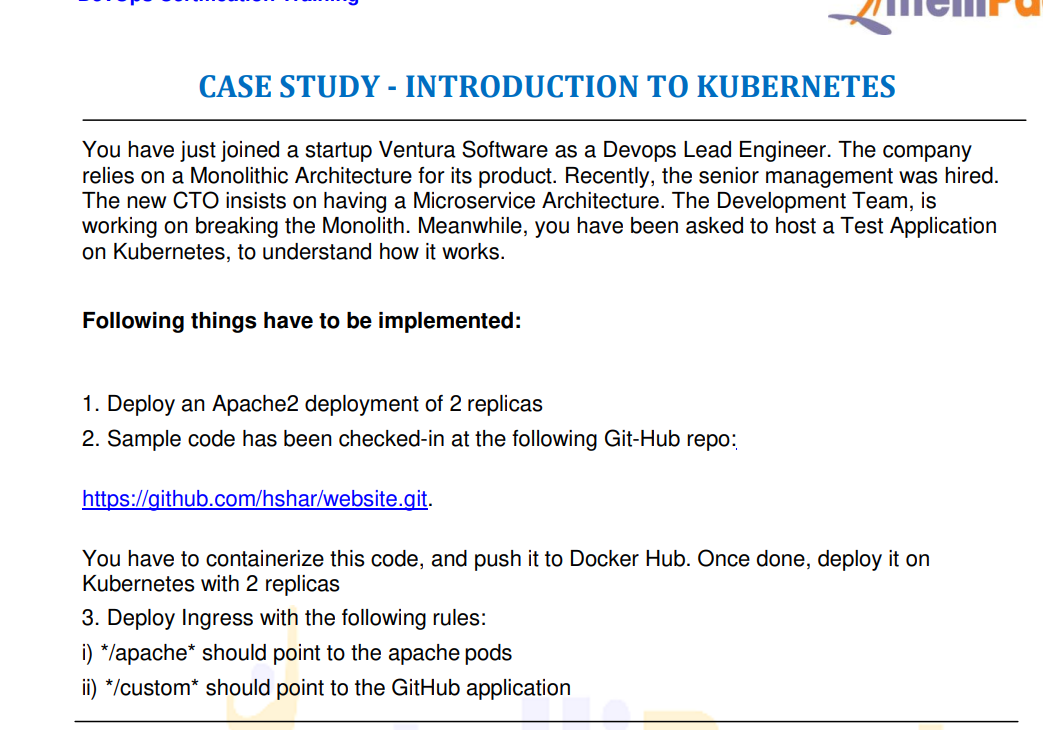
Kubernetes case study



Commands used:

ubuntu@ip-172-31-88-214:~$ ls -lrt

total 16

-rw-r--r-- 1 root root 336 Mar 4 13:56 k.sh

-rw-r--r-- 1 root root 339 Mar 4 14:07 apache.yml

drwxrwxr-x 4 ubuntu ubuntu 4096 Mar 4 14:14 website

-rw-r--r-- 1 root root 346 Mar 4 14:25 web1.yml

ubuntu@ip-172-31-88-214:~$ kubectl get deploy

NAME READY UP-TO-DATE AVAILABLE AGE

apache-deployment 2/2 2 2 162m

web-deployment 2/2 2 2 144m

ubuntu@ip-172-31-88-214:~$ kubectl get pods -o wide

NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES

apache-deployment-76695bf577-knnzn 1/1 Running 1 162m 192.168.112.14 ip-172-31-86-178 <none> <none>

apache-deployment-76695bf577-trfpp 1/1 Running 1 162m 192.168.112.13 ip-172-31-86-178 <none> <none>

web-deployment-5fc77fc4d9-f474c 1/1 Running 1 144m 192.168.112.15 ip-172-31-86-178 <none> <none>

web-deployment-5fc77fc4d9-lfczw 1/1 Running 1 144m 192.168.112.16 ip-172-31-86-178 <none> <none>

ubuntu@ip-172-31-88-214:~$ sudo nano svc1.yml

ubuntu@ip-172-31-88-214:~$ kubectl create -f svc1.yml

service/svc1 created

ubuntu@ip-172-31-88-214:~$ kubectl get svc

NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE

kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 173m

svc1 ClusterIP 10.100.185.92 <none> 80/TCP 18s

ubuntu@ip-172-31-88-214:~$ curl -i 10.100.185.92

HTTP/1.1 200 OK

Date: Fri, 04 Mar 2022 16:54:25 GMT

Server: Apache/2.4.52 (Unix)

Last-Modified: Fri, 04 Mar 2022 14:10:28 GMT

ETag: "c1-5d96513a41d00"

Accept-Ranges: bytes

Content-Length: 193

Content-Type: text/html

<html>

<head>

<title> Intellipaat </title>

</head>

<body style = "background-image:url('images/github3.jpg'); background-size: 100%">

<h2 ALIGN=CENTER>Hello world!</h2>

</body>

</html>

ubuntu@ip-172-31-88-214:~$ cat apache.yml

apiVersion: apps/v1

kind: Deployment

metadata:

name: apache-deployment

labels:

app: apache

spec:

replicas: 2

selector:

matchLabels:

app: apache

template:

metadata:

labels:

app: apache

spec:

containers:

- name: apache

image: httpd

ports:

- containerPort: 80

ubuntu@ip-172-31-88-214:~$ sudo nano svc2.yml

ubuntu@ip-172-31-88-214:~$ kubectl create -f svc2.yml

service/svc2 created

ubuntu@ip-172-31-88-214:~$ kubectl get svc

NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE

kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 178m

svc1 ClusterIP 10.100.185.92 <none> 80/TCP 5m46s

svc2 ClusterIP 10.105.156.60 <none> 80/TCP 12s

ubuntu@ip-172-31-88-214:~$ curl -i 10.105.156.60

HTTP/1.1 200 OK

Date: Fri, 04 Mar 2022 16:59:40 GMT

Server: Apache/2.4.52 (Unix)

Last-Modified: Mon, 11 Jun 2007 18:53:14 GMT

ETag: "2d-432a5e4a73a80"

Accept-Ranges: bytes

Content-Length: 45

Content-Type: text/html

<html><body><h1>It works!</h1></body></html>

ubuntu@ip-172-31-88-214:~$ sudo nano ingress1.yml

ubuntu@ip-172-31-88-214:~$ kubectl create -f ingress1.yml

ingress.networking.k8s.io/apache-ingress created

ubuntu@ip-172-31-88-214:~$ kubectl get ingress

NAME CLASS HOSTS ADDRESS PORTS AGE

apache-ingress <none> \* 80 19s

ubuntu@ip-172-31-88-214:~$ kubectl get svc --all-namespaces

NAMESPACE NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE

default kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 3h2m

default svc1 ClusterIP 10.100.185.92 <none> 80/TCP 9m15s

default svc2 ClusterIP 10.105.156.60 <none> 80/TCP 3m41s

ingress-nginx ingress-nginx-controller NodePort 10.109.9.125 <none> 80:31446/TCP,443:31667/TCP 179m

ingress-nginx ingress-nginx-controller-admission ClusterIP 10.105.54.54 <none> 443/TCP 179m

kube-system kube-dns ClusterIP 10.96.0.10 <none> 53/UDP,53/TCP,9153/TCP 3h2m

ubuntu@ip-172-31-88-214:~$ sudo nano ingress1.yml

ubuntu@ip-172-31-88-214:~$ kubectl delete ing apache-ingress

ingress.networking.k8s.io "apache-ingress" deleted

ubuntu@ip-172-31-88-214:~$ kubectl create -f ingress1.yml

ingress.networking.k8s.io/apache-ingress created

ubuntu@ip-172-31-88-214:~$ history

1 clear

2 ls -lrt

3 kubectl get deploy

4 kubectl get pods -o wide

5 sudo nano svc1.yml

6 kubectl create -f svc1.yml

7 kubectl get svc

8 curl -i 10.100.185.92

9 cat apache.yml

10 sudo nano svc2.yml

11 kubectl create -f svc2.yml

12 kubectl get svc

13 curl -i 10.105.156.60

14 sudo nano ingress1.yml

15 kubectl create -f ingress1.yml

16 kubectl get ingress

17 kubectl get svc --all-namespaces

18 sudo nano ingress1.yml

19 kubectl delete ing apache-ingress

20 kubectl create -f ingress1.yml

21 history

ubuntu@ip-172-31-88-214:~$ ls -lrt

total 28

-rw-r--r-- 1 root root 336 Mar 4 13:56 k.sh

-rw-r--r-- 1 root root 339 Mar 4 14:07 apache.yml

drwxrwxr-x 4 ubuntu ubuntu 4096 Mar 4 14:14 website

-rw-r--r-- 1 root root 346 Mar 4 14:25 web1.yml

-rw-r--r-- 1 root root 146 Mar 4 16:52 svc1.yml

-rw-r--r-- 1 root root 148 Mar 4 16:58 svc2.yml

-rw-r--r-- 1 root root 483 Mar 4 17:05 ingress1.yml

ubuntu@ip-172-31-88-214:~$ cat apache.yml

apiVersion: apps/v1

kind: Deployment

metadata:

name: apache-deployment

labels:

app: apache

spec:

replicas: 2

selector:

matchLabels:

app: apache

template:

metadata:

labels:

app: apache

spec:

containers:

- name: apache

image: httpd

ports:

- containerPort: 80

ubuntu@ip-172-31-88-214:~$ cat web1.yml

apiVersion: apps/v1

kind: Deployment

metadata:

name: web-deployment

labels:

app: web1

spec:

replicas: 2

selector:

matchLabels:

app: web1

template:

metadata:

labels:

app: web1

spec:

containers:

- name: apache4march

image: kashmir8/image1

ports:

- containerPort: 80

ubuntu@ip-172-31-88-214:~$ cat svc1.yml

apiVersion: v1

kind: Service

metadata:

name: svc1

spec:

type: ClusterIP

ports:

- targetPort: 80

port: 80

selector:

app: web1

ubuntu@ip-172-31-88-214:~$ cat svc2.yml

apiVersion: v1

kind: Service

metadata:

name: svc2

spec:

type: ClusterIP

ports:

- targetPort: 80

port: 80

selector:

app: apache

ubuntu@ip-172-31-88-214:~$ cat ingress1.yml

apiVersion: networking.k8s.io/v1

kind: Ingress

metadata:

name: apache-ingress

annotations:

nginx.ingress.kubernetes.io/rewrite-target: /

spec:

rules:

- http:

paths:

- path: /custom1

pathType: Prefix

backend:

service:

name: svc1

port:

number: 80

- path: /custom2

pathType: Prefix

backend:

service:

name: svc2

port:

number: 80

ubuntu@ip-172-31-88-214:~$ cd website/

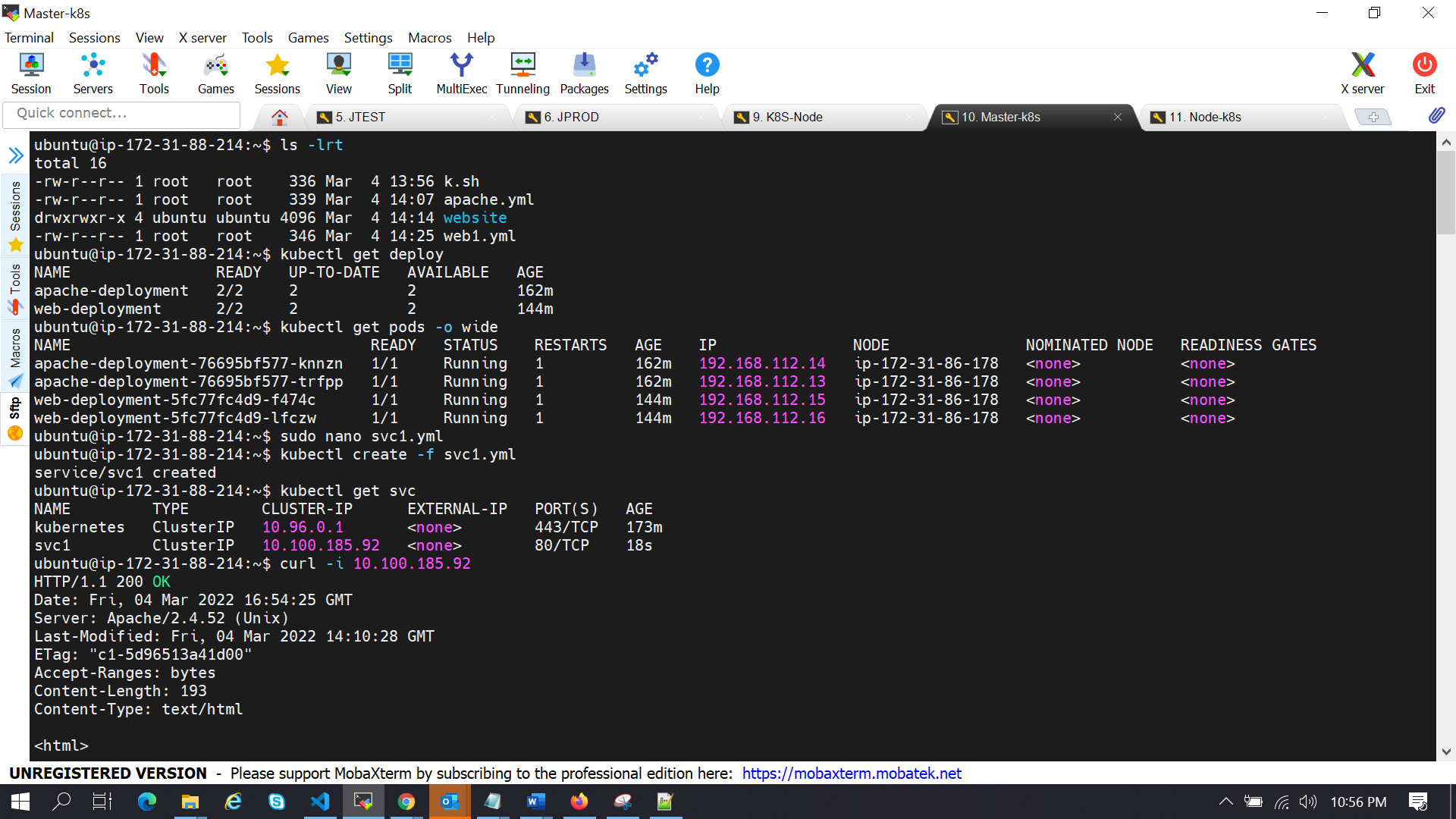
ubuntu@ip-172-31-88-214:~/website$ ls -lrt

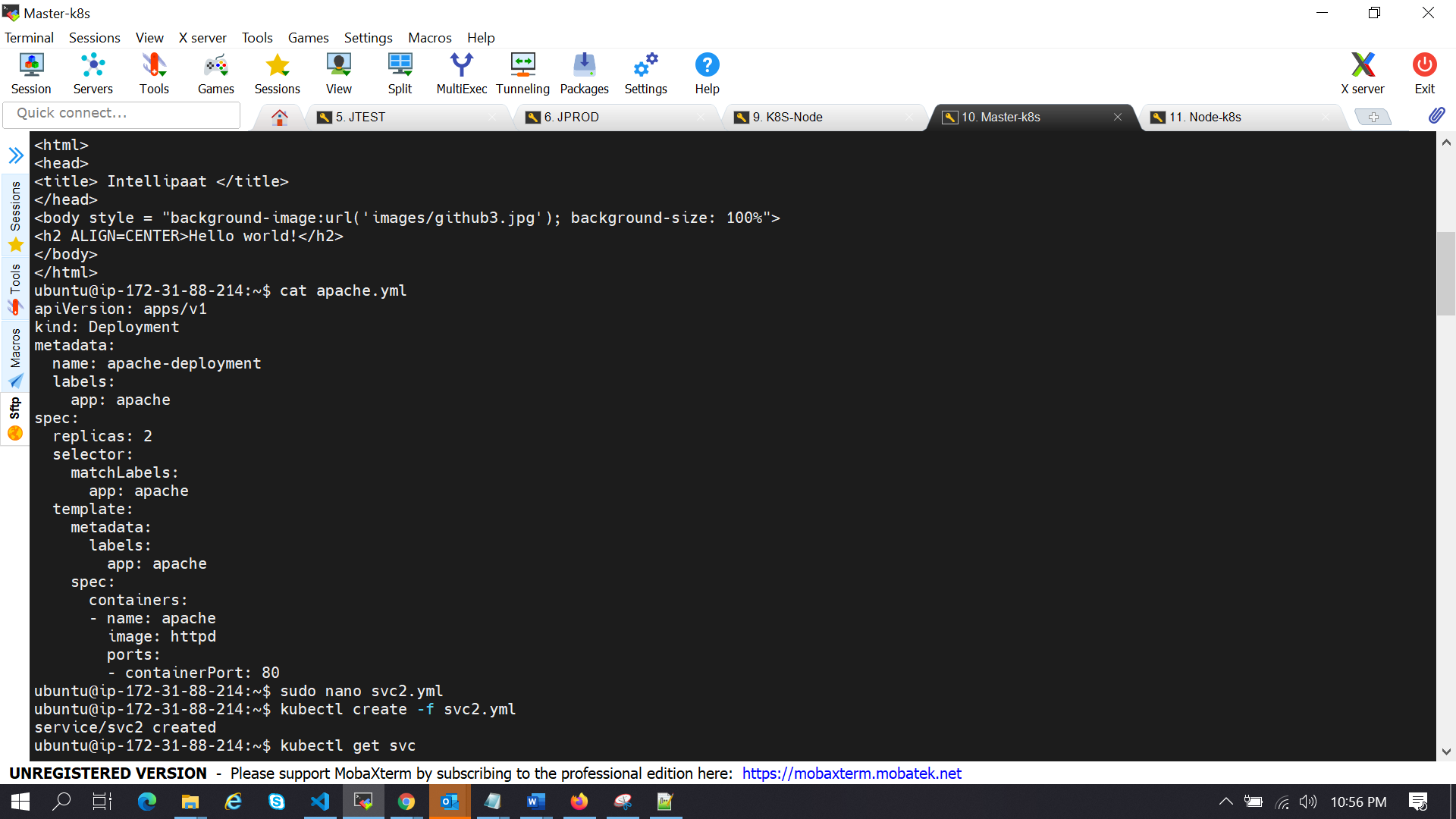
total 12

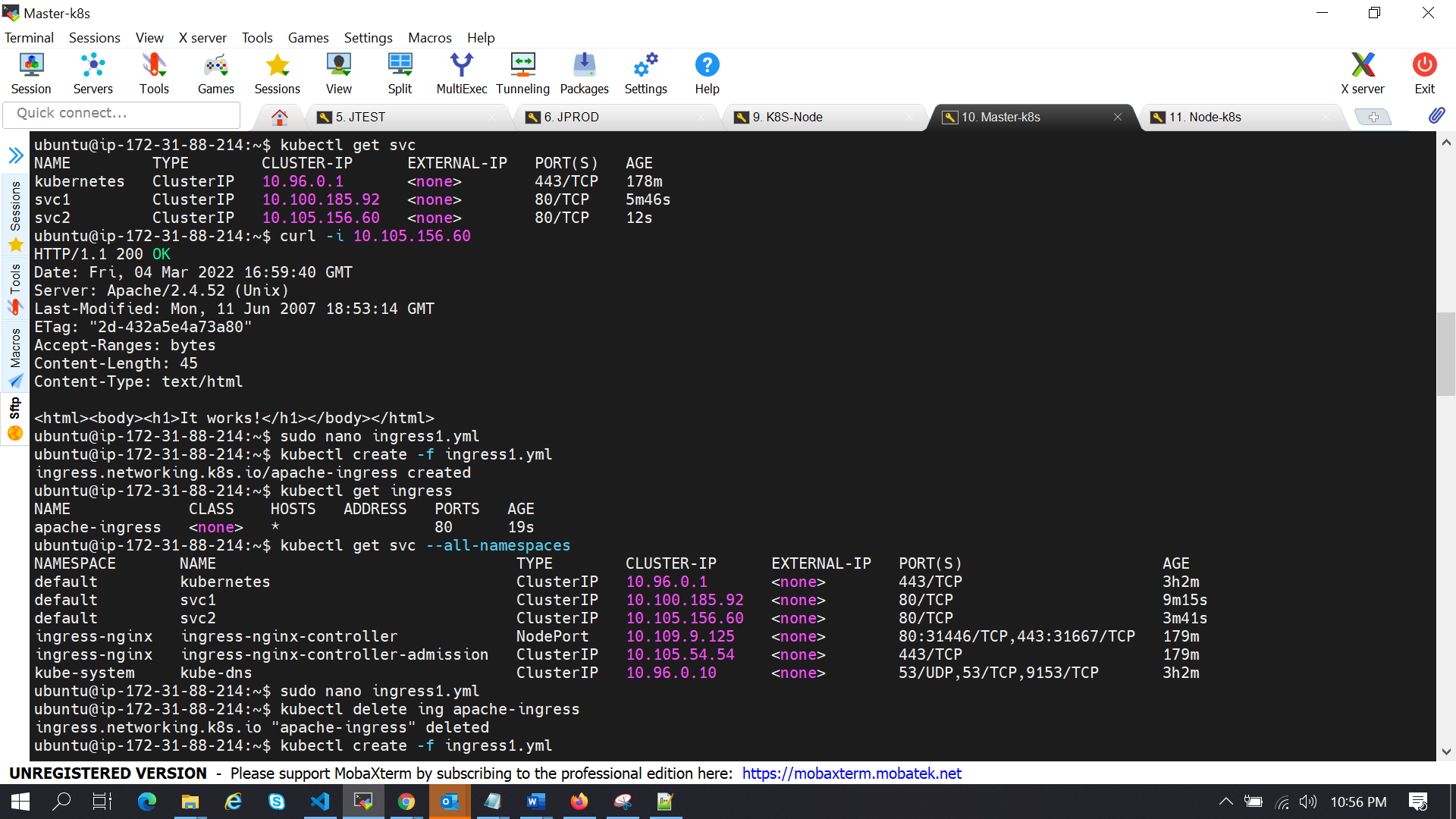
-rw-rw-r-- 1 ubuntu ubuntu 193 Mar 4 14:10 index.html

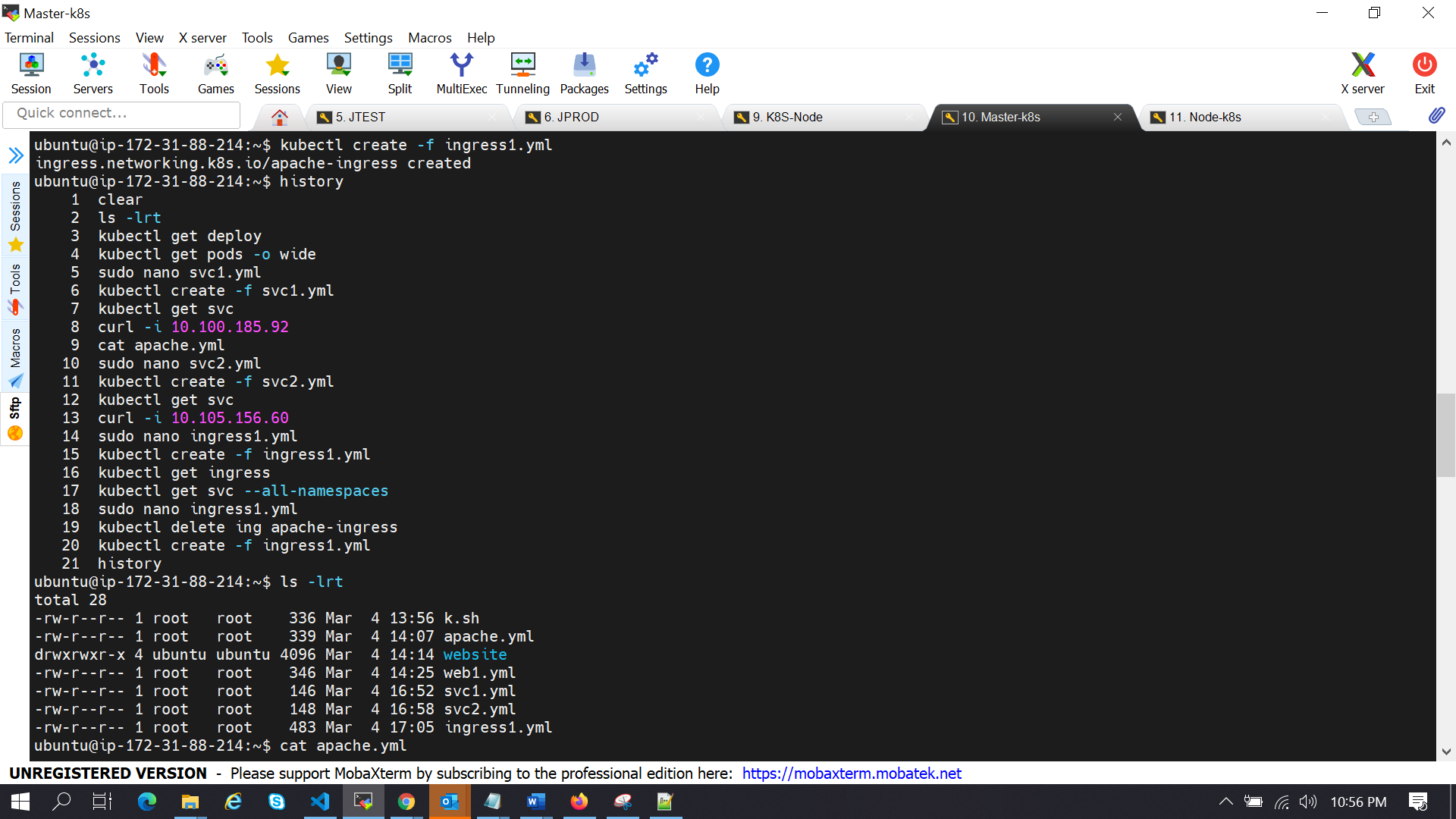
drwxrwxr-x 2 ubuntu ubuntu 4096 Mar 4 14:10 images

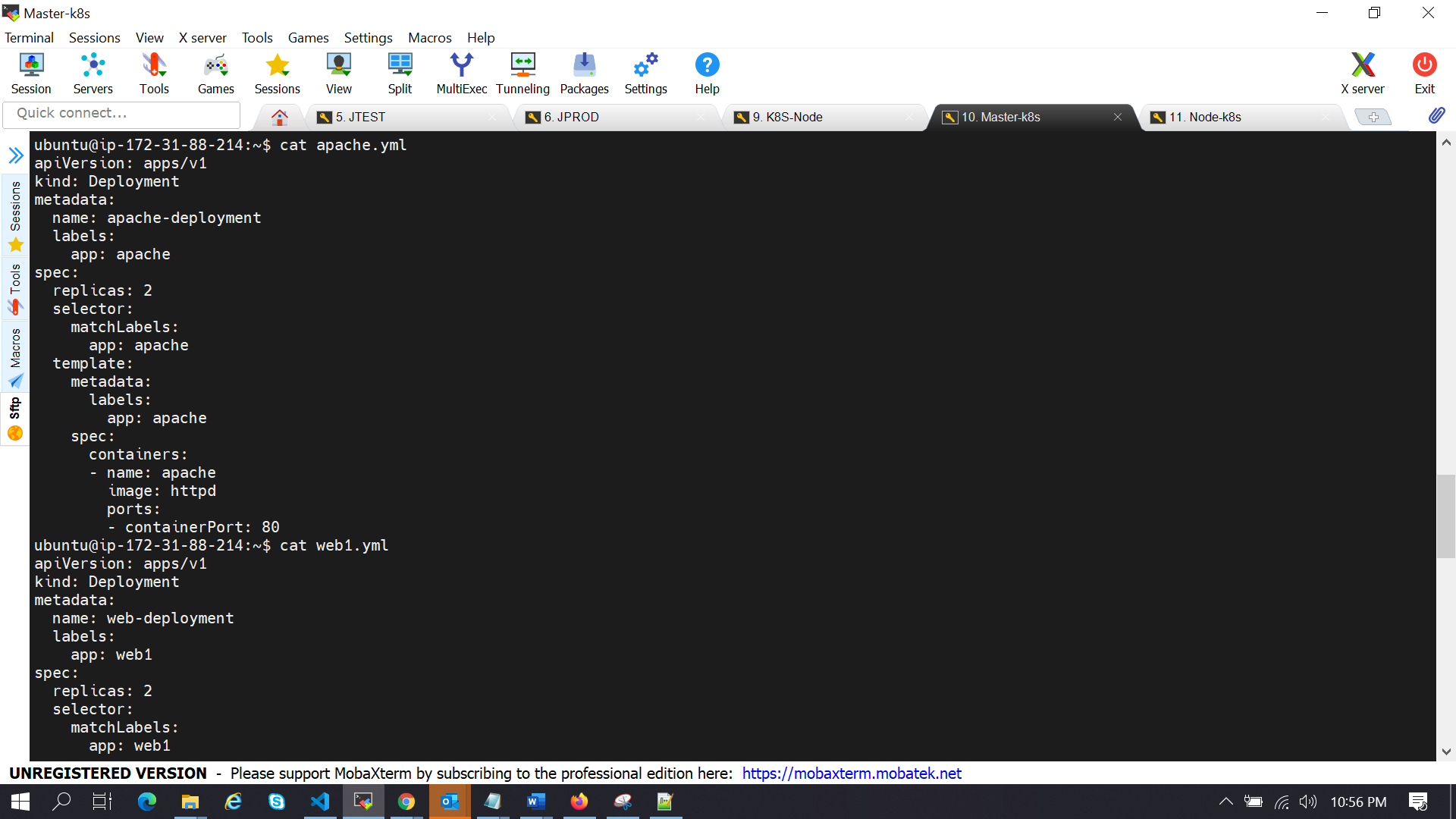
-rw-r--r-- 1 root root 49 Mar 4 14:14 Dockerfile

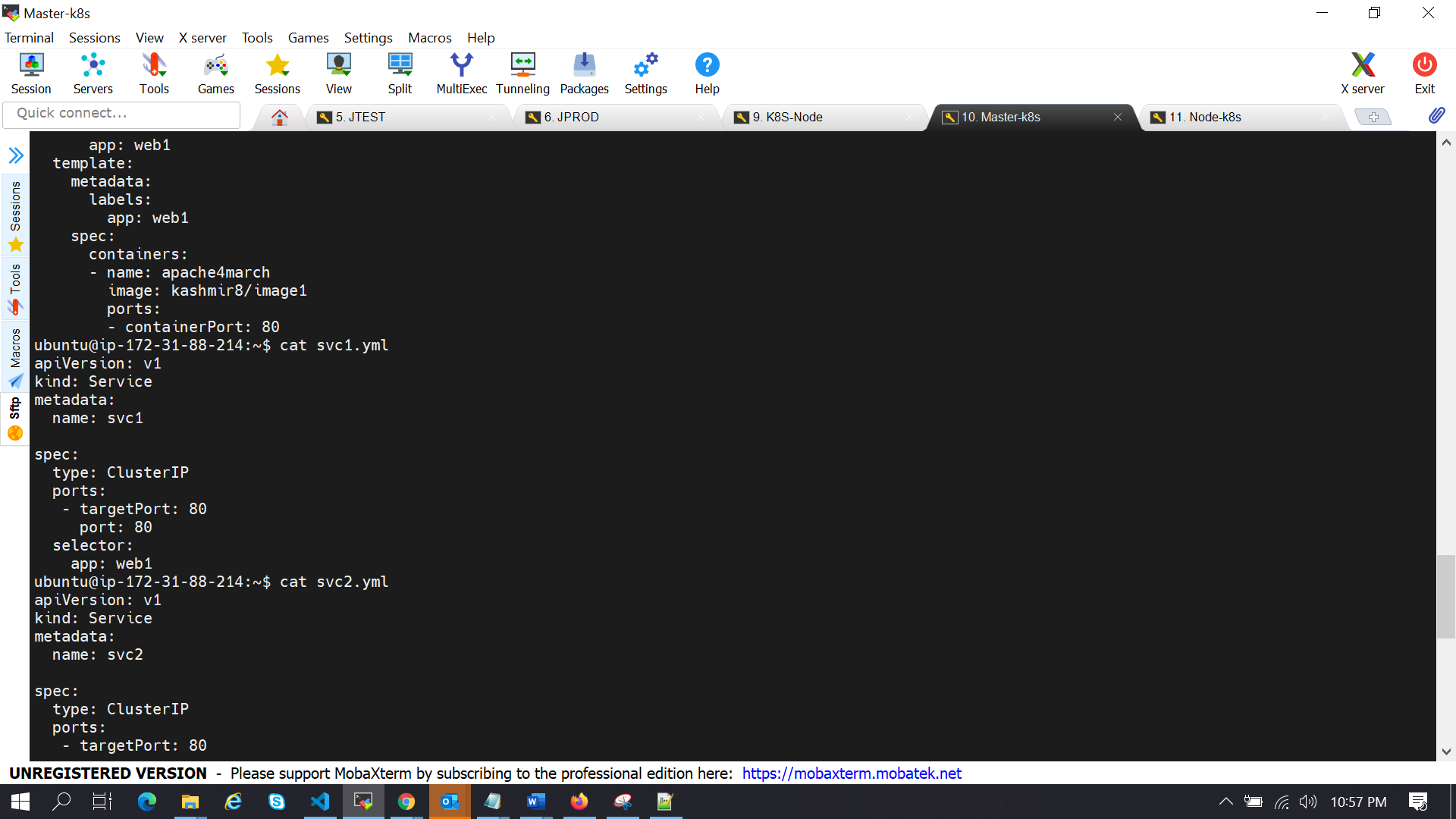


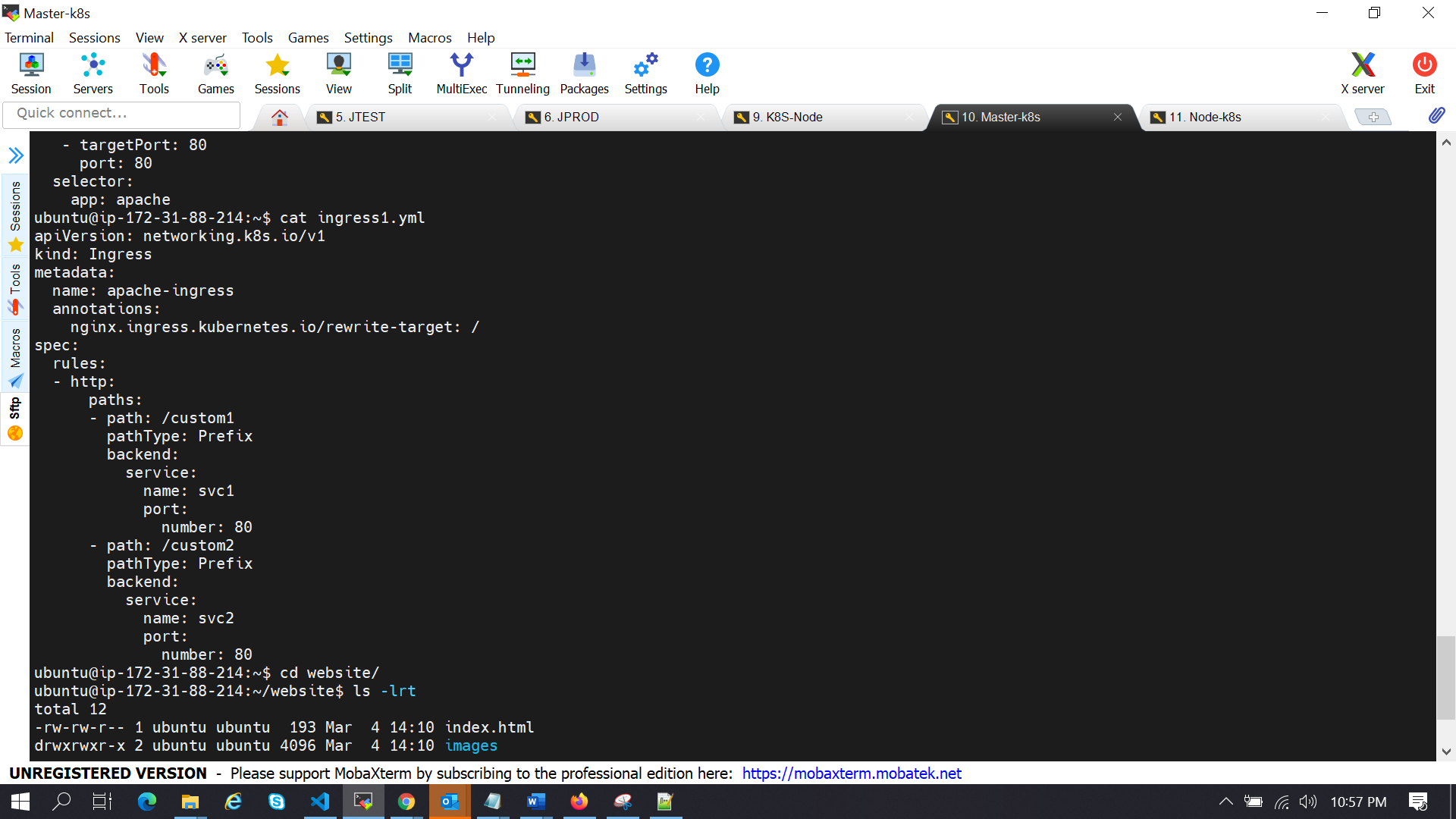


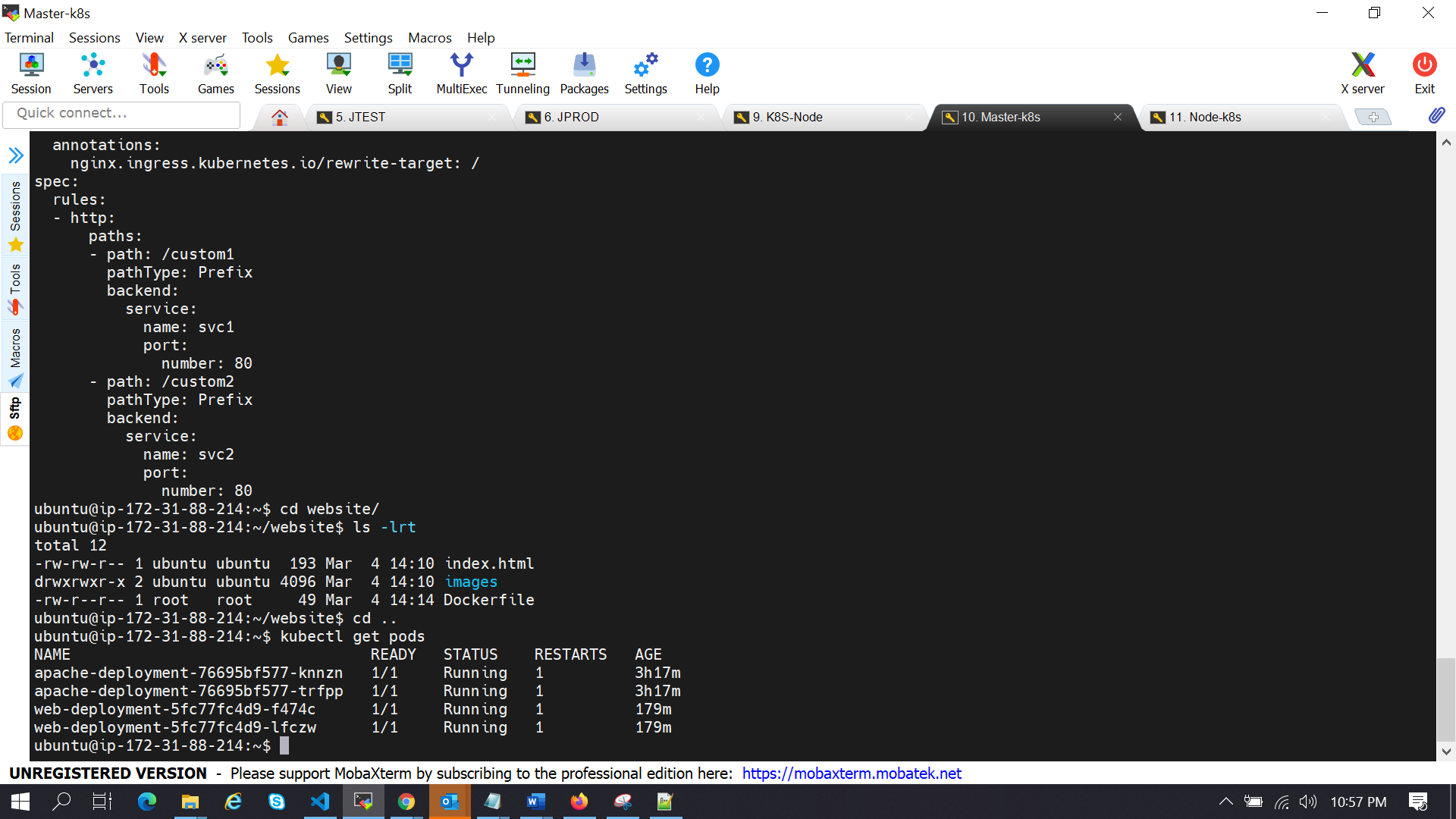




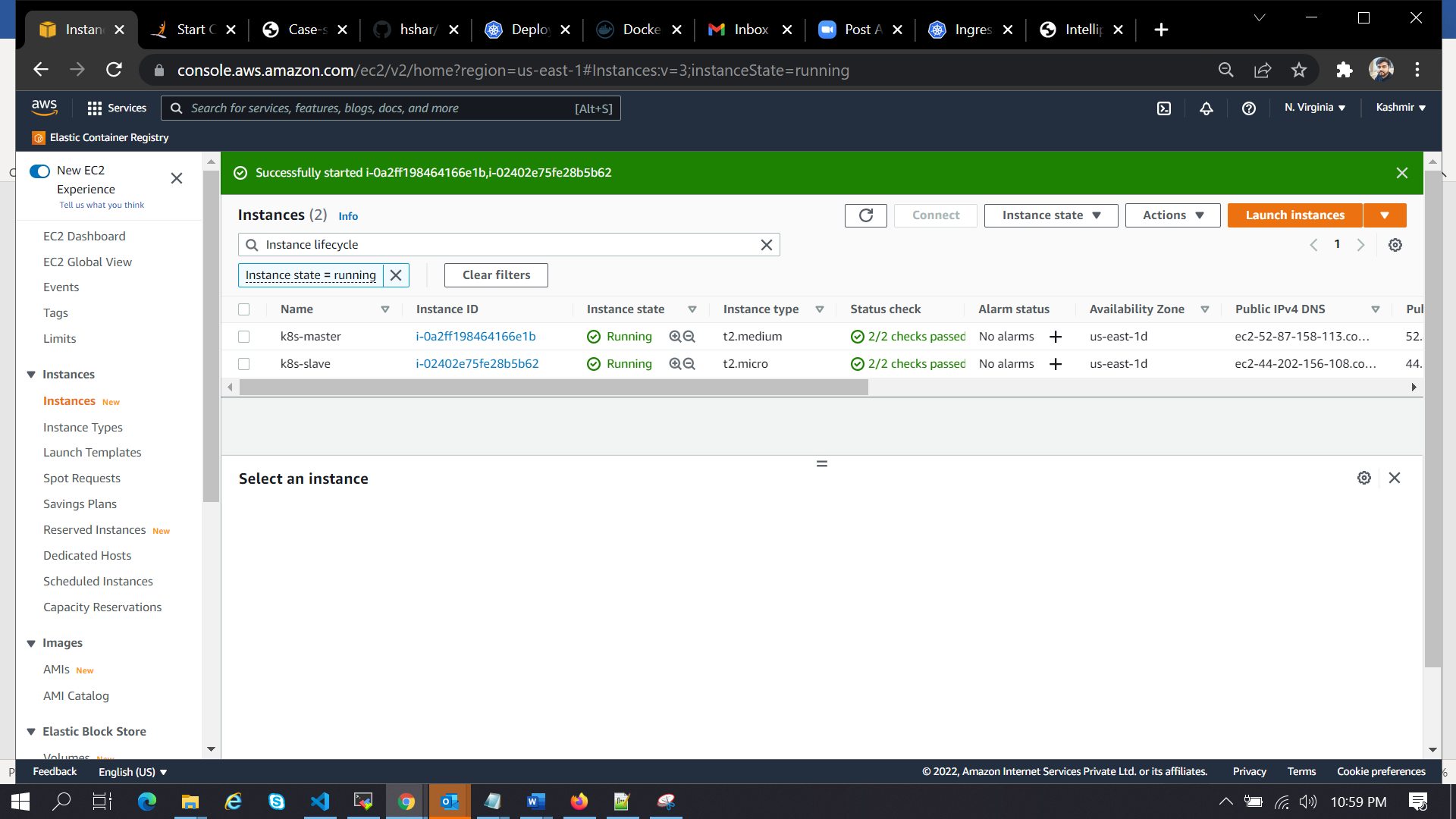


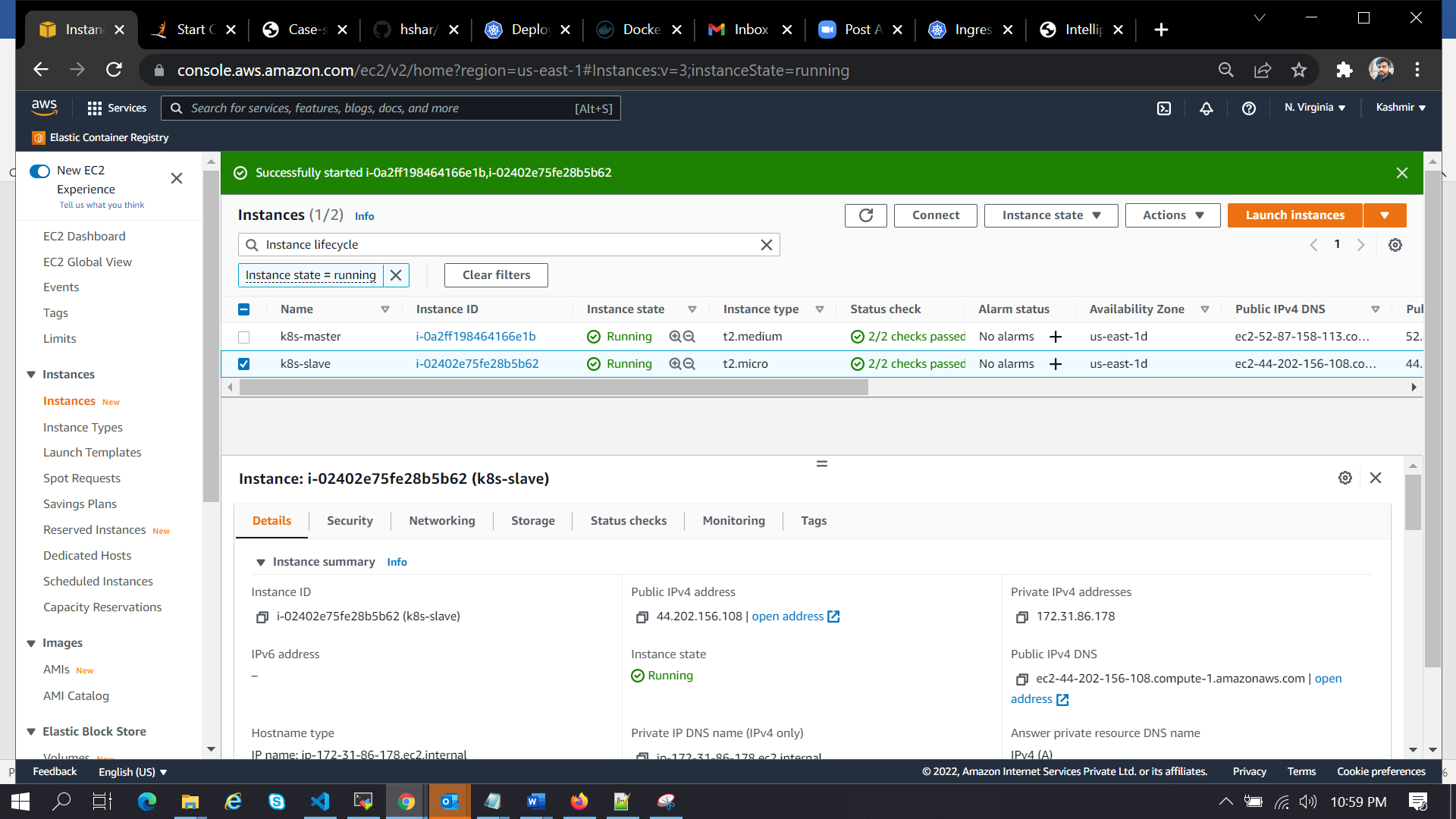


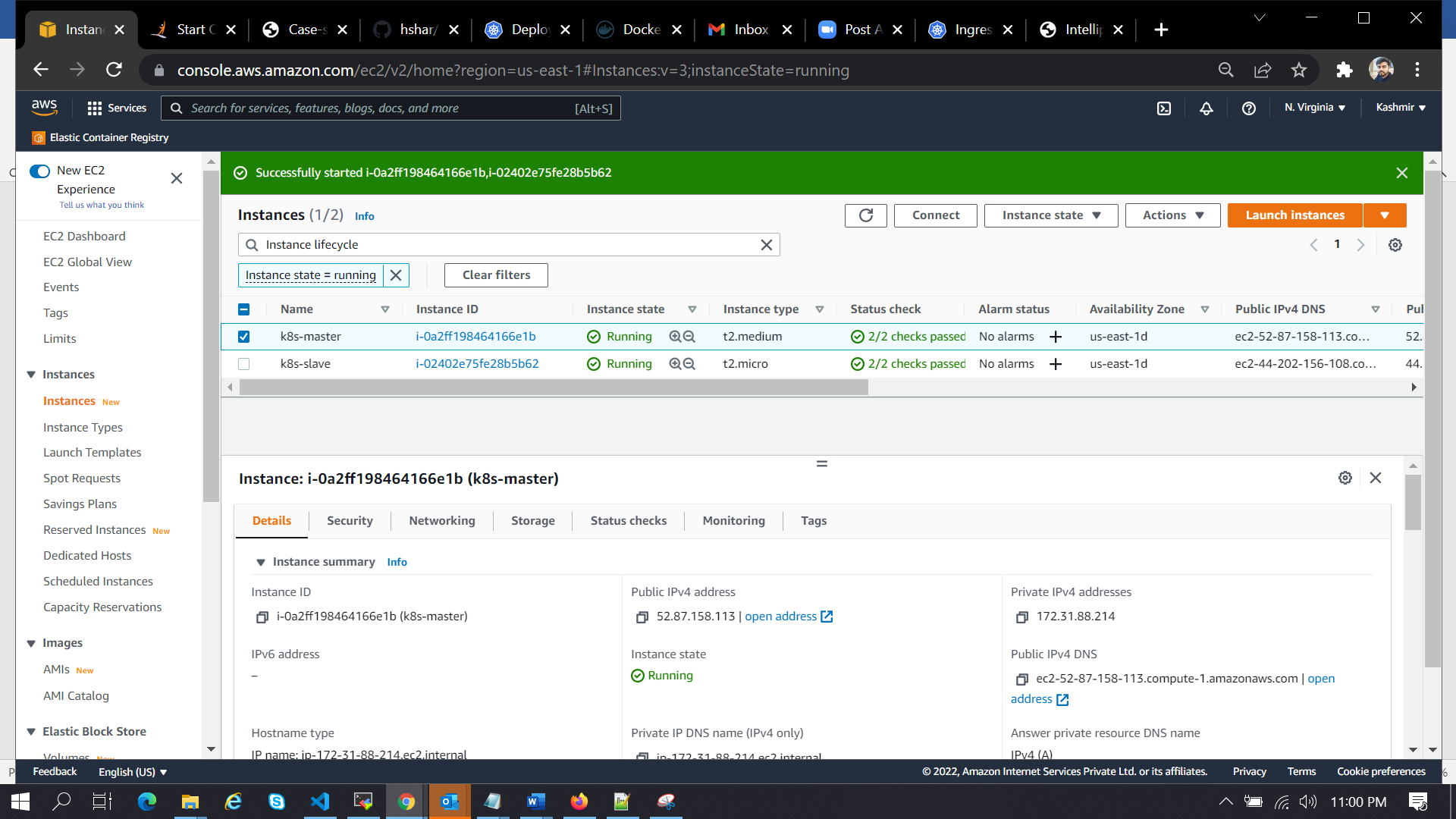


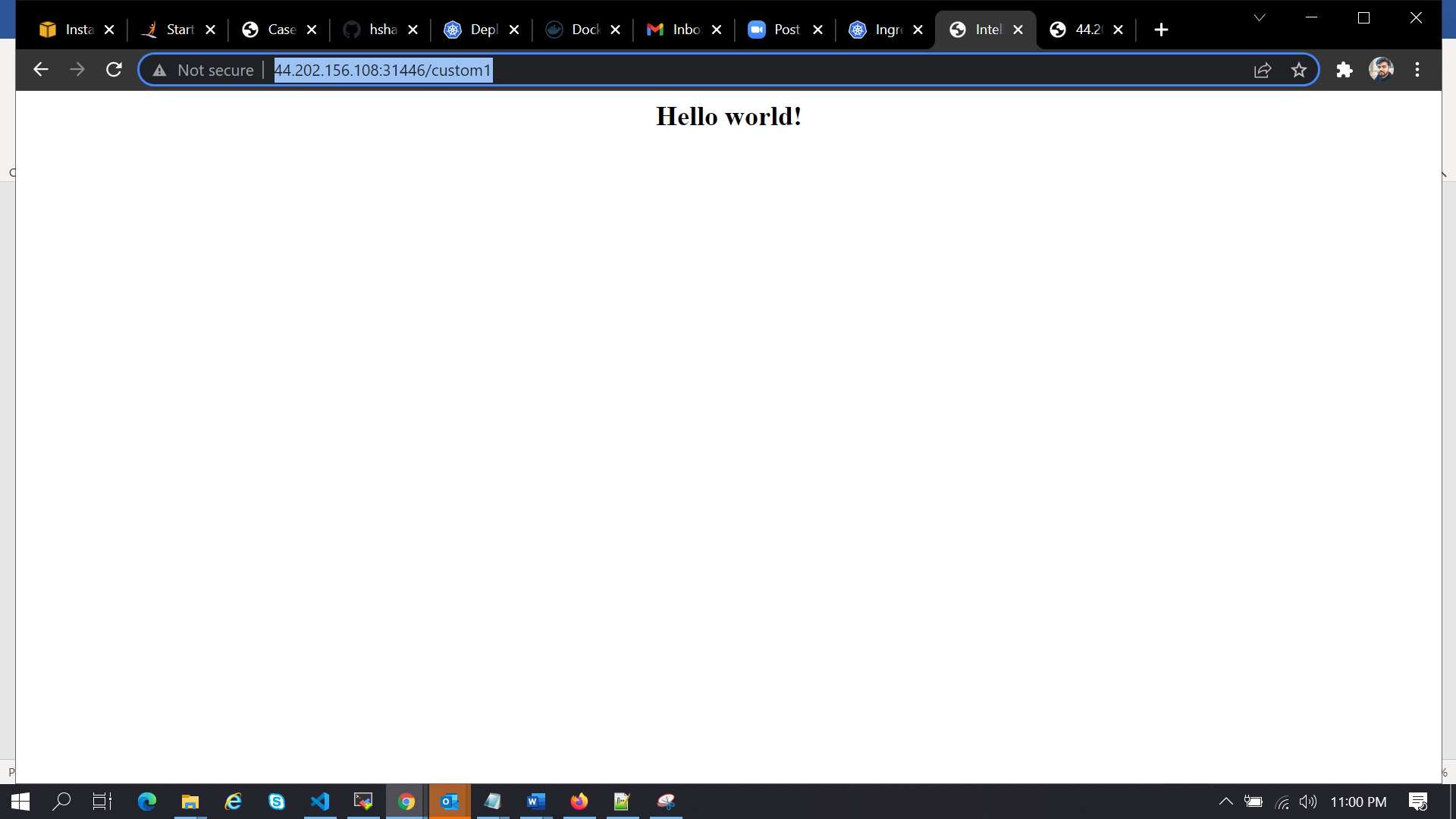


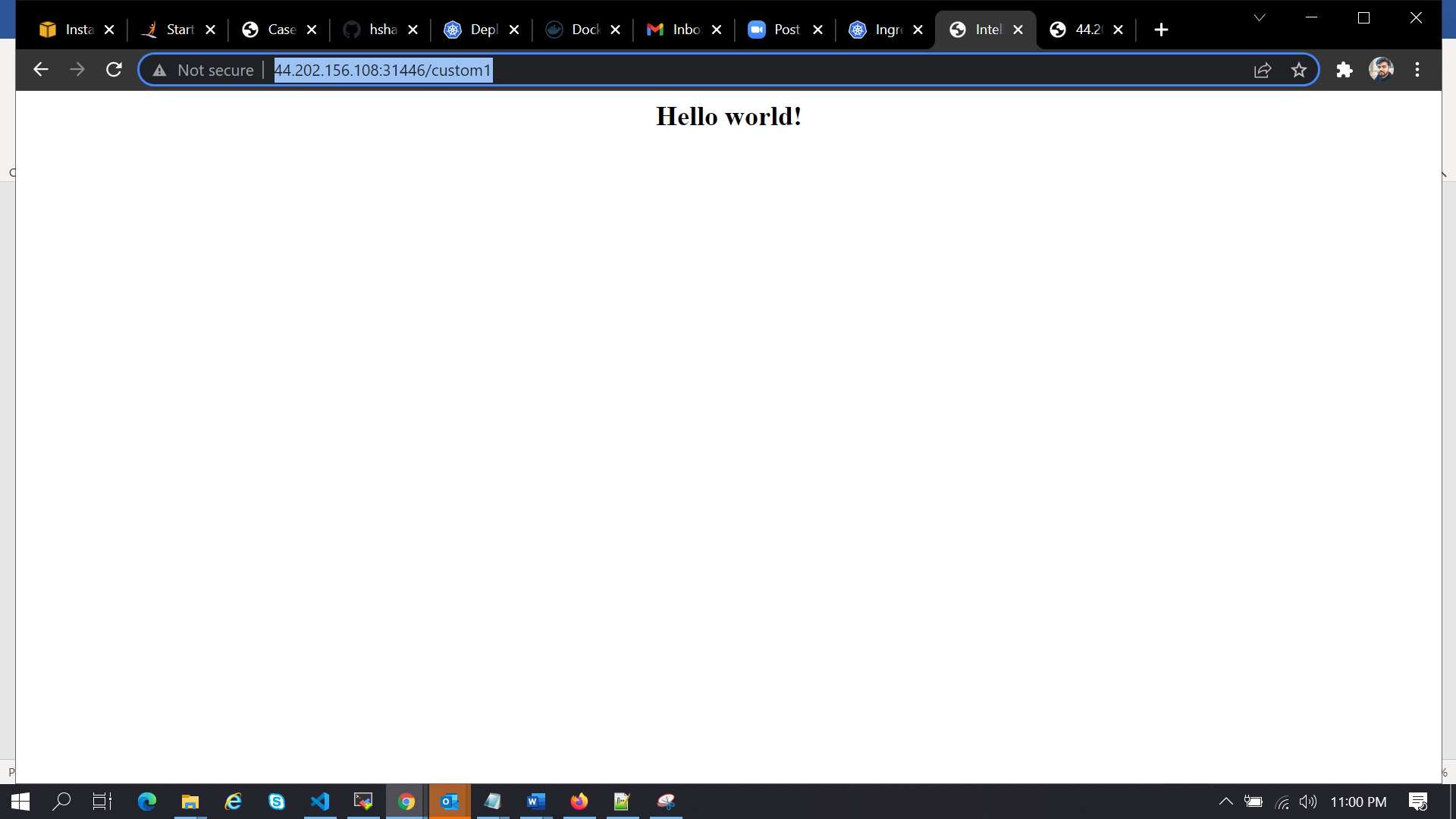
Output after Ingress deployment:











We can see the different output as per the ingress function .