IBM Assignment - 4

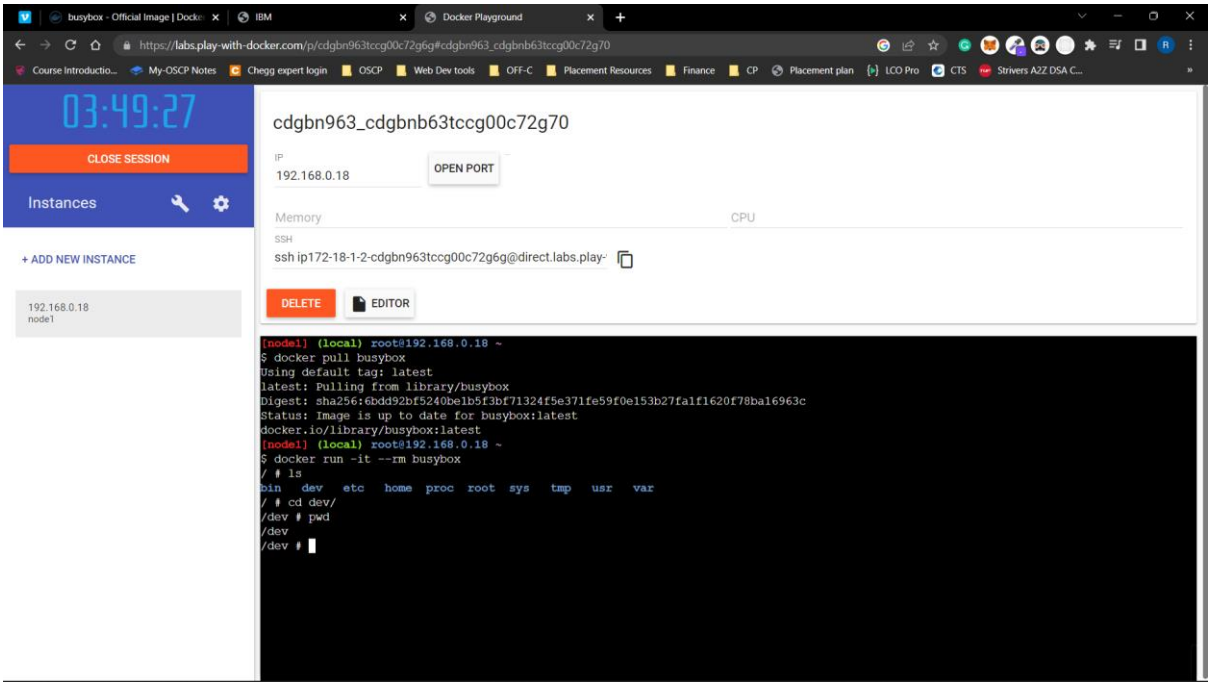
Project Title : **Personal Expense Tracker Application**

Team ID : **PNT2022TMID32657**

Name: **Rohit M**

Registration No : **813819104077**

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

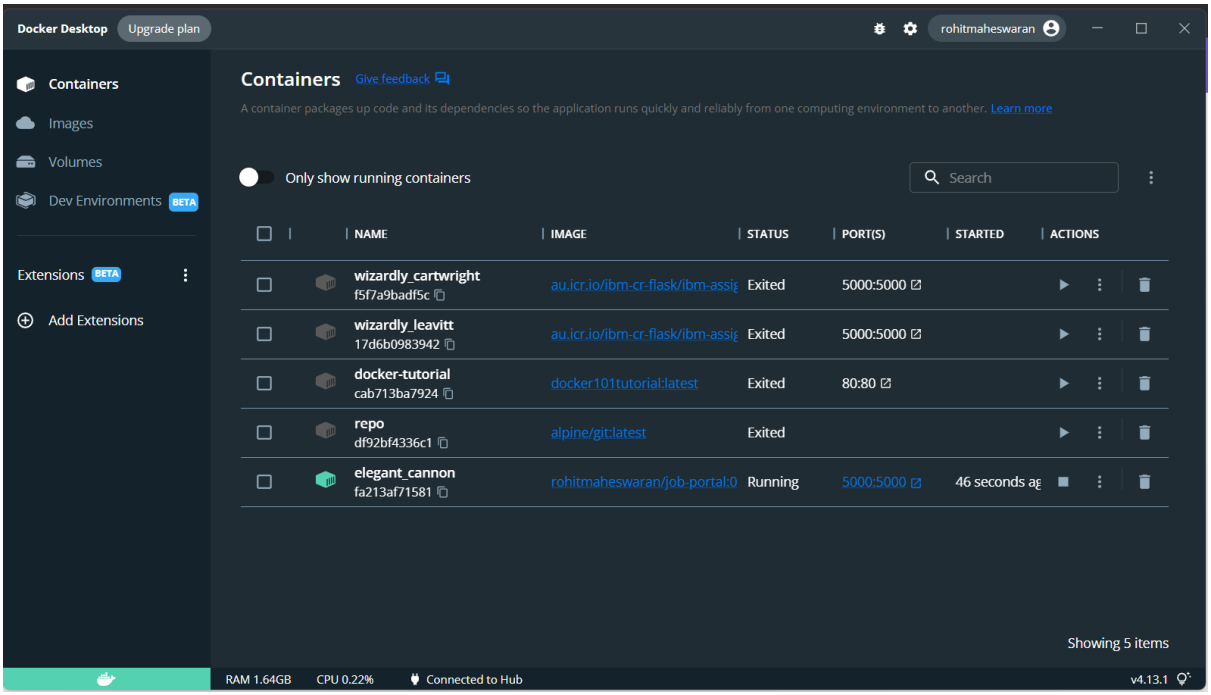


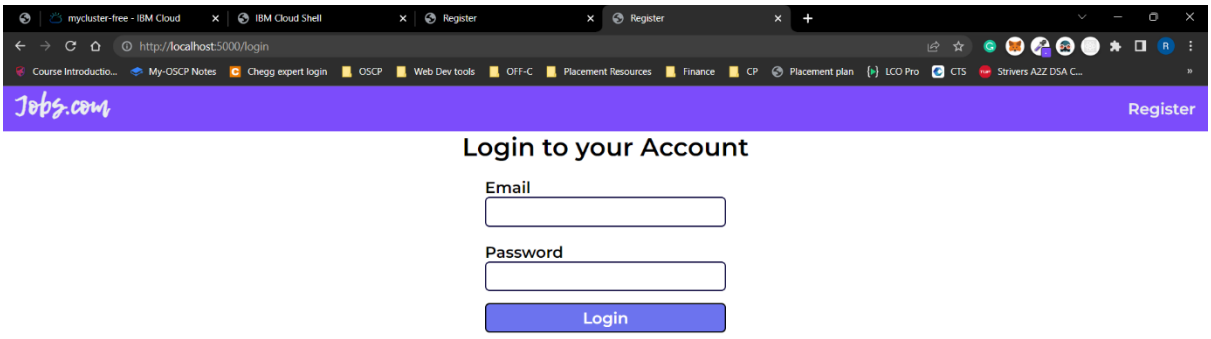
2) Create a docker file for the job portal application and deploy it in Docker desktop application.

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                                                    0.0s
 => => transferring dockerfile: 31B                                                                     0.0s
 => [internal] load .dockerignore                                                                       0.0s
 => => transferring context: 2B                                                                         0.0s
 => [internal] load metadata for docker.io/library/python:3-alpine3.15                                  1.4s
 => [internal] load build context                                                                       0.0s
 => => transferring context: 401B                                                                       0.0s
 => [1/4] FROM docker.io/library/python:3-alpine3.15@sha256:d89ac9cefd2213b99a792fd8ec4f15c0297d9340a688b00fdae136d0f03cc30e  0.0s
 => CACHED [2/4] WORKDIR /app                                                                           0.0s
 => CACHED [3/4] COPY . /app                                                                            0.0s
 => CACHED [4/4] RUN pip install -r requirements.txt                                                    0.0s
 => exporting to image                                                                                  0.0s
 => => exporting layers                                                                                 0.0s
 => => writing image sha256:8e89dbebaf6e0271b0e28ed824f5efdcfb8b629cbe92bd63eac1fa9f33a91b68            0.0s
 => => naming to docker.io/rohitmaheswaran/job-portal:0.0.1.RELEASE                                     0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
PS D:\flask\job>
```

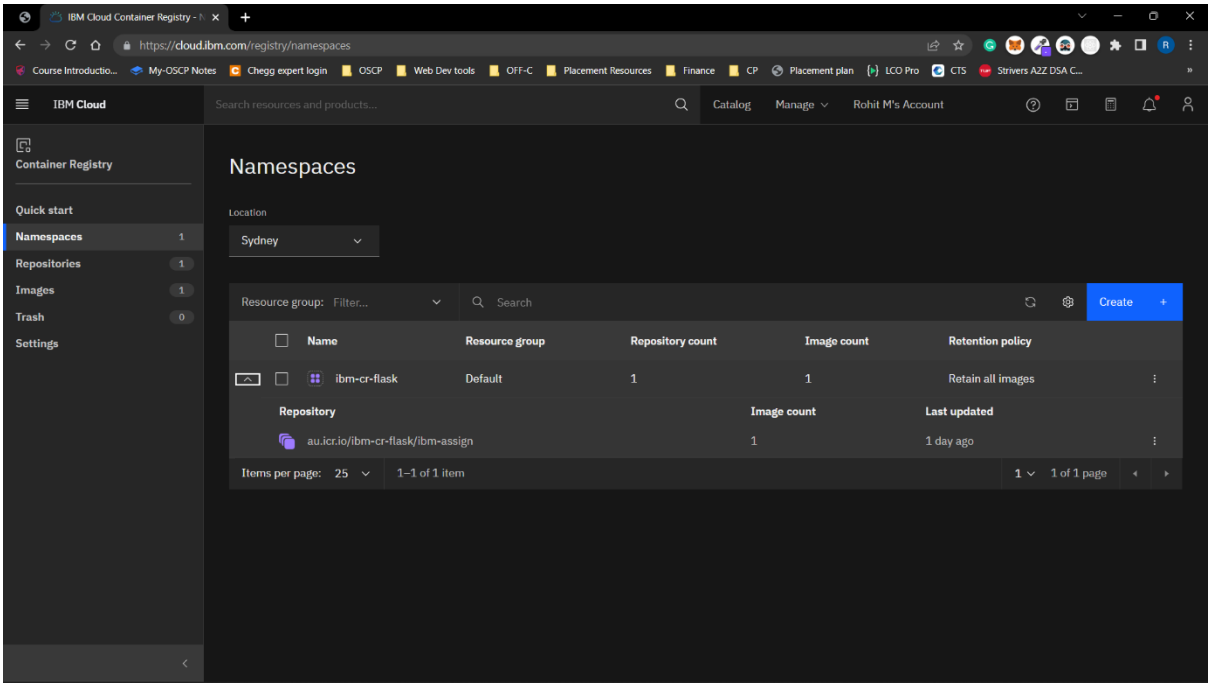




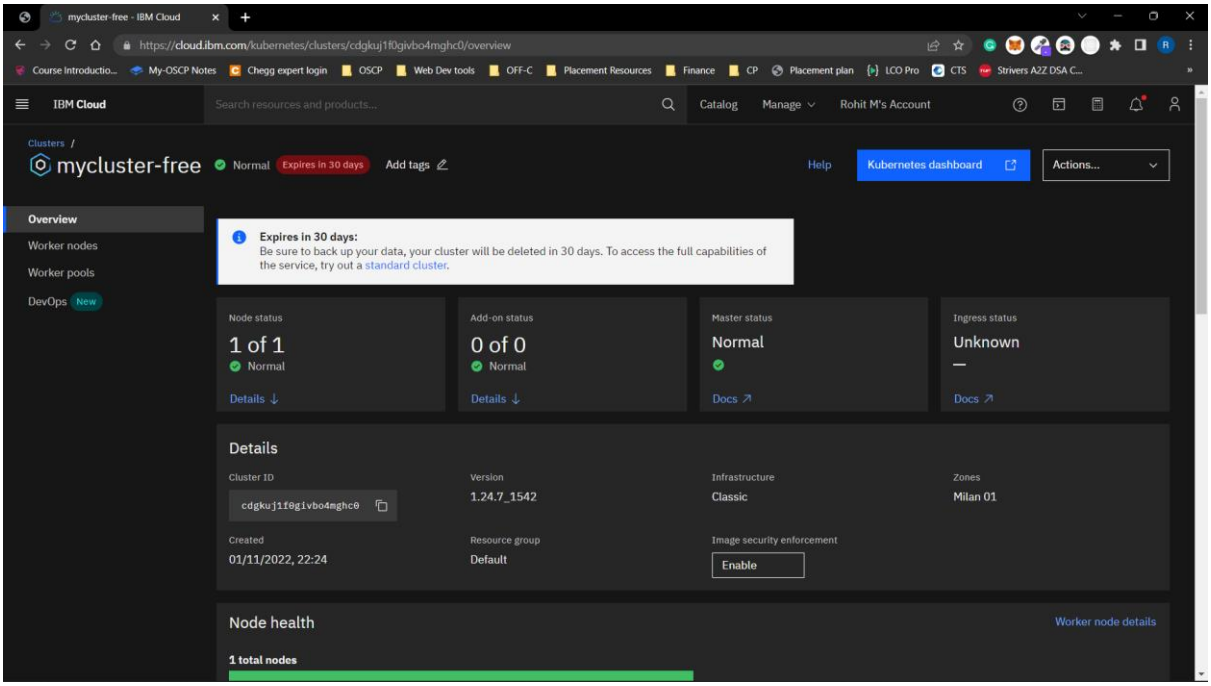
3) Create a IBM container registry and deploy hello world app or job portal app.

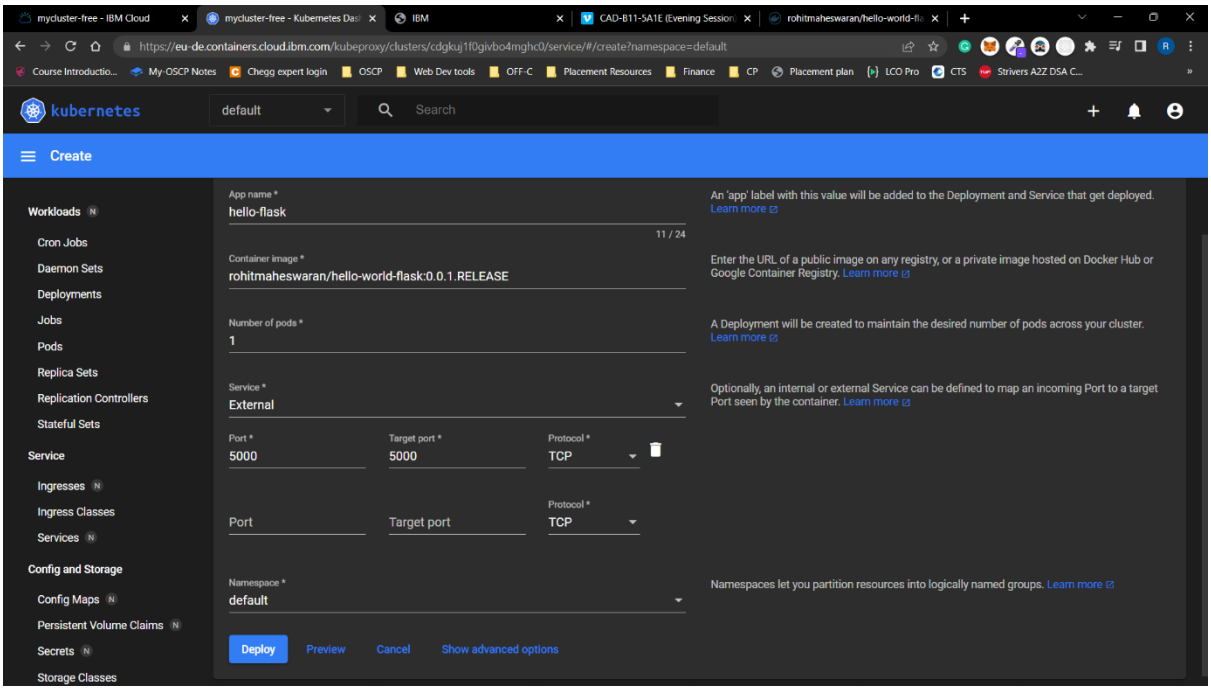
```
PS D:\flask\hello world> docker build -t rohitmaheswaran/hello-world-flask:0.0.1.RELEASE .
[+] Building 15.9s (10/10) FINISHED
 => [internal] load build definition from Dockerfile                                                0.1s
 => => transferring dockerfile: 157B                                                                0.1s
 => [internal] load .dockerignore                                                                   0.1s
 => => transferring context: 2B                                                                     0.1s
 => [internal] load metadata for docker.io/library/python:alpine3.15                                3.1s
 => [auth] library/python:pull token for registry-1.docker.io                                       0.0s
 => [1/4] FROM docker.io/library/python:alpine3.15@sha256:d89ac9cefd2213b99a792fd8ec4f15c0297d9340a688b00fdae136d0f03cc30e  0.0s
 => [internal] load build context                                                                   0.0s
 => => transferring context: 461B                                                                   0.0s
 => CACHED [2/4] WORKDIR /app                                                                       0.0s
 => [3/4] COPY . /app                                                                               0.0s
 => [4/4] RUN pip install -r requirements.txt                                                      12.1s
 => exporting to image                                                                              0.4s
 => => exporting layers                                                                             0.4s
 => => writing image sha256:6706dd5c23a0ef46b7b9a4d799a9dc444932ac9e7628269e84fdce396bc0ee66        0.0s
 => => naming to docker.io/rohitmaheswaran/hello-world-flask:0.0.1.RELEASE                           0.0s

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
```

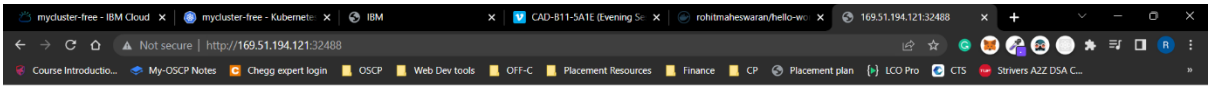
```
PS D:\flask\hello world> docker push au.icr.io/ibm-cr-flask/ibm-assign:hello-world-flask
The push refers to repository [au.icr.io/ibm-cr-flask/ibm-assign]
ad2bd141721b: Pushed
1d22bed8e9b8: Pushed
73c877a9d8cc: Pushed
f9a01ea63d59: Pushed
eb71c8b7b3b7: Pushed
76d682e14461: Pushed
d59c8eb8f9a4: Pushed
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>
```



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!! :)