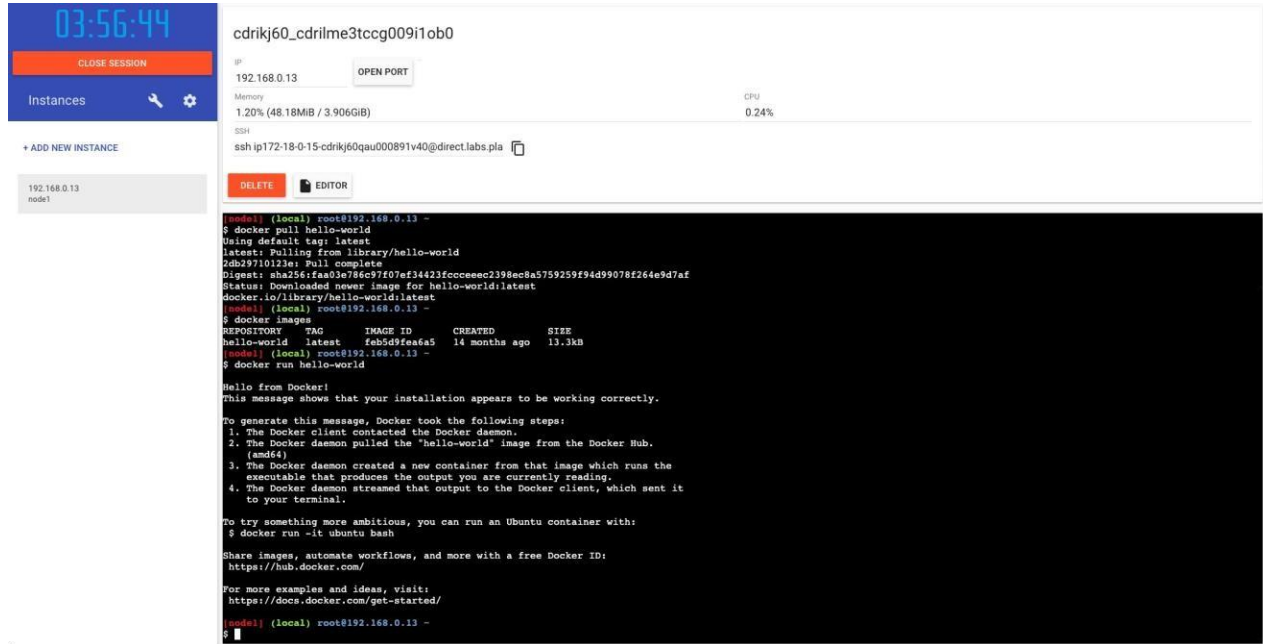


Pull an Image from docker hub and run it in docker playground.



The screenshot shows the Docker Playground interface. On the left, there's a sidebar with a clock showing 03:56:44, a 'CLOSE SESSION' button, and an 'Instances' section with a '+ ADD NEW INSTANCE' button. Below that, a list of instances shows '192.168.0.13 node1'. The main area displays the details of an instance named 'cdrikj60_cdrilme3tccg009i1ob0' with IP '192.168.0.13', memory usage '1.20% (48.18MiB / 3.906GiB)', and CPU usage '0.24%'. An 'SSH' button is present. Below the instance details, there's a terminal window showing the following commands and output:

```
[root@192.168.0.13 ~]# docker pull hello-world
latest: Pulling from library/hello-world
2db2971013e: pull complete
Digest: sha256:fa03e786c97f07ef34423f0c0e0ec2398ec8a5759259f94d99078f264e9d7af
Status: Downloaded newer image for hello-world:latest
docker.io/library/hello-world:latest
[root@192.168.0.13 ~]# docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
hello-world latest feb5d9fea6a5 14 months ago 13.3kb
[root@192.168.0.13 ~]# docker run hello-world

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

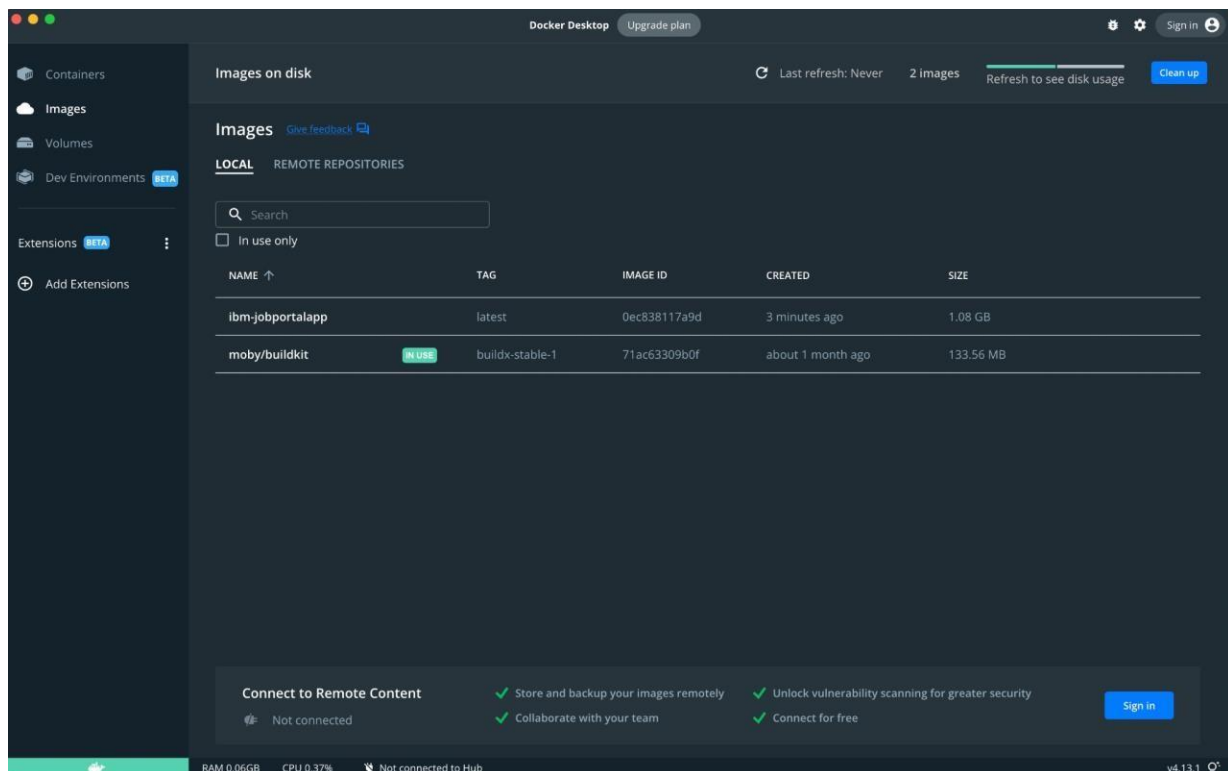
To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the 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/
[root@192.168.0.13 ~]#
```

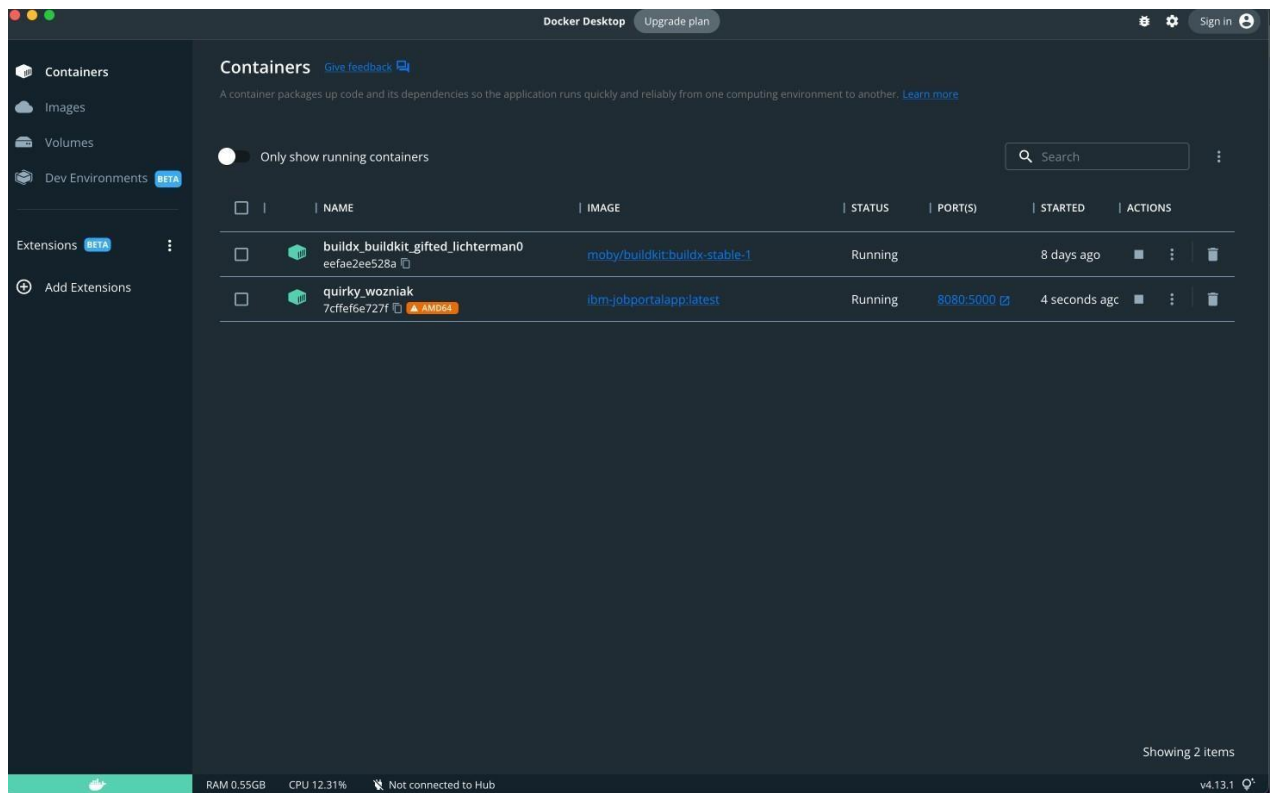
1. Create a docker file for the jobportal application and deploy it in the Docker desktop application.



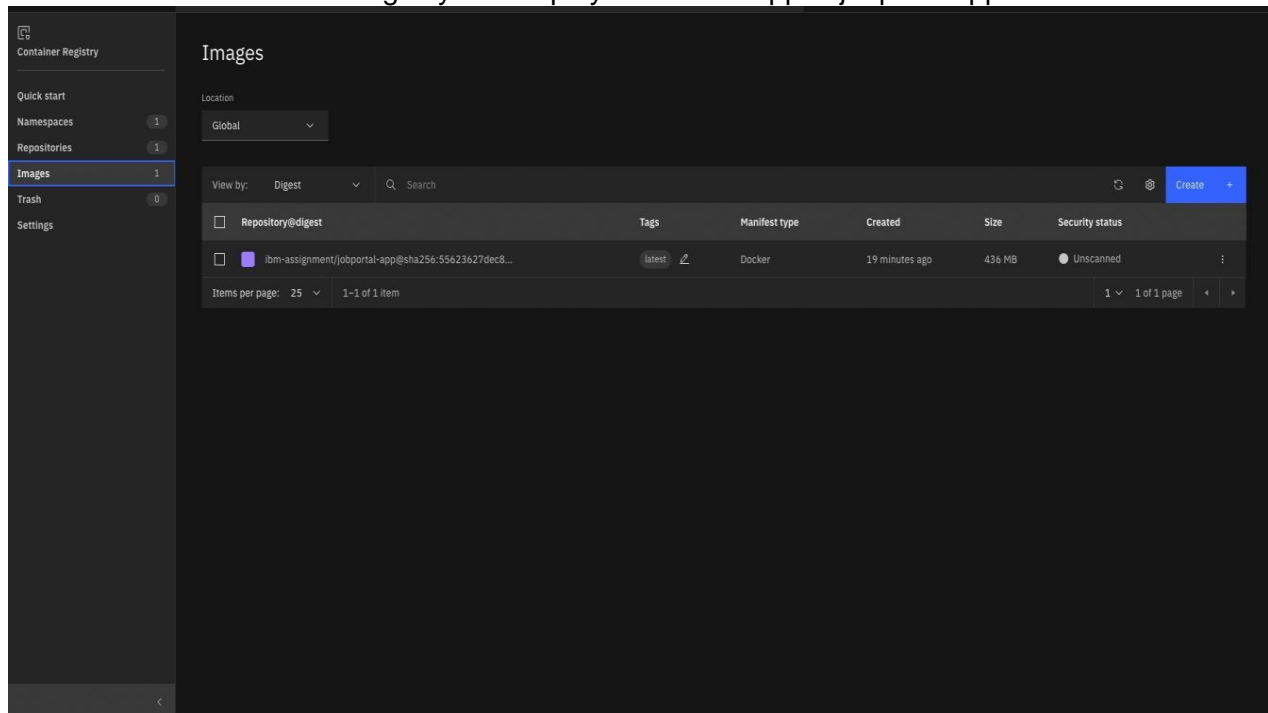
The screenshot shows the Docker Desktop interface. The top bar includes 'Docker Desktop', an 'Upgrade plan' button, and a 'Sign in' button. The left sidebar contains navigation options: 'Containers', 'Images', 'Volumes', 'Dev Environments', 'Extensions', and 'Add Extensions'. The main area is titled 'Images on disk' and shows a list of images. The 'LOCAL' tab is selected, displaying a table of images:

NAME	TAG	IMAGE ID	CREATED	SIZE
ibm-jobportalapp	latest	0ec838117a9d	3 minutes ago	1.08 GB
moby/buildkit	buildx-stable-1	71ac63309b0f	about 1 month ago	133.56 MB

At the bottom, there's a 'Connect to Remote Content' section with a 'Sign in' button and a status bar at the very bottom showing 'RAM 0.06GB', 'CPU 0.37%', and 'Not connected to Hub'.



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



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

