Assignment 4 - B11-5A1E

Karthickeyan E

2019103025

BE CSE, Batch P, College of Engineering, Guindy, Anna University

Assignment Kubernetes / Docker

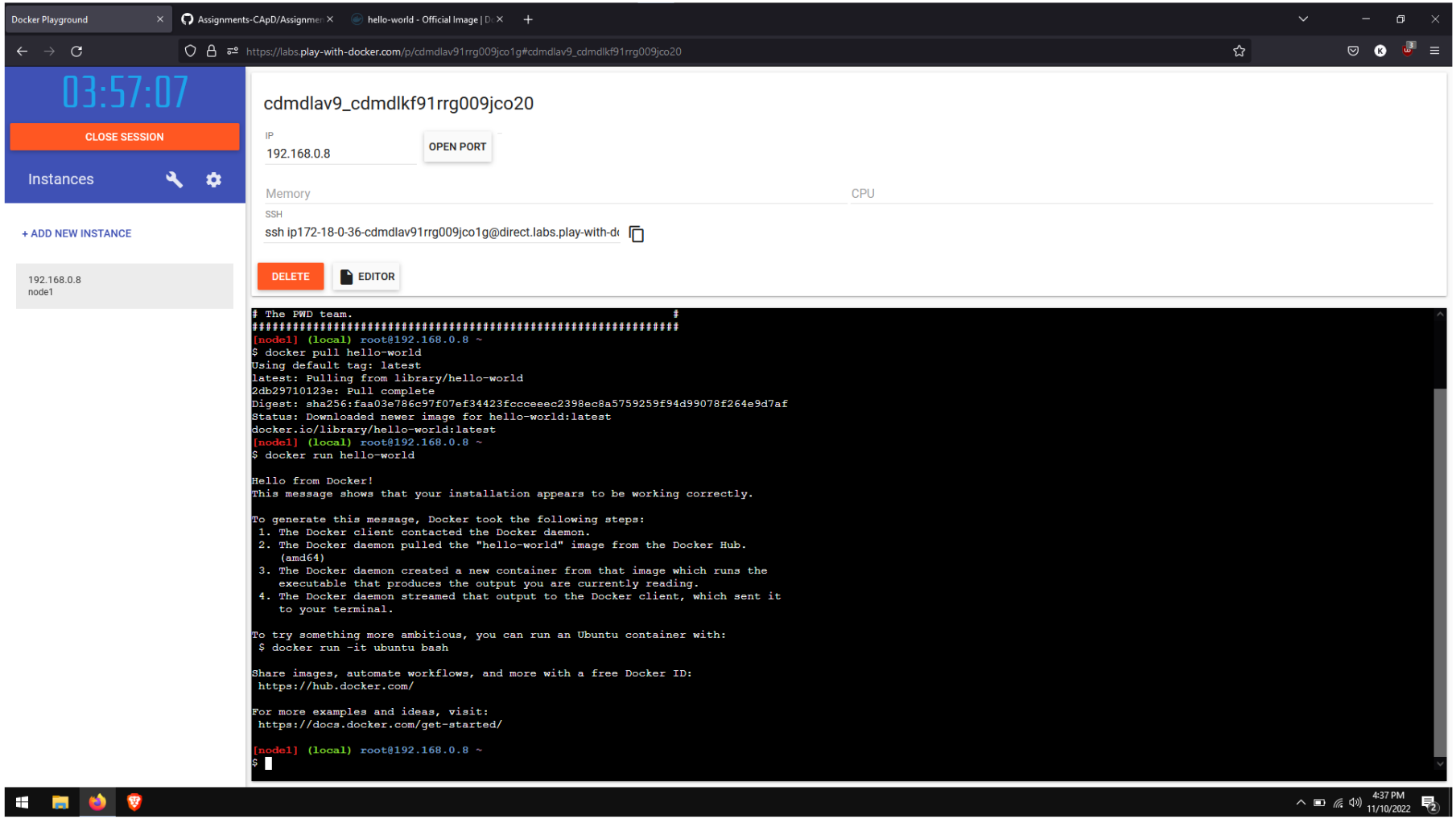
1.Pull an Image from docker hub and run it in docker playground.

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

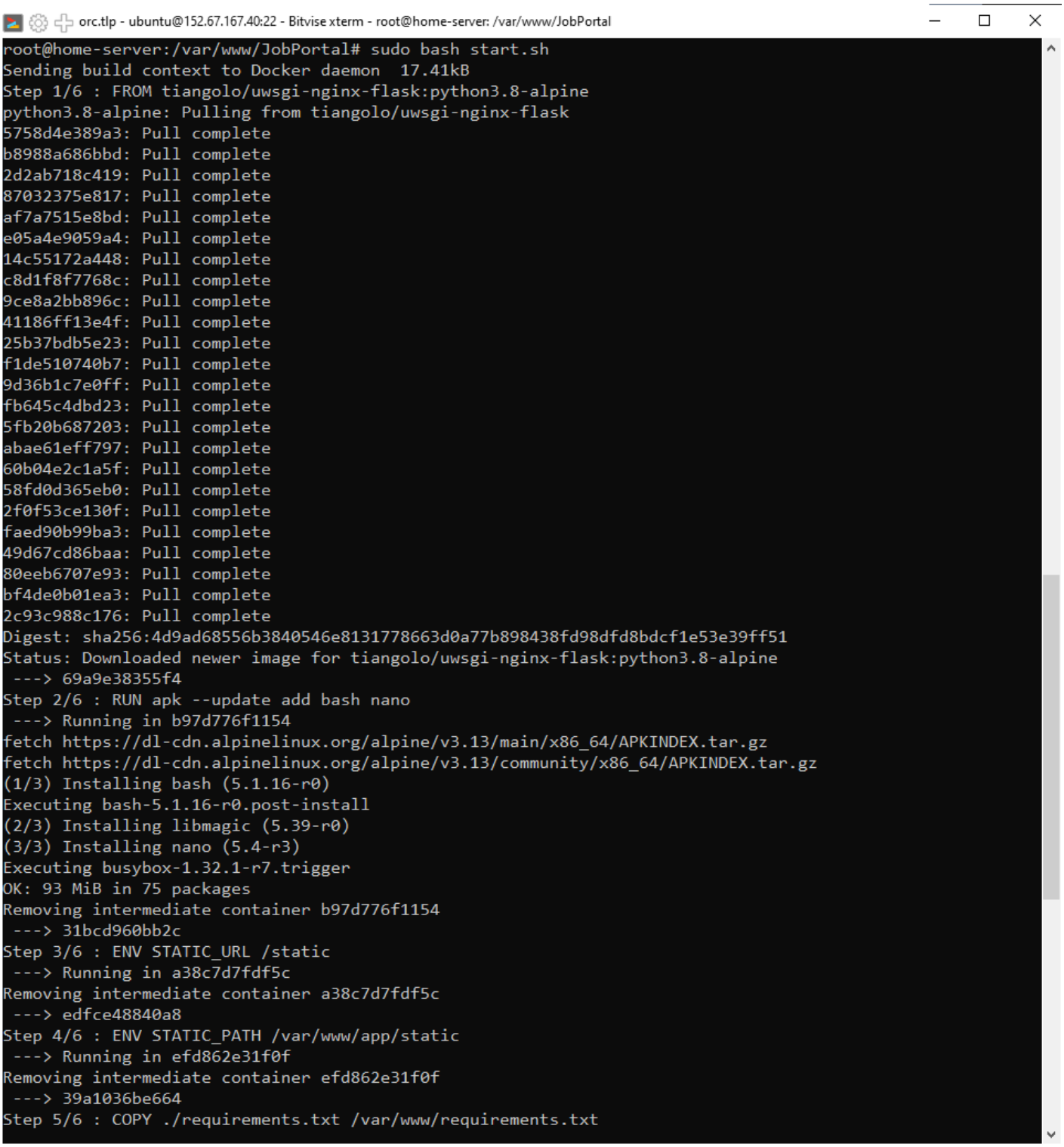
3.Create an IBM container registry and deploy helloworld app or jobportalapp.

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.

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.



orc.tlp - ubuntu@152.67.167.40:22 - Bitvise xterm - root@home-server: /var/www/JobPortal

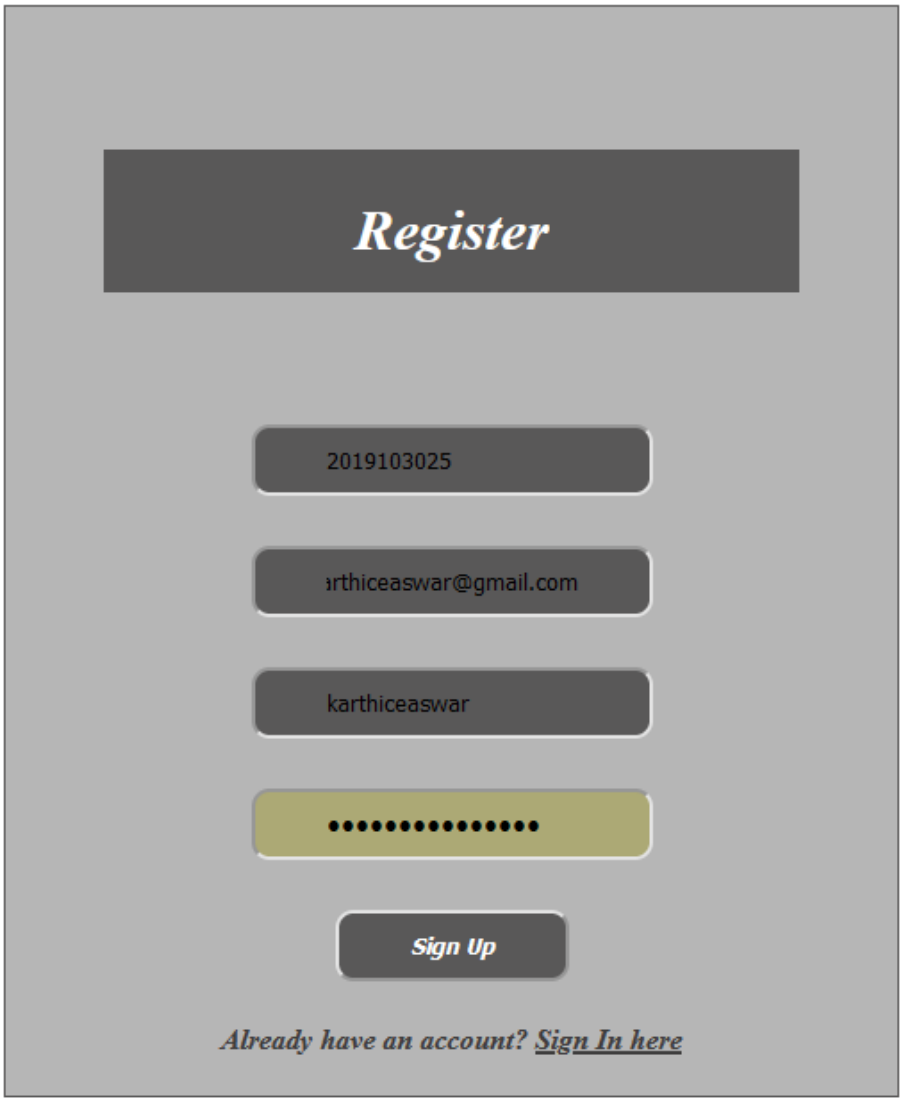
```
Executing bash-5.1.16-r0.post-install
(2/3) Installing libmagic (5.39-r0)
(3/3) Installing nano (5.4-r3)
Executing busybox-1.32.1-r7.trigger
OK: 93 MiB in 75 packages
Removing intermediate container b97d776f1154
 ---> 31bcd960bb2c
Step 3/6 : ENV STATIC_URL /static
 ---> Running in a38c7d7fdf5c
Removing intermediate container a38c7d7fdf5c
 ---> edfce48840a8
Step 4/6 : ENV STATIC_PATH /var/www/app/static
 ---> Running in efd862e31f0f
Removing intermediate container efd862e31f0f
 ---> 39a1036be664
Step 5/6 : COPY ./requirements.txt /var/www/requirements.txt
 ---> 8fd561a2e33a
Step 6/6 : RUN pip install -r /var/www/requirements.txt
 ---> Running in 1c59a5dd61a0
Collecting Flask>=2.0.2
  Downloading Flask-2.2.2-py3-none-any.whl (101 kB)
Requirement already satisfied: click>=8.0 in /usr/local/lib/python3.8/site-packages (from Flask>=2.0.2->
-r /var/www/requirements.txt (line 1)) (8.1.3)
Requirement already satisfied: itsdangerous>=2.0 in /usr/local/lib/python3.8/site-packages (from Flask>=
2.0.2->-r /var/www/requirements.txt (line 1)) (2.1.2)
Collecting importlib-metadata>=3.6.0
  Downloading importlib_metadata-5.0.0-py3-none-any.whl (21 kB)
Requirement already satisfied: Werkzeug>=2.2.2 in /usr/local/lib/python3.8/site-packages (from Flask>=2.
0.2->-r /var/www/requirements.txt (line 1)) (2.2.2)
Requirement already satisfied: Jinja2>=3.0 in /usr/local/lib/python3.8/site-packages (from Flask>=2.0.2-
>-r /var/www/requirements.txt (line 1)) (3.1.2)
Collecting zipp>=0.5
  Downloading zipp-3.10.0-py3-none-any.whl (6.2 kB)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.8/site-packages (from Jinja2>=3
.0->Flask>=2.0.2->-r /var/www/requirements.txt (line 1)) (2.1.1)
Installing collected packages: zipp, importlib-metadata, Flask
  Attempting uninstall: Flask
    Found existing installation: Flask 2.0.1
    Uninstalling Flask-2.0.1:
      Successfully uninstalled Flask-2.0.1
Successfully installed Flask-2.2.2 importlib-metadata-5.0.0 zipp-3.10.0
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with
the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/
warnings/venv
WARNING: You are using pip version 21.2.4; however, version 22.3.1 is available.
You should consider upgrading via the '/usr/local/bin/python -m pip install --upgrade pip' command.
Removing intermediate container 1c59a5dd61a0
 ---> 65c03b84678f
Successfully built 65c03b84678f
Successfully tagged jobportal.test:latest
d3326c9769587cae364ce0446b496b4673064776461763193f1e961fa4068a43
root@home-server:/var/www/JobPortal#
```

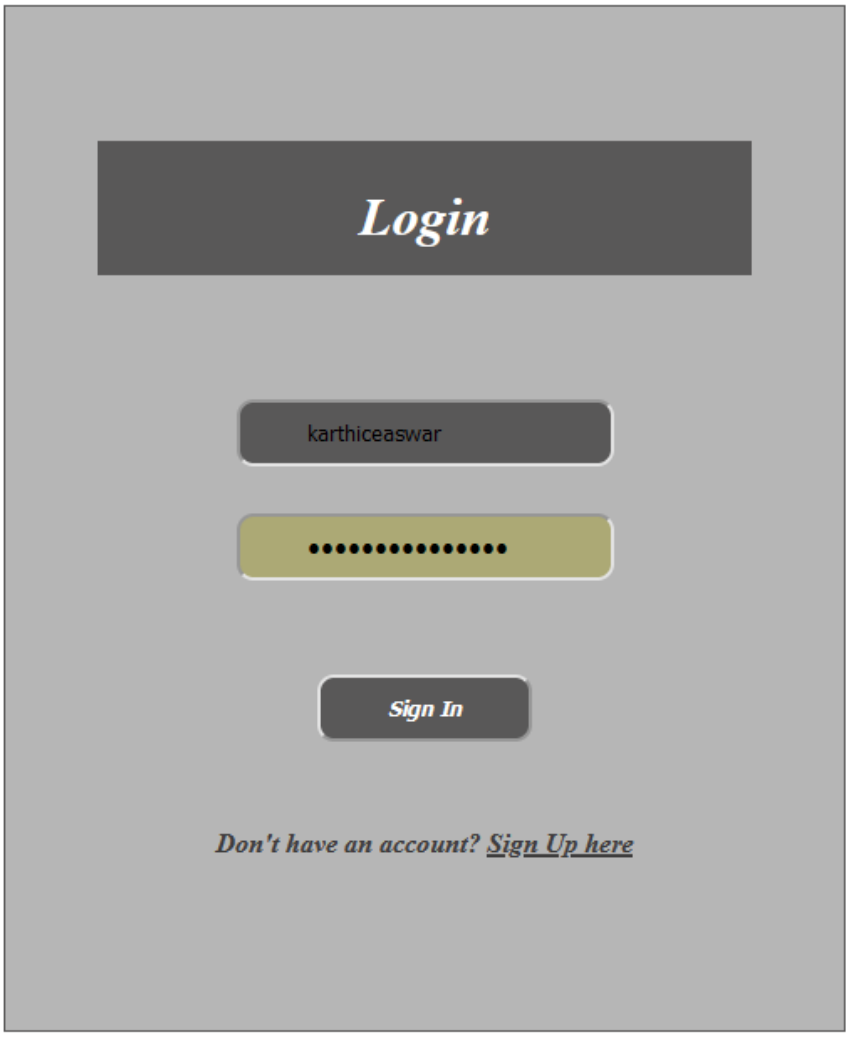
orc.tlp - ubuntu@152.67.167.40:22 - Bitvise xterm - root@home-server: /var/www/JobPortal

```
root@home-server:/var/www/JobPortal# docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED         STATUS
 PORTS                                                                                  NAMES
d3326c976958   jobportal.test                       "/entrypoint.sh /sta…"   3 minutes ago   Up 3 minutes
 443/tcp, 0.0.0.0:56733->80/tcp, :::56733->80/tcp                                       jobportal.test
9f32f2ce15c8   ghcr.io/linuxserver/jellyfin:latest   "/init"                  4 days ago      Up 4 days
 0.0.0.0:8096->8096/tcp, :::8096->8096/tcp, 8920/tcp                                    jellyfin
302a3a4105f2   jc21/nginx-proxy-manager:latest      "/init"                  4 days ago      Up 4 days
 0.0.0.0:80-81->80-81/tcp, :::80-81->80-81/tcp, 0.0.0.0:443->443/tcp, :::443->443/tcp   npm
root@home-server:/var/www/JobPortal#
```

orc.tlp - ubuntu@152.67.167.40:22 - Bitvise xterm - root@home-server: /var/www/JobPortal

```
root@home-server:/var/www/JobPortal# ls -R
.:
Dockerfile  JobPortal.zip  __pycache__  app  main.py  requirements.txt  start.sh  uwsgi.ini

./__pycache__:
main.cpython-38.pyc

./app:
__init__.py  __pycache__  static  templates  views.py

./app/__pycache__:
__init__.cpython-38.pyc  views.cpython-38.pyc

./app/static:
style.css

./app/templates:
index.html  login.html  register.html
root@home-server:/var/www/JobPortal#
```




Login
karthiceaswar
Sign In
Don't have an account? Sign Up here

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

```
[node1] (local) root@192.168.0.13 ~
$ ibmcloud login
API endpoint: https://cloud.ibm.com

Email> 2019103025@student.annauniv.edu

Password>
Authenticating...
OK

Targeted account Karthickeyan E's Account (0276affbdd74403daf5519bed5fce942)


Select a region (or press enter to skip):
1. au-syd
2. in-che
3. jp-osa
4. jp-tok
5. kr-seo
6. eu-de
7. eu-gb
8. ca-tor
9. us-south
10. us-east
11. br-sao
Enter a number>


API endpoint:      https://cloud.ibm.com
Region:
User:              2019103025@student.annauniv.edu
Account:           Karthickeyan E's Account (0276affbdd74403daf5519bed5fce942)
Resource group:    No resource group targeted, use 'ibmcloud target -g RESOURCE_GROUP'
CF API endpoint:
Org:
Space:
```

```
[node1] (local) root@192.168.0.13 ~
$ ibmcloud plugin install container-registry -r 'IBM Cloud'
Looking up 'container-registry' from repository 'IBM Cloud'...
Plug-in 'container-registry[cr] 1.0.2' found in repository 'IBM Cloud'
Plug-in 'container-registry 1.0.2' was already installed. Do you want to update it with 'container-registry[cr] 1.0.2' or not? [y/N] > n
Plugin installation was canceled.
[node1] (local) root@192.168.0.13 ~
$ ibmcloud cr region-set global
The region is set to 'global', the registry is 'icr.io'.

OK
[node1] (local) root@192.168.0.13 ~
$ ibmcloud cr namespace-add 2019103025
No resource group is targeted. Therefore, the default resource group for the account ('Default') is targeted.

Adding namespace '2019103025' in resource group 'Default' for account Karthickeyan E's Account in registry icr.io...

Successfully added namespace '2019103025'

OK
[node1] (local) root@192.168.0.13 ~
$ ibmcloud cr login
Logging 'docker' in to 'icr.io'...
Logged in to 'icr.io'.

OK
[node1] (local) root@192.168.0.13 ~
$ docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:faa03e786c97f07ef34423fcceeeec2398ec8a5759259f94d99078f264e9d7af
Status: Downloaded newer image for hello-world:latest
docker.io/library/hello-world:latest
```

```
No resource group is targeted. Therefore, the default resource group for the account ('Default') is targeted.

Adding namespace '2019103025' in resource group 'Default' for account Karthickeyan E's Account in registry icr.io...

Successfully added namespace '2019103025'

OK
[node1] (local) root@192.168.0.13 ~
$ ibmcloud cr login
Logging 'docker' in to 'icr.io'...
Logged in to 'icr.io'.

OK
[node1] (local) root@192.168.0.13 ~
$ docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:faa03e786c97f07ef34423fccceeec2398ec8a5759259f94d99078f264e9d7af
Status: Downloaded newer image for hello-world:latest
docker.io/library/hello-world:latest
[node1] (local) root@192.168.0.13 ~
$ docker tag hello-world icr.io/2019103025/test:latest            [node1] (local) root@192.168.0.13 ~
$ docker push icr.io/2019103025/test:latest
The push refers to repository [icr.io/2019103025/test]
e07ee1baac5f: Pushed
latest: digest: sha256:f54a58bc1aac5ea1a25d796ae155dc228b3f0e11d046ae276b39c4bf2f13d8c4 size: 525
[node1] (local) root@192.168.0.13 ~
$ ibmcloud cr image-list
Listing images...

Repository               Tag      Digest         Namespace    Created      Size     Security status
icr.io/2019103025/test   latest   f54a58bc1aac   2019103025   1 year ago   2.5 kB   -

OK
[node1] (local) root@192.168.0.13 ~
$
```

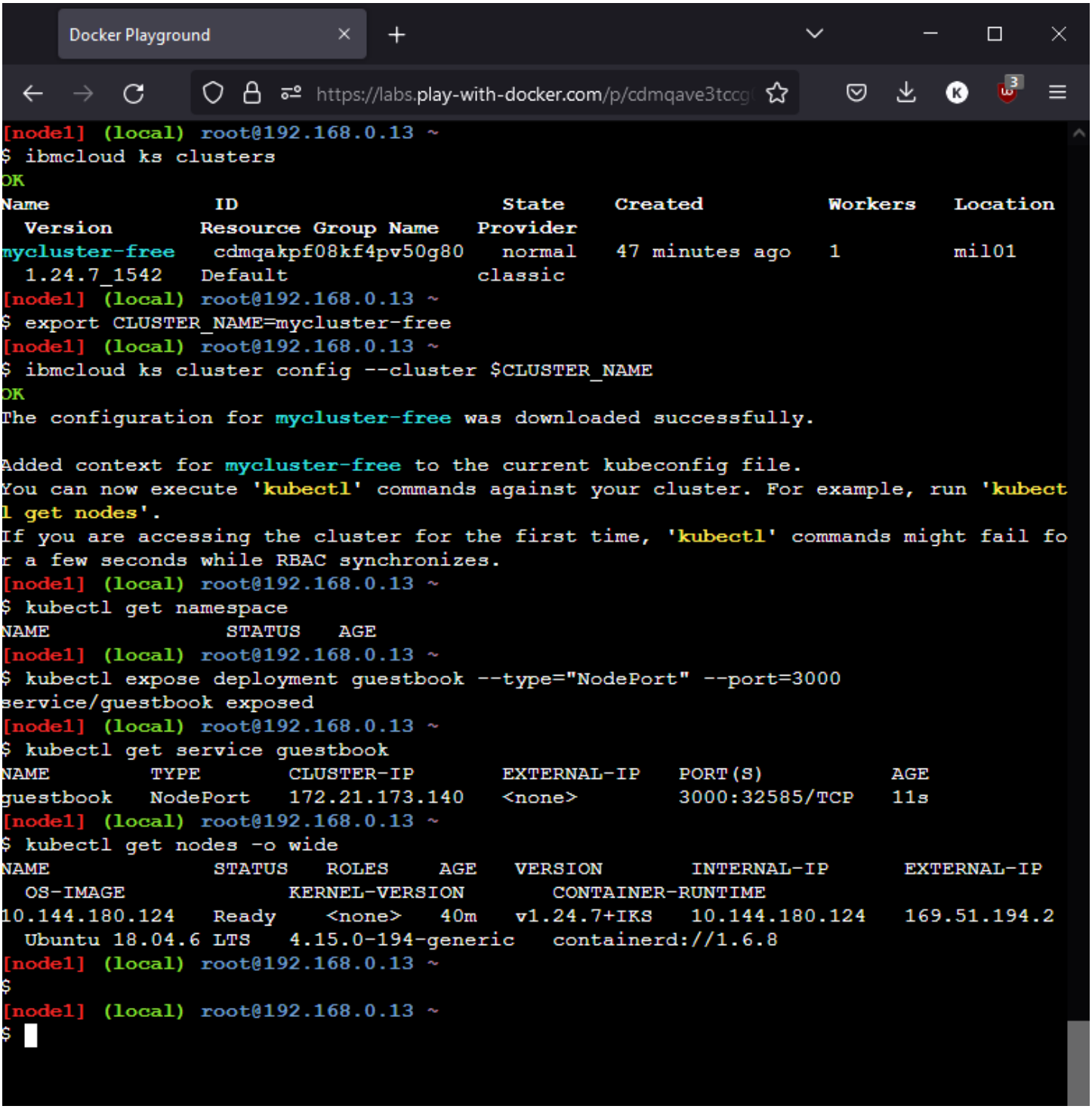
```
[node1] (local) root@192.168.0.13 ~
$ docker pull icr.io/2019103025/test
Using default tag: latest
latest: Pulling from 2019103025/test
Digest: sha256:f54a58bc1aac5ea1a25d796ae155dc228b3f0e11d046ae276b39c4bf2f13d8c4
Status: Image is up to date for icr.io/2019103025/test:latest
icr.io/2019103025/test:latest
[node1] (local) root@192.168.0.13 ~
$ docker images
REPOSITORY               TAG       IMAGE ID       CREATED         SIZE
hello-world              latest    feb5d9fea6a5   13 months ago   13.3kB
icr.io/2019103025/test   latest    feb5d9fea6a5   13 months ago   13.3kB
[node1] (local) root@192.168.0.13 ~
$ docker run icr.io/2019103025/test

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/

[node1] (local) root@192.168.0.13 ~
```

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.

```
# The PWD team.                                                #
################################################################
[node1] (local) root@192.168.0.13 ~
$ curl -fsSL https://clis.cloud.ibm.com/install/linux | sh
Current platform is linux64. Downloading corresponding IBM Cloud CLI...
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 11.9M  100 11.9M    0     0  25.8M      0 --:--:-- --:--:-- --:--:-- 25.8M
Download complete. Executing installer...
Bluemix_CLI/
Bluemix_CLI/bin/
Bluemix_CLI/bin/ibmcloud
Bluemix_CLI/bin/ibmcloud.sig
Bluemix_CLI/bin/NOTICE
Bluemix_CLI/bin/LICENSE
Bluemix_CLI/bin/CF_CLI_Notices.txt
Bluemix_CLI/bin/CF_CLI_SLC_Notices.txt
Bluemix_CLI/autocomplete/
Bluemix_CLI/autocomplete/bash_autocomplete
Bluemix_CLI/autocomplete/zsh_autocomplete
Bluemix_CLI/install
Bluemix_CLI/uninstall
Bluemix_CLI/install_bluemix_cli
Install complete.
[node1] (local) root@192.168.0.13 ~
$ ibmcloud login
API endpoint: https://cloud.ibm.com

Email> 2019103025@student.annauniv.edu

Password>
Authenticating...
OK

Targeted account Karthickeyan E's Account (0276affbdd74403daf5519bed5fce942)
```

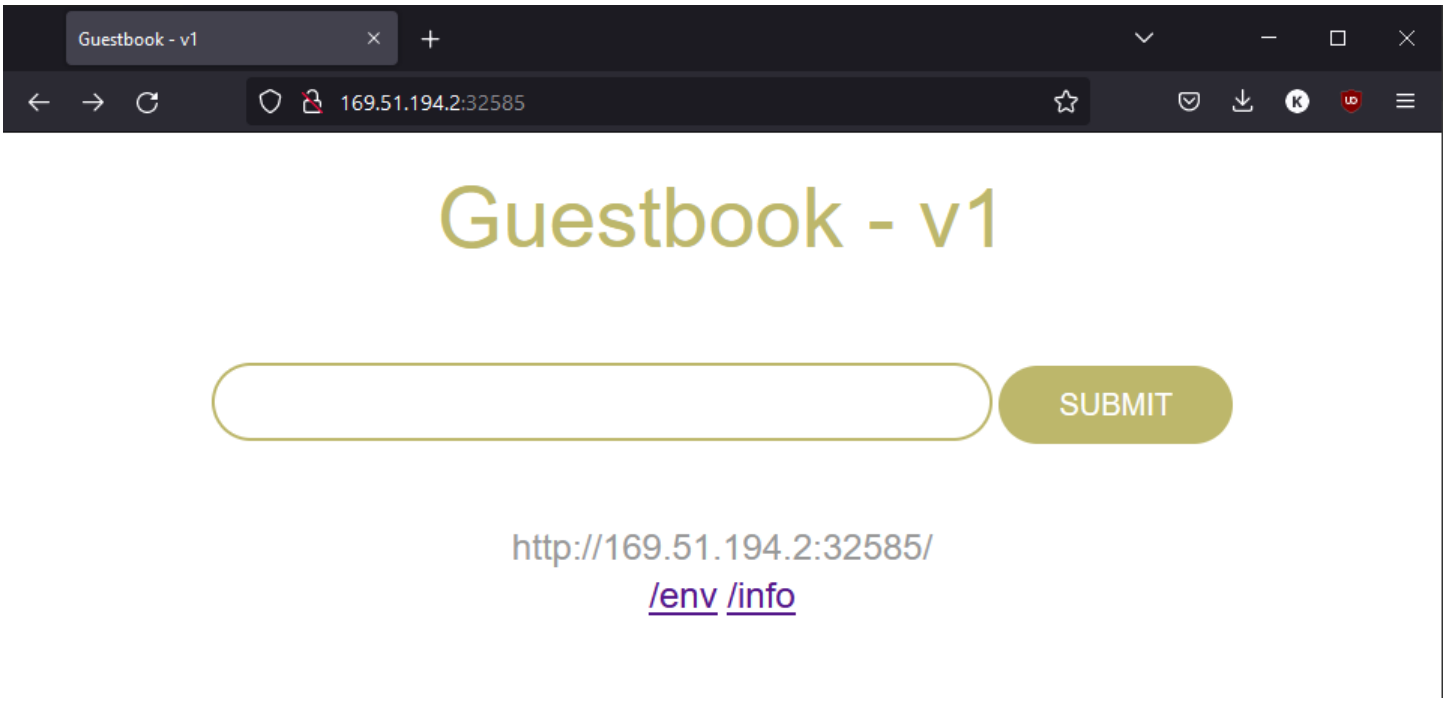
```
[node1] (local) root@192.168.0.13 ~
$ ibmcloud ks clusters
OK
Name             ID                     State    Created          Workers   Location   Version       Resource Group Name   Provider
mycluster-free   cdmqakpf08kf4pv50g80   normal   47 minutes ago   1         mil01      1.24.7_1542   Default               classic
[node1] (local) root@192.168.0.13 ~
$ export CLUSTER_NAME=mycluster-free
[node1] (local) root@192.168.0.13 ~
$ ibmcloud ks cluster config --cluster $CLUSTER_NAME
OK
The configuration for mycluster-free was downloaded successfully.

Added context for mycluster-free to the current kubeconfig file.
You can now execute 'kubectl' commands against your cluster. For example, run 'kubectl get nodes'.
If you are accessing the cluster for the first time, 'kubectl' commands might fail for a few seconds while RBAC synchronizes.
[node1] (local) root@192.168.0.13 ~
$ kubectl get namespace
NAME              STATUS   AGE
default           Active   41m
ibm-cert-store    Active   30m
ibm-operators     Active   38m
ibm-system        Active   41m
kube-node-lease   Active   41m
kube-public       Active   41m
kube-system       Active   41m
[node1] (local) root@192.168.0.13 ~
$
```

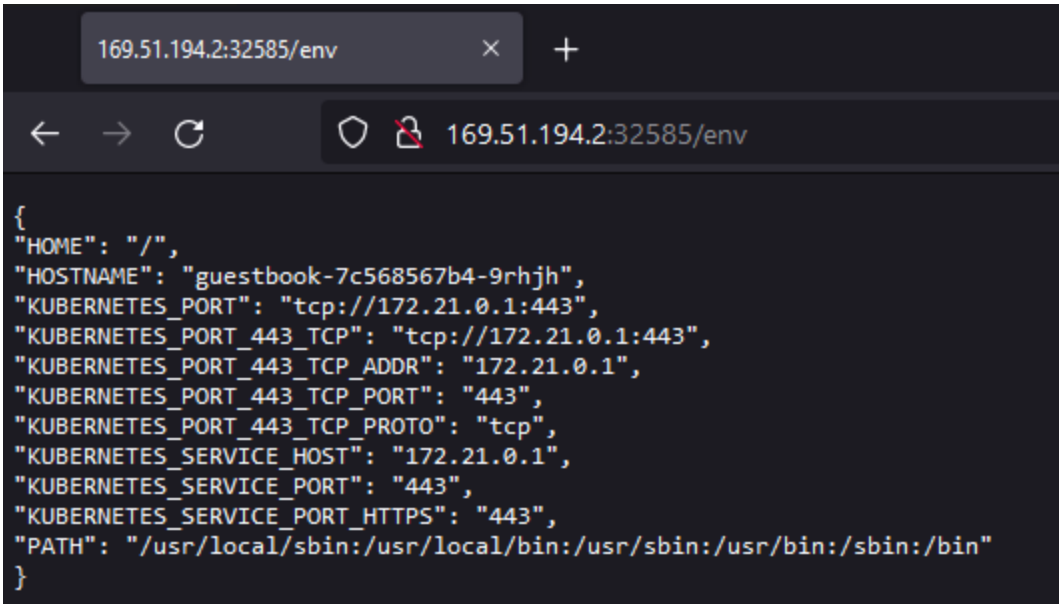
```
[node1] (local) root@192.168.0.13 ~
$ git clone https://github.com/IBM/guestbook.git
Cloning into 'guestbook'...
remote: Enumerating objects: 448, done.
remote: Total 448 (delta 0), reused 0 (delta 0), pack-reused 448
Receiving objects: 100% (448/448), 205.42 KiB | 11.41 MiB/s, done.
Resolving deltas: 100% (264/264), done.
[node1] (local) root@192.168.0.13 ~
$ git clone https://github.com/IBM/kube101.git
Cloning into 'kube101'...
remote: Enumerating objects: 678, done.
remote: Counting objects: 100% (188/188), done.
remote: Compressing objects: 100% (135/135), done.
remote: Total 678 (delta 58), reused 125 (delta 27), pack-reused 490
Receiving objects: 100% (678/678), 2.83 MiB | 23.17 MiB/s, done.
Resolving deltas: 100% (298/298), done.
[node1] (local) root@192.168.0.13 ~
$ kubectl create deployment guestbook --image=ibmcom/guestbook:v1
deployment.apps/guestbook created
[node1] (local) root@192.168.0.13 ~
$ kubectl get pods
NAME                         READY   STATUS    RESTARTS   AGE
guestbook-7c568567b4-9rhjh   1/1     Running   0          37s
[node1] (local) root@192.168.0.13 ~
$
```


Docker Playground
https://labs.play-with-docker.com/p/cdmqave3tccg
[node1] (local) root@192.168.0.13 ~
$ ibmcloud ks clusters
OK
Name ID State Created Workers Location
Version Resource Group Name Provider
mycluster-free cdmqakpf08kf4pv50g80 normal 47 minutes ago 1 mil01
1.24.7_1542 Default classic
[node1] (local) root@192.168.0.13 ~
$ export CLUSTER_NAME=mycluster-free
[node1] (local) root@192.168.0.13 ~
$ ibmcloud ks cluster config --cluster $CLUSTER_NAME
OK
The configuration for mycluster-free was downloaded successfully.
Added context for mycluster-free to the current kubeconfig file.
You can now execute 'kubectl' commands against your cluster. For example, run 'kubectl get nodes'.
If you are accessing the cluster for the first time, 'kubectl' commands might fail for a few seconds while RBAC synchronizes.
[node1] (local) root@192.168.0.13 ~
$ kubectl get namespace
NAME STATUS AGE
[node1] (local) root@192.168.0.13 ~
$ kubectl expose deployment guestbook --type="NodePort" --port=3000
service/guestbook exposed
[node1] (local) root@192.168.0.13 ~
$ kubectl get service guestbook
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
guestbook NodePort 172.21.173.140 <none> 3000:32585/TCP 11s
[node1] (local) root@192.168.0.13 ~
$ kubectl get nodes -o wide
NAME STATUS ROLES AGE VERSION INTERNAL-IP EXTERNAL-IP
OS-IMAGE KERNEL-VERSION CONTAINER-RUNTIME
10.144.180.124 Ready <none> 40m v1.24.7+IKS 10.144.180.124 169.51.194.2
Ubuntu 18.04.6 LTS 4.15.0-194-generic containerd://1.6.8
[node1] (local) root@192.168.0.13 ~
$
[node1] (local) root@192.168.0.13 ~
$

```
[node1] (local) root@192.168.0.13 ~
$ kubectl expose deployment guestbook --type="NodePort" --port=3000
service/guestbook exposed
[node1] (local) root@192.168.0.13 ~
$ kubectl get service guestbook
NAME        TYPE       CLUSTER-IP       EXTERNAL-IP   PORT(S)          AGE
guestbook   NodePort   172.21.173.140   <none>        3000:32585/TCP   11s
[node1] (local) root@192.168.0.13 ~
$ kubectl get nodes -o wide
NAME             STATUS   ROLES    AGE   VERSION       INTERNAL-IP      EXTERNAL-IP    OS-IMAGE             KERNEL-VERSION       CONTAINER-RUNTIME
10.144.180.124   Ready    <none>   40m   v1.24.7+IKS   10.144.180.124   169.51.194.2   Ubuntu 18.04.6 LTS   4.15.0-194-generic   containerd://1.6.8
[node1] (local) root@192.168.0.13 ~
$
```

```
service/guestbook exposed
[node1] (local) root@192.168.0.13 ~
$ kubectl get service guestbook
NAME        TYPE       CLUSTER-IP       EXTERNAL-IP   PORT(S)          AGE
guestbook   NodePort   172.21.173.140   <none>        3000:32585/TCP   11s
[node1] (local) root@192.168.0.13 ~
$ kubectl get nodes -o wide
NAME             STATUS   ROLES    AGE   VERSION       INTERNAL-IP      EXTERNAL-I
P     OS-IMAGE             KERNEL-VERSION       CONTAINER-RUNTIME
10.144.180.124   Ready    <none>   40m   v1.24.7+IKS   10.144.180.124   169.51.194
.2    Ubuntu 18.04.6 LTS   4.15.0-194-generic   containerd://1.6.8
[node1] (local) root@192.168.0.13 ~
$
```




169.51.194.2:32585/env
169.51.194.2:32585/env
{
"HOME": "/",
"HOSTNAME": "guestbook-7c568567b4-9rhjh",
"KUBERNETES_PORT": "tcp://172.21.0.1:443",
"KUBERNETES_PORT_443_TCP": "tcp://172.21.0.1:443",
"KUBERNETES_PORT_443_TCP_ADDR": "172.21.0.1",
"KUBERNETES_PORT_443_TCP_PORT": "443",
"KUBERNETES_PORT_443_TCP_PROTO": "tcp",
"KUBERNETES_SERVICE_HOST": "172.21.0.1",
"KUBERNETES_SERVICE_PORT": "443",
"KUBERNETES_SERVICE_PORT_HTTPS": "443",
"PATH": "/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
}