instance_id}</b>"


if __name__ == "__main__":

    app.run(port=5000,host="0.0.0.0")
```

desktop.yml

```
apiVersion: v1

clusters:

- cluster:

    certificate-authority-data:
LS0tLS1CRUdJTlBDRVJUSUZJQ0FURS0tLS0tCk1JSUMvakNDQWVhZ0F3SUJBZ0lCQURBTkJ
na3Foa2lHOXcwQkFRc0ZBREFWTVJNd0VRWURWUVFERXdwcmlRSmwKY201bGRHVnpNQj
RYRFRJeU1UQXNdOVEUxTWpjME9Gb1hEVE15TVRBd01qRTFNaWwT0Zvd0ZURVRNQkVHQT
FVRQpBeE1LYTNWaVpYSnVaWFJsY3pDQ0FTSXdEUVlKS29aSWh2Y05BUUVCQlFBRGdnRVB
BRENDQVFvQ2dnRUJBS3J5CnM2cC9iZTRBaTIPNmY3VjBuRHFYU0R0ai9oQmlyWEhIT2dISHJ
```

kNWs0V1NjRTVzeUtWdERydUx3MVFrUEJVVkwKenlNWWQ2OEFsRDdnaGxpYzRhUnFJV09v
OVNSVEVSaUo0RDJqZWJGWFR0d1BpZGFEUFIEVm85UIFBS0hBQXliMQpSZmMvT2FkaEEvb
2FHNGxLSFRrQUIKV1JlclFCbThTWfExTlhkemtVVTv6NElJbE5CNm9YcUtmbGJpbDBxMG1UC
lkxSndWQ3pIVFVxcXo1OUFNvkFvQ01RQkhXcWZ5enQyODVkoVN2YVpyUTF3VVBqY2Zzc0Zx
TUNPeUx3MzI5dkMKVlhienFLMDA5bzVkbWsvZXZJQUxSNmRKL3NNMTg5T2k4bWFVQ1NL
MzVUU2xJNVI0M2IML3l1UVd4aVFUb0Rwawo3OU9wQjJZQ25DQTdRZWhxR3JFQ0F3RUFBY
U5aTUZjd0RnWURWUjBQqVFIL0JBUURBZ0trTUE4R0ExVWRFd0VCCi93UUZNQU1CQWY4d0h
RWURWUjBPQkJZRUZNZHhDVVdRSTVGRmQvU0pwZEd0cUNVL25kMVINQIVHQTfVZEVRUU
8KTUF5Q0NtdDFZbVZ5Ym1WMFpYTXdEUVIKS29aSWh2Y05BUUVMQIFBRGdnRUJBRHIIYnJ
Xa3ZITDU1ZjRQaFI5UwpycER2UnQwZ1IMZHZ4RXJSTkJPnFFGSUJXTUhxRDFTMIVITWNzME
NnUGpQQ3ptWnUybzNCNkp4eUpkWUwrRWtjCjJQYkFuWXdpQ3AzaEwxN2NjcXZ1bXNDcng3
MVpGdElNNFhtTmd0bjFiTnpWR3QvMzdWaWNsNjdsTFVydnP0RUcKOFcwMVRoQ2hwVlJiQm
ptYStOZ295SGNuNFJZY0dJTHZFTVNQa1BZRXI0VTN5MmtXRkgwaElMMlZRYmJzUIYxTApJ
OEIVM3VQZEV3ZktLN0dNeW85VGoxc3NsRWVBTnFUeWVUYzg1MHIZYThYekxVTzloM1kwV
Wc4a2c1NlFaZVJUCnE1a2lGQ3p6OGRjZi9xN2tHN0RBaENvcFhhc3VQYIV5TnNvQVRBOTVRd
W83MEQzdDNCQ1F2OTIvV2NjNnBKRW0KVUtrPQotLS0tLUVORCBDRVJUSUZJQ0FURS0tLS0
tCg==

server: https://kubernetes.docker.internal:6443

name: docker-desktop

contexts:

-context:

cluster: docker-desktop

user: docker-desktop

name: docker-desktop

current-context: docker-desktop

kind: Config

preferences: {}

users:

-name: docker-desktop

user:

client-certificate-data:
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSURRakNDQWlxZ0F3SUJBZ0lJU3lNR3o4K
1lDK013RFFZSkvWklodmNOQVFFTEJRQXdGVEVUTUJFR0ExVUUKQXhNS2EzVmllaWEp1Wlh
SbGN6QWVGdzB5TWpFd01EVXhOVEkzTkRoYUZZ3MHINekV3TURreE16TXdNVFJhTURZeApG
ekFWQmdOVk1JbB1REbk41YzNSbGJUcHRZWE4wWlhhKek1Sc3dHUVIEVlFRREV4SmtiMk5yWl
hJdFptOXIMV1JsCmMydDBiM0F3Z2dFaU1BMEdDU3FHU0liM0RRRUJBUVVBQTRJQkR3QXd

nZ0VLQW9JQkFRRHR2T1VUbElOZmw3MG8KK1hpR1o1d1VLbVZ6dnFP0W9ycXB40E15aXVi
MUQ0dXkyMHIScUhPTEVwRHkxalpPV0szUHBGTUxGcXQ3QmZDLwpVUGJPN3pGbXBYQ2Vu
VkhmSThqRmRSRUlnQ0U3a2dJNGdqMzNzRlVWbklclhUbIFZQWpqS3lIRkYrTFphbyt3Ck5nU
TI4d0xiQTkrQIY5ZXJYMkZObHoxRXBnNi9CYWNIMDVaQnJGNGVhR25VdUpQTUI3L0NFVHVS
ajVJRW5KUnIKUW1YSnBJdVo4TXkzbnJseXVGZzNIYzNaNW91YU10c3B4NEtBSXRyMXhwZU
xLU21VTGNISTBIUkhLZEVmM3UwRgp5akNaWEFoZjRIREo2MitJUXZ2UmJ3Y0x3eGFYRytMTI
VNMU5zdHhpamtOSE9PSUZ1R1JFcIIISRFIPTHByVk55CjhaMEhCelZGQWdNQkFBR2pkVEJ6T
UE0R0ExVWREd0VCL3dRRUF3SUZvREFUQmdOVkhTVUVEREFLQmdnckJnRUlYKQlFjREFqQU1
CZ05WSFJNQkFmOEVBakFBTUI4R0ExVWRJd1FZTUJhQUZNZHHdVdVdRSTVGRmQvU0pwZEd
0cUNVLwpuZDFZTUlwR0ExVWRFUVFXTUJTQ0VtUnZZMnRsY2kxbWlzSXRaR1Z6YTNSdmNE
QU5CZ2txaGtpRzl3MEJBUXNGCkFBT0NBUEVBSCtELzQ4NXkweWI4cVlxd3U4Q2k3K1phbklU
UVdlb1MvaWJObjQxNlpMdDZRRUxhZlVxTHo3S3EKcWhFamFGMkhvQ1RrTDAyUVR1WXIPaF
hFQIF1eTI0UkFkTml2d2VObW1qc3lyZ2Ric25ua1hCQld6RU80UzVySwpCbVlzOENxUWxFL1dj
dktXb0hmVTIvZU1obzYvQlIpc1gxQ1JKVVI3NGFNU2JzNTQybGs0UnQrb3lvRmxZVXU1CmJYT
WF3QUNhT0xoNUF5ZW9sS0xrOFV0aFg3VHIHTy9scVJWaWN5dlhvcXdPSFZXWENBNklzTFR
5SUVGUExxNGIKdlRjMFhmcFpyeWVYNDd1QVZyUHY1ODdib1B3WTNtdTArTTg5RzFwM0Ztd2
FDRIV1bGJJJaVZUNW9mWjRSS1VpOQpFTWWhhK0x4czBibTFFeUtrLzdZY3pJbE8yVjdkb3c9PQ
otLS0tLUVORCBDRVJUSUZJQ0FURS0tLS0tCg==

client-key-data:
LS0tLS1CRUdJTlBSU0EgUFJJVkFURSBLRVktLS0tLQpNSUIFb2dJQkFBS0NBUEVBVN2J6bEU1
U0RYNWU5S1BsNGhtZWNGQ3BsYzc2anZhSzZxY2ZETW9ybTIRK0xzdHRNCmthaHppeEtROH
RZMIRsaXR6NIJUQ3hhcmV3WHd2MUQyenU4eFpxVnducDFSM3lQSXhYVWJDSUFoTzVJQ09J
STkKOTdCVkZaeUlhMTA1MEdBSTR5c2h4UmZpMldxUHNEWUVQY2NDMndQZmdWZlhxMTlo
VFpjOVJLWU92d1duQjIPVwpRYXhlSG1ocDFMaVR6QWUvd2hFN2tZK1NCSnlVYTBKbHlU0xt
ZkRNdDU2NWNyaFIOeDNOMmVhTG1qTGJLY2VDCmdDTGE5Y2FYaXlrcGxDM0J5TklwUnluU
kg5N3RCY293bVZ3SVgrQnd5ZXR2aUVMNzBXOEhDOE1XbHh2aXpWRE4KVGJMY1lvNURSe
mppQmJoa1JLMkVRMklpNmExVGN2R2RCd2MxUIFJREFRQUJBb0lCQUhVWnV4aEc2ZUwzS
WZTQgoxUzBXOGRrNC9ObmpTRklMU3JpQUxoVENPSHJ6R2xPU0U0Z1htam9oZmdZdkgzYT
kxWGRXc0pUZ3FMeWpNMjk1CkdPamJCdEZSb2tSdXB6b3JMU1Nkd2pNV3NjZmU1bnNoSkJ
iUjQwY080RjlKLzBVMVdJdFUzb0J2Z0l5cElYZWgKVTD0K1FGalM4bytLVGpBYWZob3k2Ym5i
dFoyKzN0Nnhad3dqelJYcldIRW9DNVZRRm1tbXNKWnFXYYVWBeEZSUgpQdTFCEwPqGQnlvd1
MxVVRSMnhIV2xtb1J1Y3RGeS9tZDJNcXZ1bngzb2JHSHVdWxGUDNVdXhrbFErbE5LWDZvC
jY3SExCOWRKUDNrNWt1WjFROUNwclplNmhCSjZ2NHdSejdzC0luTkd3Q2Fra90NWtUVIR4bX
FIc0tEU0hXWWcKN2taZHowMENnWUVBnZBldW93MGdSRIRvWG5SS0hYQzM1aUJ2V3VjL2N
yeXRFKzZJaEo1c1pQT05OWG01dGNJRQoybEZ2NUU1MnRtUG5sOS9wcmNDL1IRUklrQldKbC
t1ZmpPdFRHaWVWUi81dTdlYWVsOU0xNDBvTUZpL2c4YzlkQjJJU3JCdW1oUUVXTXN2U2Nm
N2hXZDRrSnBDOE5rMnNpemJSa1pWVVV6NWNOUNGxYV1lIQ0pCMWNDZ1IFQS9sbWcKV3o5
UFdqVW95cTlMaWdGYVBiYnd2ajhtaHVkN3RVTmFjWk9rQWgxS05qTW9oK0phSmpLZIY4Mm
VRNIJ6TUxSVgpRbEV6TSs2N1NqZEpreGxPdGdNR3VPcDd5dUJ5bHM4dVRsK211UVFHYjZF
WIA1NkdrY2MvQTZhei9kME1RYVRJCKNRNnNOMDFuSUFQaXRTS2NCYTAvV0QvRkVwdGhhR
GJLUZlQaGtzTUNnWUJaWm1qalZyWWxrYXU0RHE3YVRSRUkKbERuR3R2dkh2S0xWL1p0WUI
YRFNUd1VGWtIOMUMyYVF3emlRKzZIS2g0MHgrMkR0b0VSZzFSN2Zab3lnREZMeQpiY0IBM

WFVNjN0OUd6aEwzR3R6WmJRTVREeFIyaEV3RHk1MnkyWHQ5ekR5RGFFMUlrdkVMZkZlejlvaXdpYWpaClBucjhPVXVyTy82Z3VhVjlKczZyeFFLQmdFNFVDVXU1ZFI5RmdRNG5lWk14c3hmNnpwQTRKb1dIRGN5bzdseXIKSzgrU1dqT2xKSzRCNkJhTnQ2aG1jRXpuY0l6azdyODYyZkw1Ti9raVc2MlNPVDFhVkhmelV5a0dHV2lrRm9McwpqZXVCdHRBUnJnUi9qMk8wdkQxTXNUZ2U0U0pOaVRKdWxLTXRXWmpVV0lXdUtDTDkrZEIIVdIUzNPWDIyOGJlCk5obXRBB0dBR3ZPeXR3emRIUGR0TzNscFowS3pibnF6NkQ3SHpHblYyaGdCWUNvTDdQaVpHVWMrdHdtOFJwVW4KU29PYmsvOVM1S3Y0NXNITnRjTjd5VS8wVjhpUUFwUmZkVDVSYnNjT3ZrUIFSQmo1VkJxa0pMVlppeS85QmJDbgpXNU1ZV0NqSi9lWHk0ZGVMDnZFckRJVHhDRFVYbkNWRU1ReUIXcDB4T2Y4b1lla2h6U009Ci0tLS0tRU5EIFJTQSBQUklWQVRFIEtFWS0tLS0tCg==

razorops.yaml

version:v2

tasks:

build:

runner: razorci/python:3.9

steps:

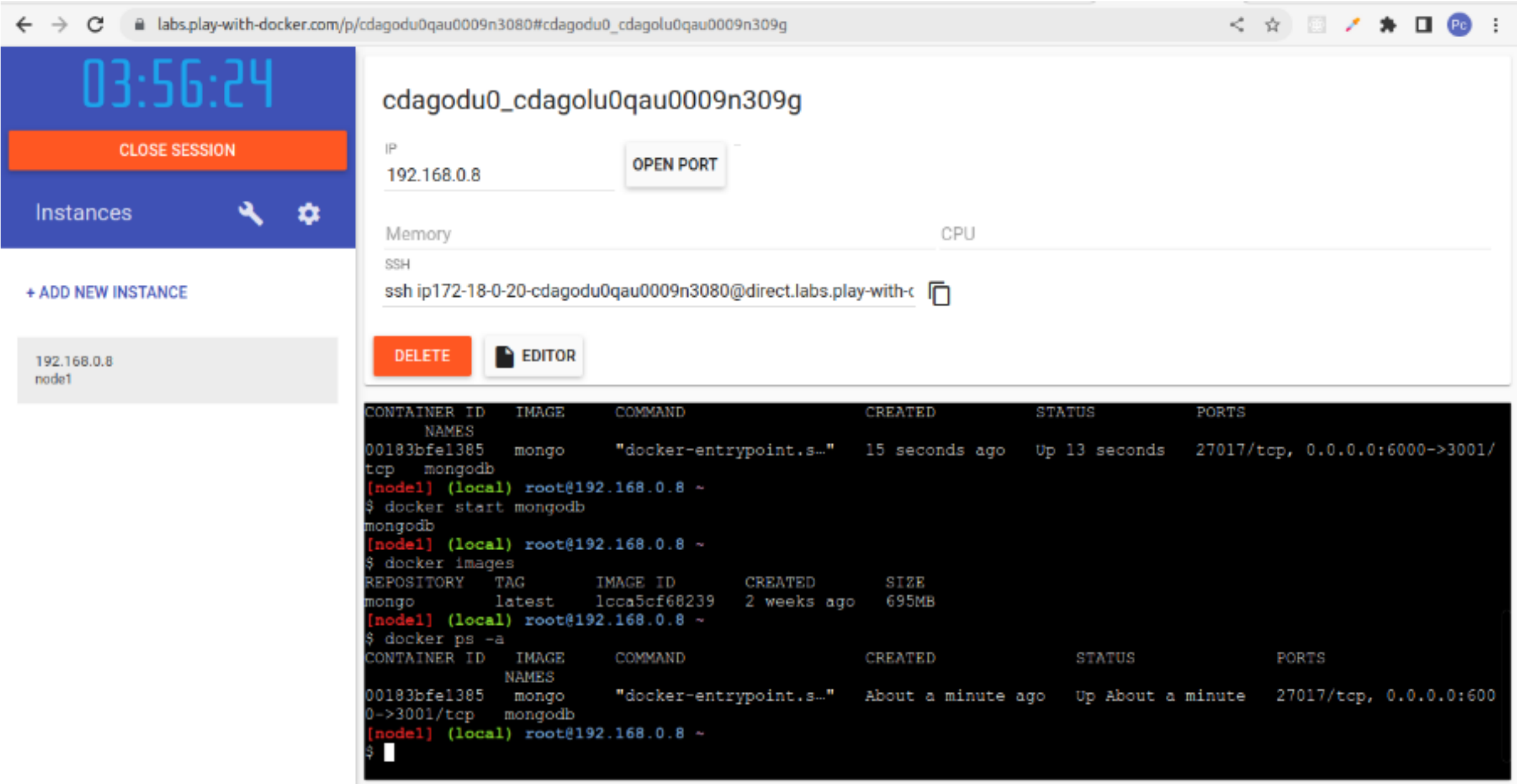
- checkout
- cache/pull: pip-deps-{{ checksum "requirements.txt" }}
- commands:
 - python-c 'import site; print(site.getsitepackages())'
 - pip install -r requirements.txt
- commands:
 - pythontest.py
- cache/push:
 - key: pip-deps-{{ checksum "requirements.txt" }}
- paths:
 - /usr/local/lib/python3.9/site-packages
- reports/junit: test-reports/*.xml

Requirements.txt

flask

gunicorn

Outputs:



03:53:15

CLOSE SESSION

Instances

+ ADD NEW INSTANCE

192.168.0.8
node1

cdagodu0_cdagolu0qau0009n309g

IP
192.168.0.8

OPEN PORT

MemoryCPU

SSH
ssh ip172-18-0-20-cdagodu0qau0009n3080@direct.labs.play-with-c

DELETEEDITOR

```
$ docker run -d -p6010:3000 hello-world
9b7c8f4adb31df3a7ceaf431f63e2eae868a579059402bc72b630b7e5e911875
^[[D(node1) (local) root@192.168.0.8 ~
$ docker run -d -p6020:3000 hello-world
6b0f1699a8ced253d9df780a42eef298a0c2fb2c6389e709681da2e63879b0b
(node1) (local) root@192.168.0.8 ~
$ docker run -p6120:3000 hello-world

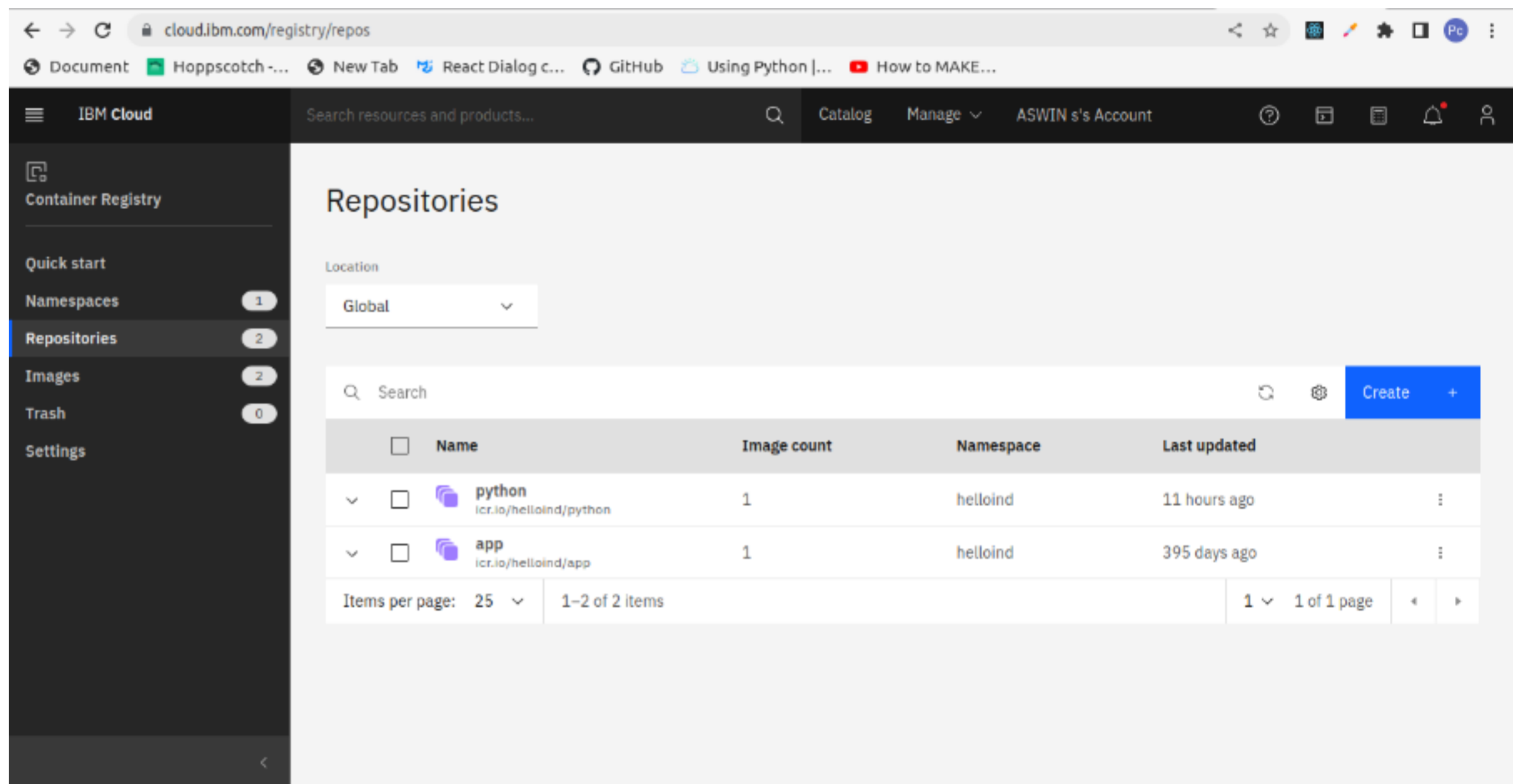
Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
```

root@admin: /home/ash/Desktop/Flask ClassWorrkroot@admin: /home/ash/Desktop/Flask ClassWorrk

Login Succeeded

Logging in with your password grants your terminal complete access to your account.
For better security, log in with a limited-privilege personal access token. Learn more at <https://docs.docker.com/go/access-tokens/>
root@admin: /home/ash/Desktop/Flask ClassWorrk# docker run -p5000:5000 flakapp:1.0
Unable to find image 'flakapp:1.0' locally
docker: Error response from daemon: pull access denied for flakapp, repository does not exist or may require 'docker login': denied: requested access to the resource is denied.
See 'docker run --help'.
root@admin: /home/ash/Desktop/Flask ClassWorrk# docker run -p5000:5000 -t flakapp:1.0
Unable to find image 'flakapp:1.0' locally
docker: Error response from daemon: pull access denied for flakapp, repository does not exist or may require 'docker login': denied: requested access to the resource is denied.
See 'docker run --help'.
root@admin: /home/ash/Desktop/Flask ClassWorrk# docker run -p5000:5000 -t flakapp
Unable to find image 'flakapp:latest' locally
docker: Error response from daemon: pull access denied for flakapp, repository does not exist or may require 'docker login': denied: requested access to the resource is denied.
See 'docker run --help'.
root@admin: /home/ash/Desktop/Flask ClassWorrk# docker run -p5000:5000 -t flaskapp:1.0
docker: Error response from daemon: driver failed programming external connectivity on endpoint silly_bardeen (717385e0278ed4890d3355b9d59728f1e3577afe801b18edb0257c07b377889d): Bind for 0.0.0.0:5000 failed: port is already allocated.
ERRO[0002] error waiting for container: context canceled
root@admin: /home/ash/Desktop/Flask ClassWorrk# docker run -t flaskapp:1.0
* Serving Flask app 'app'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://172.17.0.3:5000
Press CTRL+C to quit
172.17.0.1 - - [23/Oct/2022 17:21:29] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [23/Oct/2022 17:21:29] "GET /favicon.ico HTTP/1.1" 404 -
172.17.0.1 - - [23/Oct/2022 17:21:50] "GET / HTTP/1.1" 200 -



```
node 13-alpine3.10 9e3fc25f69d2 2 years ago 114MB
postgres 10.10 9a05a2b9e69f 3 years ago 211MB
root@admin:/home/ash/Desktop/Flask ClassWorrk# docker tag flaskapp:1.0 icr.io/helloind/python:1.0
root@admin:/home/ash/Desktop/Flask ClassWorrk# docker push icr.io/helloind/python:1.0
The push refers to repository [icr.io/helloind/python]
f7a508b1c151: Pushed
b7b353df37a8: Pushing [=====>] 12.38MB
7216ce2329ba: Pushed
0f61ccc2347a: Pushed
bfc1deb8136e: Pushed
1f123186824c: Pushed
3d6eb1152931: Pushing [=====>] 7.748MB/57.12MB
100796cdf3b1: Pushing [=====>] 11.14MB/18.48MB
54acb5a6fa0b: Pushing [>] 4.294MB/528.7MB
8d51c618126f: Pushing [>] 1.063MB/152MB
9ff6e4d46744: Waiting
a89d1d47b5a1: Waiting
655ed1b7a428: Waiting
^Croot@admin:/home/ash/Desktop/Flask ClassWorrk# docker push icr.io/helloind/python:1.0
The push refers to repository [icr.io/helloind/python]
f7a508b1c151: Layer already exists
b7b353df37a8: Layer already exists
7216ce2329ba: Layer already exists
0f61ccc2347a: Layer already exists
bfc1deb8136e: Layer already exists
1f123186824c: Layer already exists
3d6eb1152931: Pushed
100796cdf3b1: Pushed
54acb5a6fa0b: Pushed
8d51c618126f: Pushed
9ff6e4d46744: Pushed
a89d1d47b5a1: Pushed
655ed1b7a428: Pushed
1.0: digest: sha256:9d7b9f369391c18d6aa88270ebb467ab6b93b3f18e61a46c9467b7fa5cc4b404 size: 3050
root@admin:/home/ash/Desktop/Flask ClassWorrk#
```

Kubernetes

Dashboard-adminuser.yaml

apiVersion: v1

kind: ServiceAccount

metadata:

name: admin-user

namespace: kubernetes-dashboard

—

apiVersion: v1

kind: Secret

metadata:

name: admin-user-token

namespace: kubernetes-dashboard

annotations:

kubernetes.io/service-account.name: admin-user

type: kubernetes.io/service-account-token

flask_deployment.yaml

apiVersion: apps/v1

kind: Deployment

metadata:

name: flask-app

spec:

replicas: 3

selector:

matchLabels:

app: flask-app

template:

metadata:

labels:

app: flask-app

spec:

containers:

- name: flask-app-container

image: flask-app-testing

imagePullPolicy: Never

ports:

- containerPort: 5000

protocol: TCP

flask_ingress.yaml

apiVersion: networking.k8s.io/v1

kind: Ingress

metadata:

name: flask-app-ingress

annotations:

kubernetes.io/ingress.class: nginx

nginx.ingress.kubernetes.io/ssl-redirect: "false"

spec:

#ingressClassName: nginx

rules:

- http:

paths:

- backend:

service:

name: flask-app-service

```
    port:
      number: 5000
  path: /
  pathType: Prefix
```

flask_service.yaml

```
apiVersion: v1
kind: Service
metadata:
  name: flask-app-service
spec:
  type: ClusterIP
  ports:
    - port: 5000
  selector:
    app: flask-app
```

```
—
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: admin-user
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: cluster-admin
subjects:
```

-kind: ServiceAccount

name: admin-user

namespace: kubernetes-dashboard