

IBM Assignment - 4

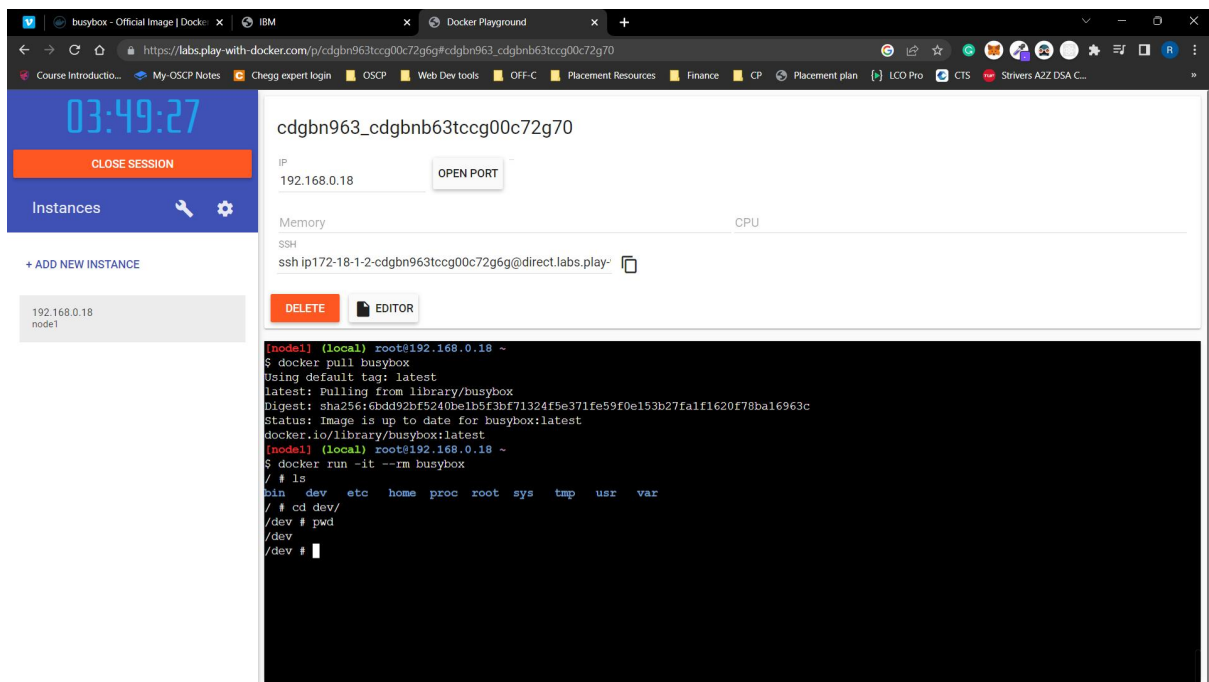
Project Title : **Personal Expense Tracker Application**

Team ID : **PNT2022TMID32657**

Name: **Nithin T**

Registration No : **813819104063**

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

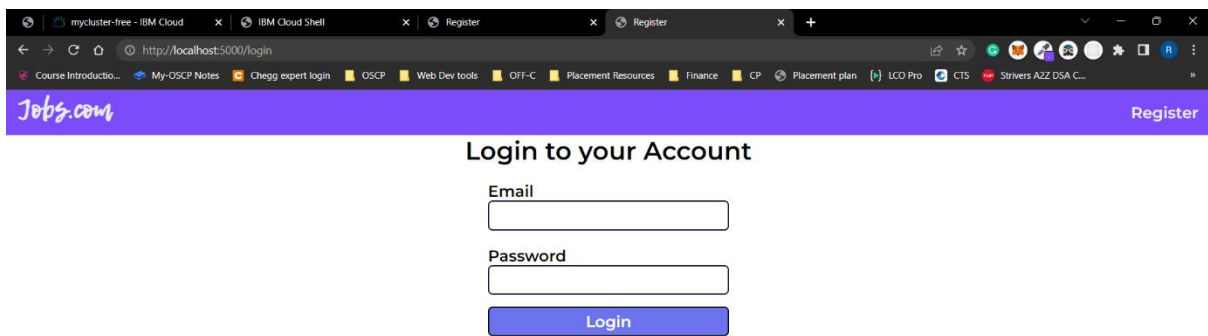
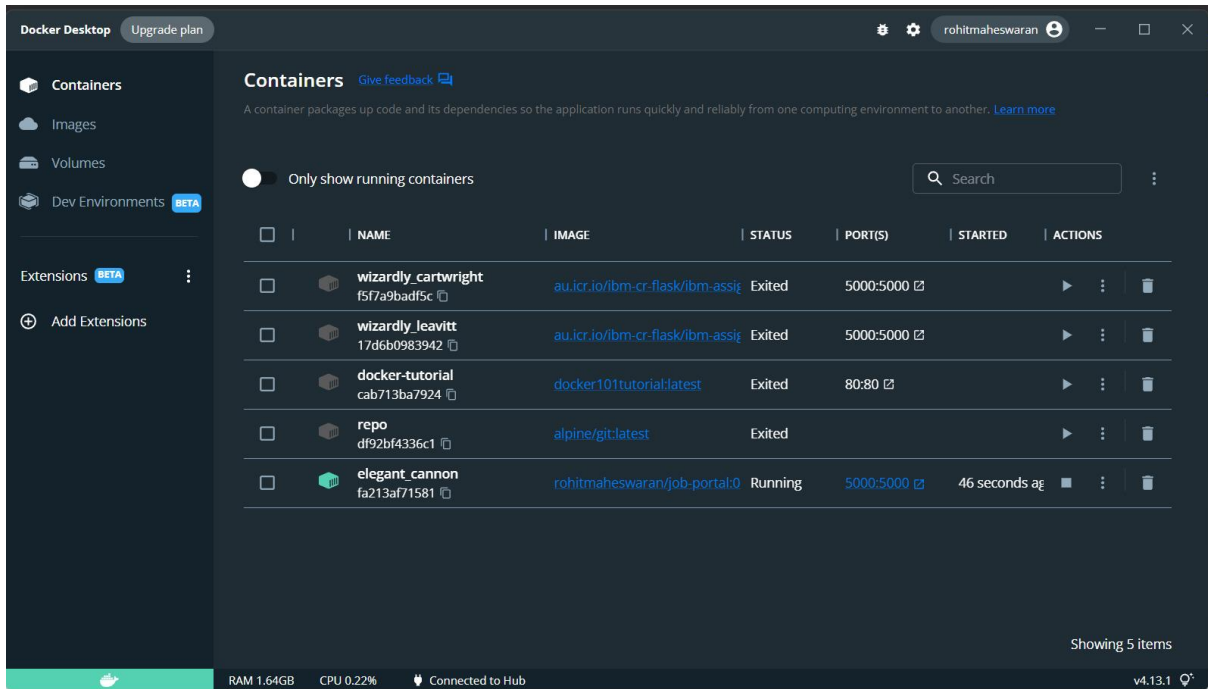


2) Create a docker file for the job portal application and deploy it in Docker desktop application.

Dockerfile

```
FROM python:3-alpine3.15
WORKDIR /app
COPY . /app
RUN pip install -r requirements.txt
EXPOSE 5000
CMD python ./app.py
```

```
PS D:\Flask\job> docker build -t rohitmaheswaran/job-portal:0.0.1.RELEASE .
[+] Building 1.5s (9/9) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 31B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/python:3-alpine3.15
=> [internal] load build context
=> => transferring context: 401B
=> [1/4] FROM docker.io/library/python:3-alpine3.15@sha256:d89ac9cefd2213b99a792fd8ec4f15c0297d9340a688b00fd8e136d0f03cc30e
=> CACHED [2/4] WORKDIR /app
=> CACHED [3/4] COPY . /app
=> CACHED [4/4] RUN pip install -r requirements.txt
=> exporting image
=> => exporting layers
=> => writing image sha256:8e89dbefaf6e0271b0e28ed824f5efdcfb8b629ce92bd63eac1fa9f33a91b68
=> => naming to docker.io/rohitmaheswaran/job-portal:0.0.1.RELEASE
Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
PS D:\Flask\job>
```



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

```
PS D:\flask\hello world> docker build -t rohitmaheswaran/hello-world-flask:0.0.1.RELEASE .
[+] Building 15.9s (10/10) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 157B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/python:alpine3.15
=> [auth] library/python:pull token for registry-1.docker.io
=> [1/4] FROM docker.io/library/python:alpine3.15@sha256:d89ac9cefd2213b99a792fd8ec4f15c0297d9340a688b00fd8e136d0f03cc30e
=> [internal] load build context
=> => transferring context: 461B
=> CACHED [2/4] WORKDIR /app
=> [3/4] COPY . /app
=> [4/4] RUN pip install -r requirements.txt
=> exporting image
=> => writing image sha256:6706dd5c23a0ef46b7b9a4d799a9dc444932ac9e7628269e84fdce396bc0ee66
=> => naming to docker.io/rohitmaheswaran/hello-world-flask:0.0.1.RELEASE

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
PS D:\flask\hello world> docker run -d -p 5000:5000 rohitmaheswaran/hello-world-flask:0.0.1.RELEASE
17d6b0983942ea00590105ac76fb15f310d339250e26cd2675a67e346da62c53
```

```
PS D:\flask\hello world> docker push au.icr.io/ibm-cr-flask/ibm-assign:hello-world-flask
The push refers to repository [au.icr.io/ibm-cr-flask/ibm-assign]
ad2bd141721b: Pushed
1d22bed8e9b8: Pushed
73c877a9d8cc: Pushed
f9a01ea63d59: Pushed
eb71c8b7b3b7: Pushed
76d682e14461: Pushed
d59c8eb8f9a4: Pushed
34d5ebaa5410: Pushed
hello-world-flask: digest: sha256:82a32086d2347c8893cf6f06f5a88406ed34277a91d50c4300e41c170037e0d7 size: 1993
PS D:\flask\hello world> ibmcloud cr image-list
listing images...

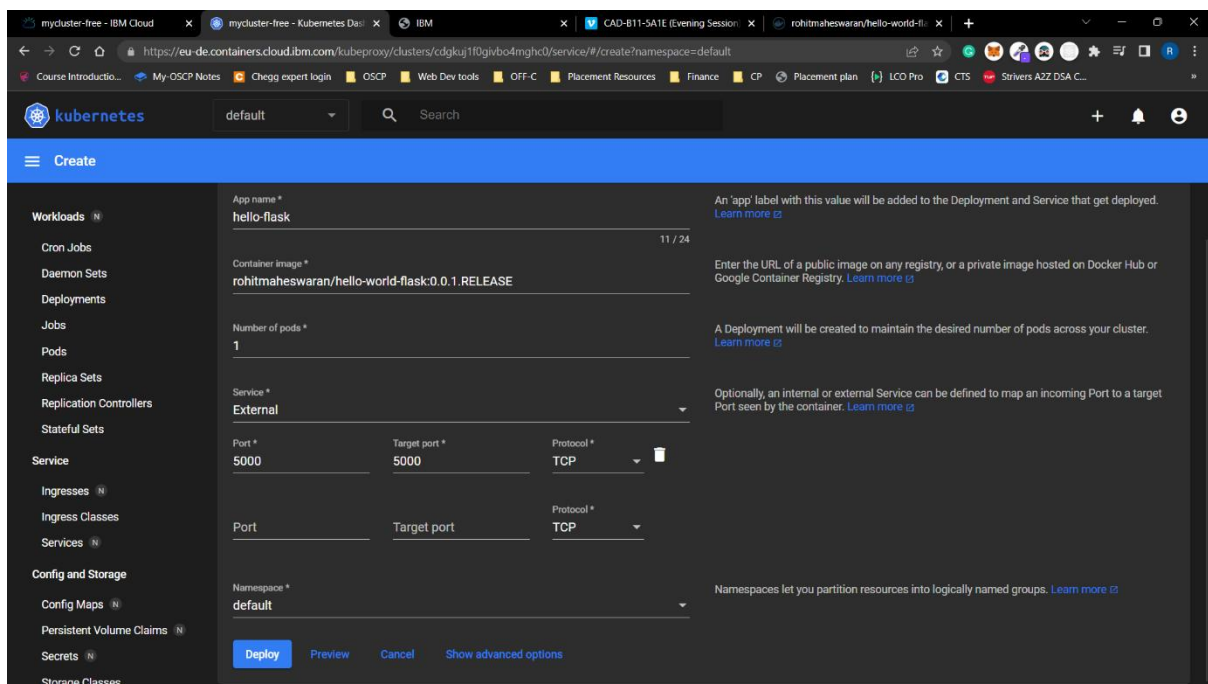
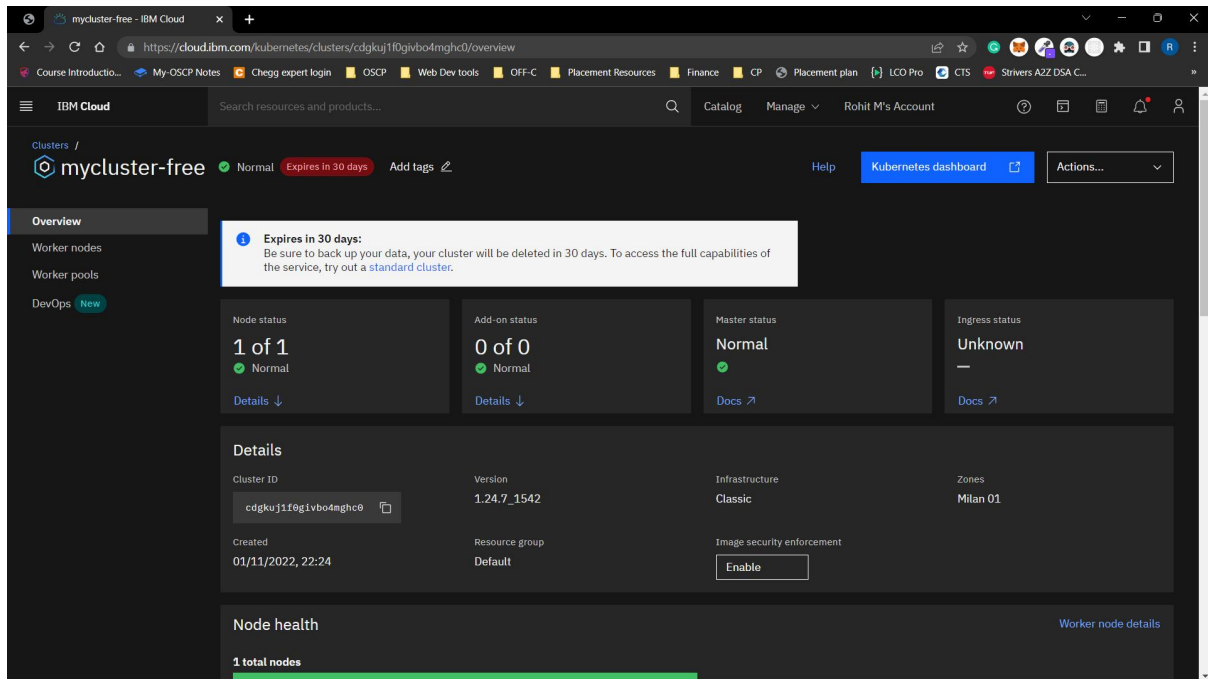
Repository          Tag          Digest          Namespace    Created      Size    Security status
au.icr.io/ibm-cr-flask/ibm-assign  hello-world-flask  82a32086d234  ibm-cr-flask  11 minutes ago  27 MB  -

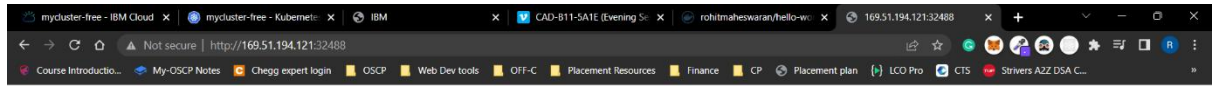
OK
PS D:\flask\hello world>
```

The screenshot shows the IBM Cloud Container Registry console. The left sidebar contains navigation links: Container Registry, Quick start, Namespaces, Repositories, Images, Trash, and Settings. The main content area is titled 'Namespaces' and shows a list of namespaces. The 'au.icr.io/ibm-cr-flask/ibm-assign' namespace is selected, and its details are shown in a table.

Name	Resource group	Repository count	Image count	Retention policy
ibm-cr-flask	Default	1	1	Retain all images
au.icr.io/ibm-cr-flask/ibm-assign		1	1	1 day ago

- 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 node port.





Welcome to Flask App!! :)
