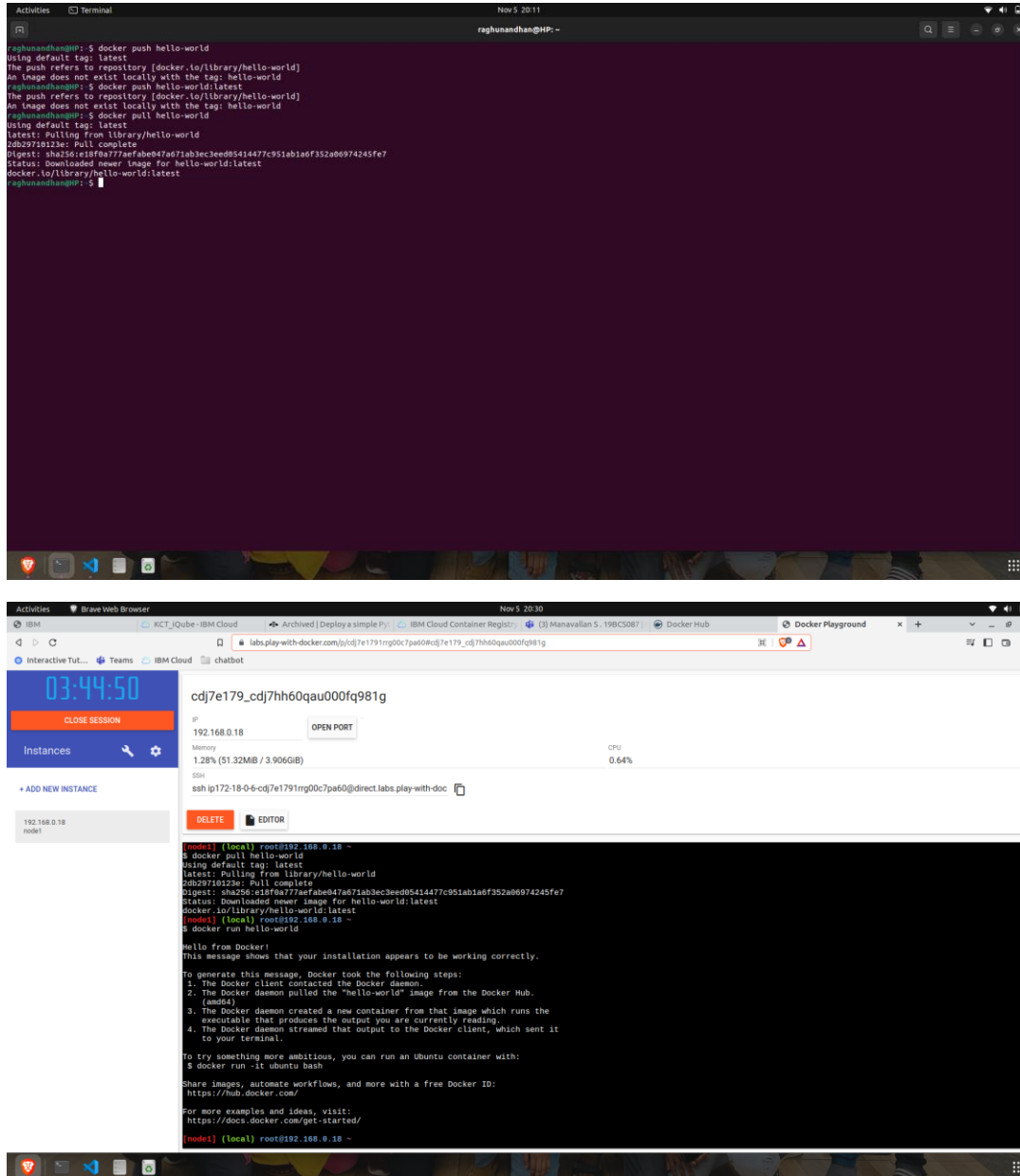


## Assignment 4

Joodith Hermminal

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



The image consists of two screenshots. The top screenshot is a terminal window with a dark background. It shows the following commands and output:

```
raghunandhan@HP: ~$ docker push hello-world
Using default tag: latest
The push refers to repository [docker.io/library/hello-world]
An image does not exist locally with the tag: hello-world
raghunandhan@HP: ~$ docker push hello-world:latest
The push refers to repository [docker.io/library/hello-world]
An image does not exist locally with the tag: hello-world
raghunandhan@HP: ~$ docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:c18f0a777efabe047a071ab3ec3eed05414477c951ab1a6f352a06974245fe7
Status: Downloaded newer image for hello-world:latest
docker.io/library/hello-world:latest
raghunandhan@HP: ~$
```

The bottom screenshot is a web browser window showing the Docker Playground interface. The URL is [https://labs.play-with-docker.com/p/cdj7e1791mg00c7pa60rcdj7e179\\_cdj7hh60qau000fq981g](https://labs.play-with-docker.com/p/cdj7e1791mg00c7pa60rcdj7e179_cdj7hh60qau000fq981g). The interface shows a container named "cdj7e179\_cdj7hh60qau000fq981g" with IP 192.168.0.18, Memory 1.28% (51.32MiB / 3.90GiB), and CPU 0.64%. Below the container details, there is a terminal window showing the following commands and output:

```
[nodes] (local) root@192.168.0.18 ~$ docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:c18f0a777efabe047a071ab3ec3eed05414477c951ab1a6f352a06974245fe7
Status: Downloaded newer image for hello-world:latest
docker.io/library/hello-world:latest
[nodes] (local) root@192.168.0.18 ~$ docker run hello-world

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

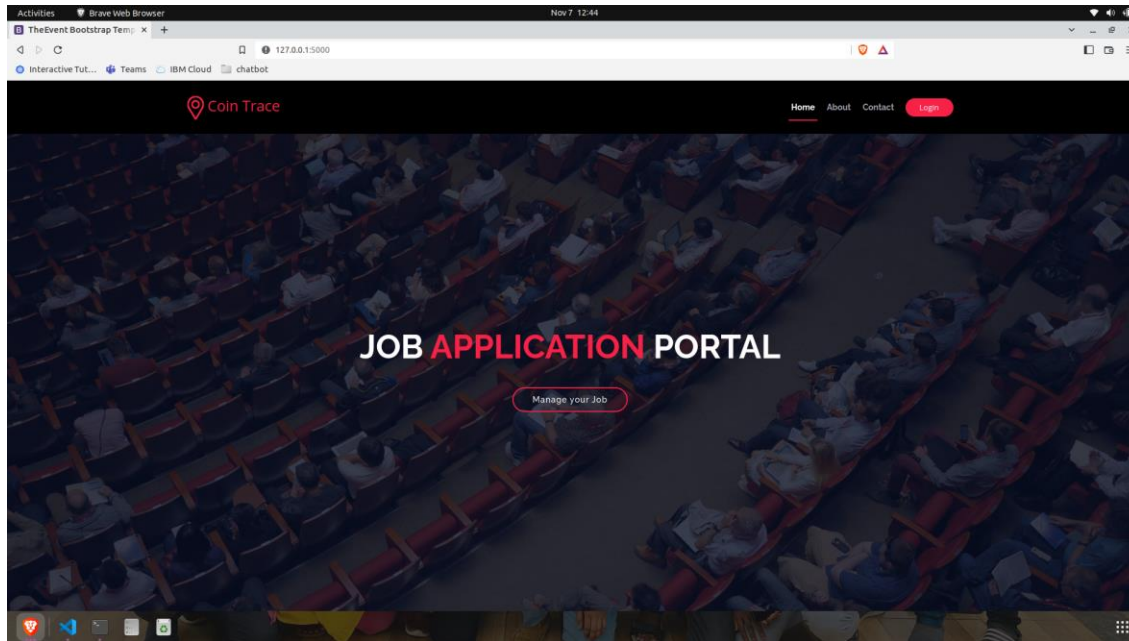
To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

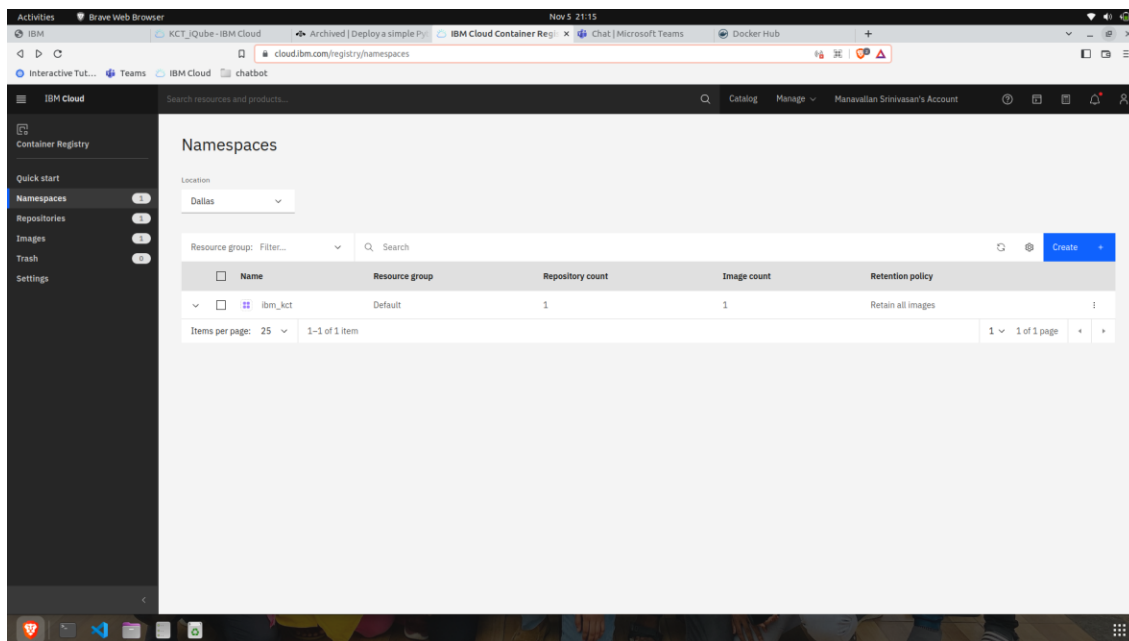
Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

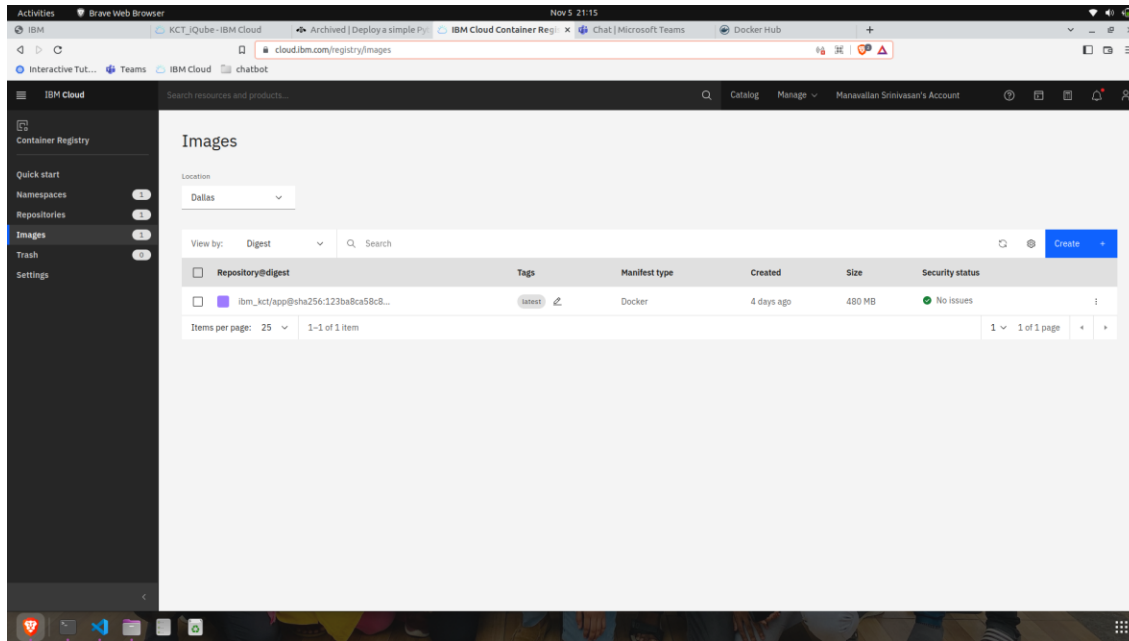
For more examples and ideas, visit:
https://docs.docker.com/get-started/
[nodes] (local) root@192.168.0.18 ~$
```

2. Create a docker file for the jobportal application and deploy it in Docker desktop application.

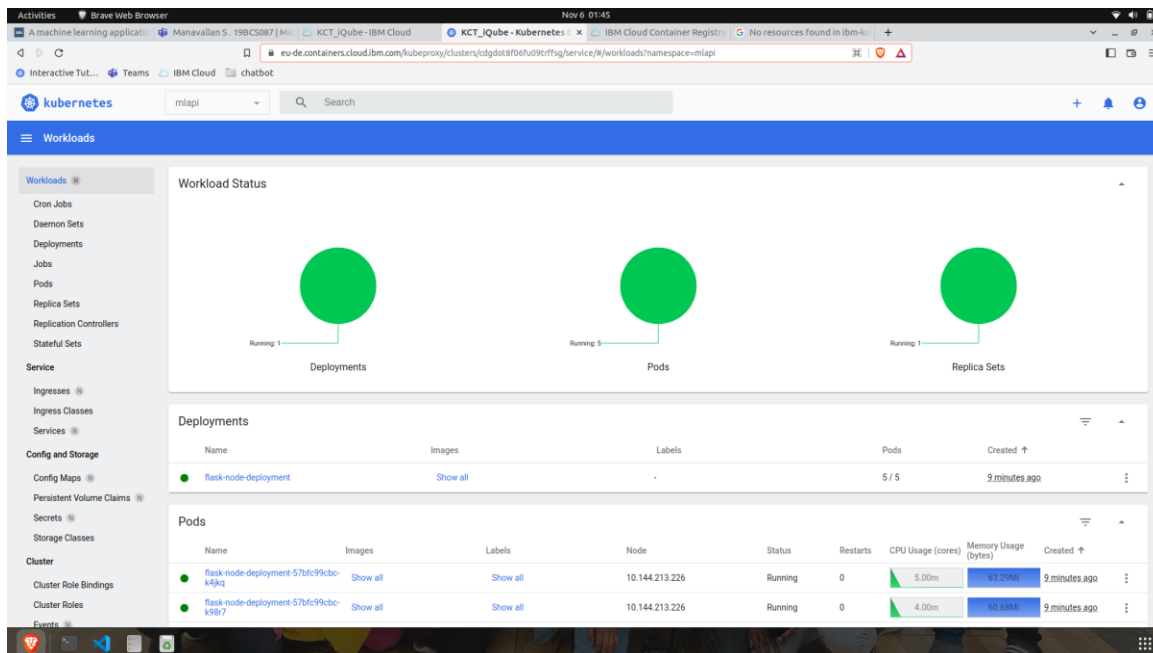


3. Create a IBM container registry and deploy helloworld app or jobportalapp.





4. Create a Kubernetes cluster in IBM cloud and deploy helloworld image or jobportal image and also expose the same app to run in nodeport.



Accessing the portal using the public IP and the port: 169.51.203.25: 31837

# JOB APPLICATION PORTAL

Manage your Job