

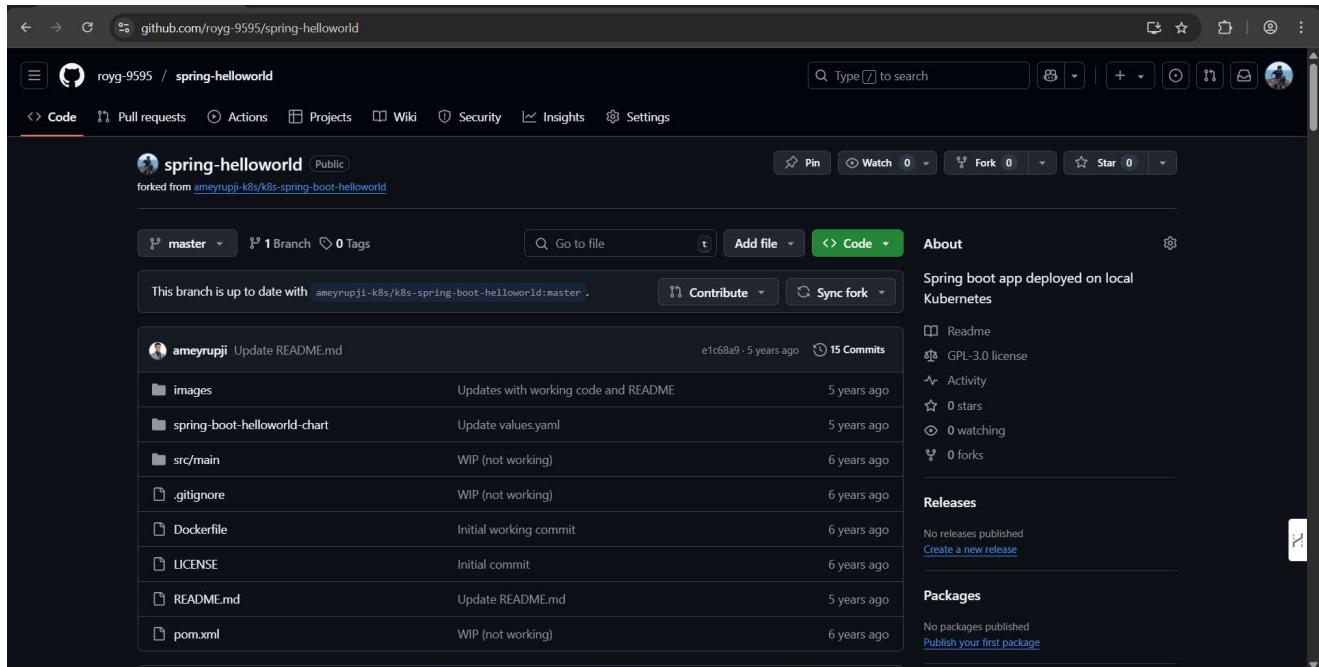
# Jenkins-Docker-Kubernetes Assessment

## Task: 1 Fork a Java Repository

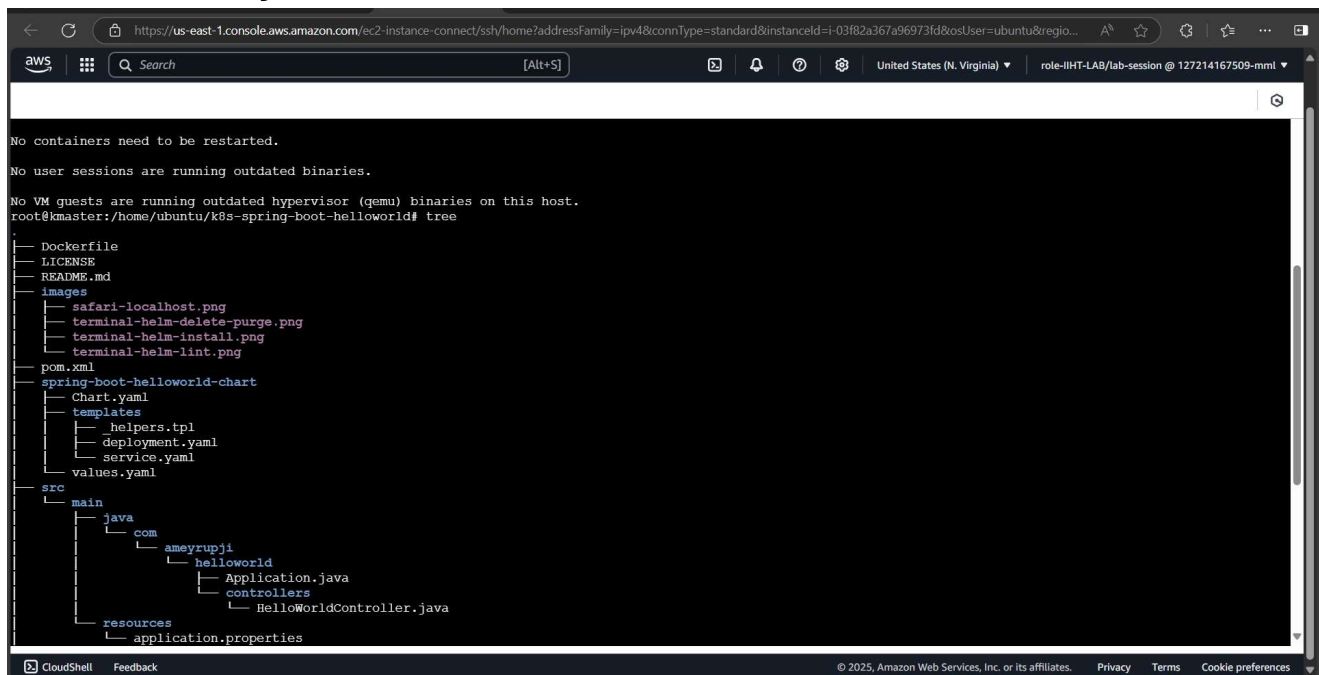
I have forked a GitHub repo from the source <https://github.com/ameyrupji-k8s/k8s-spring-boot-helloworld> to my GitHub account.

Following is my GitHub repo url:

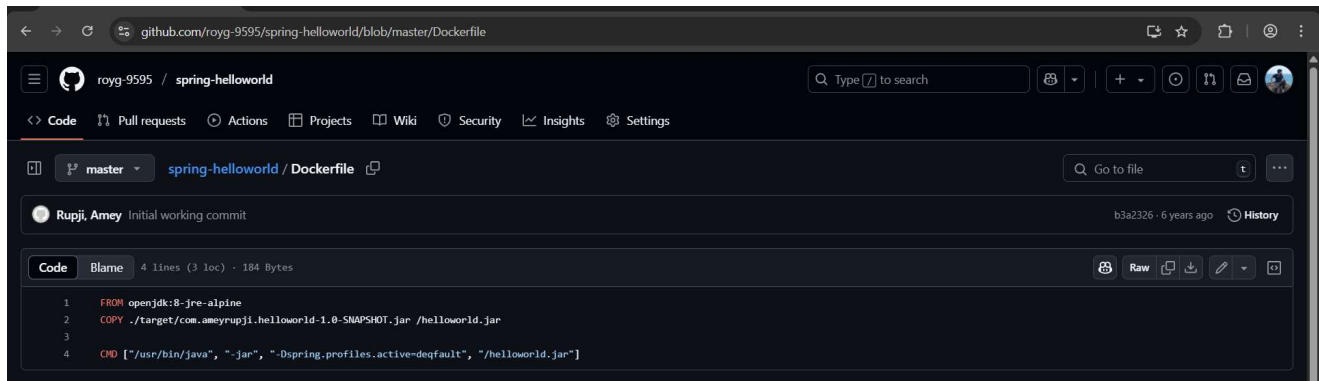
<https://github.com/royg-9595/spring-helloworld>



## Cloned it into my EC2 machine



## Task 2: Define a Dockerfile



The screenshot shows a GitHub repository for 'royg-9595 / spring-helloworld'. The file 'Dockerfile' is selected, showing its content. The Dockerfile has 4 lines of code: 1. FROM openjdk:8-jre-alpine, 2. COPY ./target/com.ameyrupji.helloworld-1.0-SNAPSHOT.jar /helloworld.jar, 3. (empty line), 4. CMD ["usr/bin/java", "-jar", "-Dspring.profiles.active=deqfault", "/helloworld.jar"].

```
1 FROM openjdk:8-jre-alpine
2 COPY ./target/com.ameyrupji.helloworld-1.0-SNAPSHOT.jar /helloworld.jar
3
4 CMD ["usr/bin/java", "-jar", "-Dspring.profiles.active=deqfault", "/helloworld.jar"]
```

Docker File:

FROM openjdk:8-jre-alpine

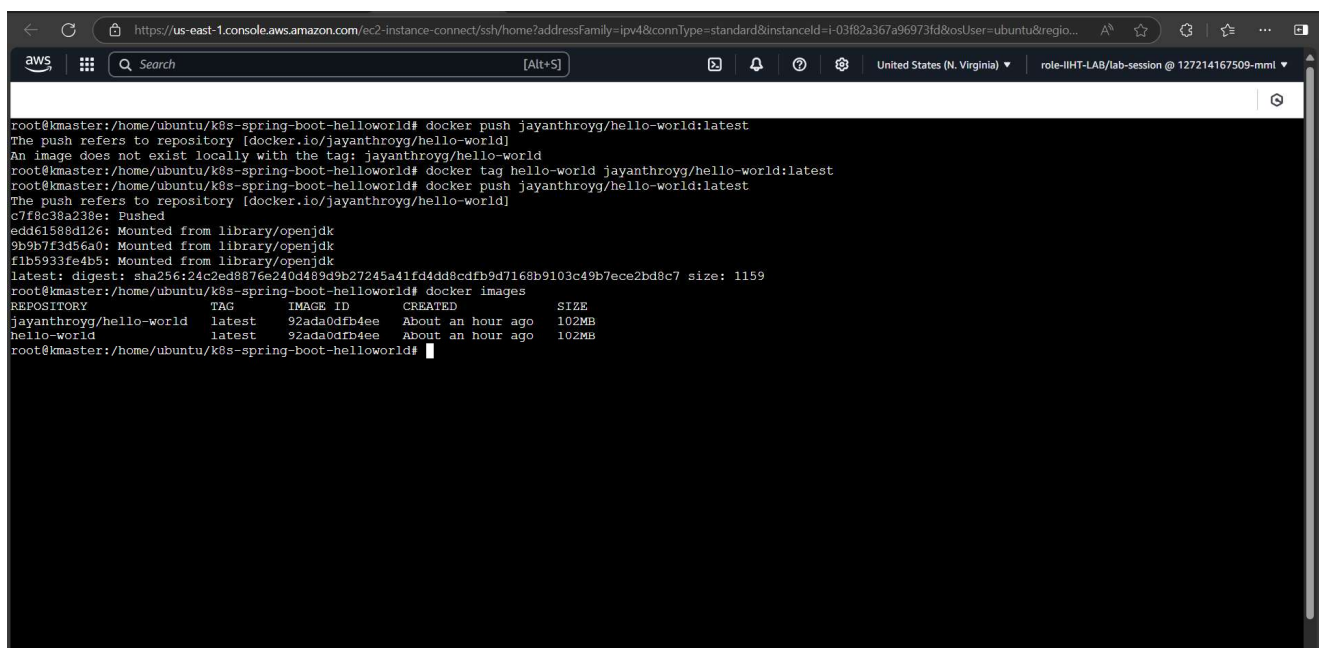
COPY ./target/com.ameyrupji.helloworld-1.0-SNAPSHOT.jar /helloworld.jar

CMD ["usr/bin/java", "-jar", "-Dspring.profiles.active=deqfault", "/helloworld.jar"]

## Task 3 : Build and Push Docker Image

Installed Docker in my EC2 instance

And now I have created the image using Dockerfile above



The screenshot shows a terminal window on an AWS EC2 instance. The user runs the following commands: 1. docker push jayanthroyg/hello-world:latest (fails with 'The push refers to repository [docker.io/jayanthroyg/hello-world]' and 'An image does not exist locally with the tag: jayanthroyg/hello-world'), 2. docker tag hello-world jayanthroyg/hello-world:latest, 3. docker push jayanthroyg/hello-world:latest (succeeds). The output shows the image being pushed to Docker Hub. The user then runs 'docker images' and the output shows the image 'jayanthroyg/hello-world:latest' with ID '92ada0dfb4ee' and size '102MB'.

```
root@kmaster:/home/ubuntu/k8s-spring-boot-helloworld# docker push jayanthroyg/hello-world:latest
The push refers to repository [docker.io/jayanthroyg/hello-world]
An image does not exist locally with the tag: jayanthroyg/hello-world
root@kmaster:/home/ubuntu/k8s-spring-boot-helloworld# docker tag hello-world jayanthroyg/hello-world:latest
root@kmaster:/home/ubuntu/k8s-spring-boot-helloworld# docker push jayanthroyg/hello-world:latest
The push refers to repository [docker.io/jayanthroyg/hello-world]
c7f9c38a238e: Pushed
edd61588d126: Mounted from library/openjdk
9b9b7f3d56a0: Mounted from library/openjdk
f1b5933fe4b5: Mounted from library/openjdk
latest: digest: sha256:24c2ed8876e240d489d9b27245a41fd4dd8cdfb9d7168b9103c49b7ece2bd8c7 size: 1159
root@kmaster:/home/ubuntu/k8s-spring-boot-helloworld# docker images
REPOSITORY          TAG          IMAGE ID      CREATED      SIZE
jayanthroyg/hello-world latest      92ada0dfb4ee  About an hour ago  102MB
hello-world         latest      92ada0dfb4ee  About an hour ago  102MB
root@kmaster:/home/ubuntu/k8s-spring-boot-helloworld#
```

Pushed the docker image to my docker hub and following is the link

<https://hub.docker.com/repository/docker/jayanthroyg/hello-world>

## Task 4: Create a Kubernetes Deployment

Deploy.yaml

apiVersion: apps/v1

kind: Deployment

metadata:

annotations:

deployment.kubernetes.io/revision: "1"

creationTimestamp: "2025-03-29T07:08:42Z"

generation: 1

labels:

app: hello-world-deployment

name: hello-world-deployment

namespace: default

resourceVersion: "10340"

uid: 27b8513a-60a7-4c0f-8891-2d25b5305059

spec:

progressDeadlineSeconds: 600

replicas: 1

revisionHistoryLimit: 10

selector:

matchLabels:

app: hello-world-deployment

strategy:

rollingUpdate:

maxSurge: 25%

maxUnavailable: 25%

type: RollingUpdate

template:

metadata:

creationTimestamp: null

labels:

app: hello-world-deployment

spec:

containers:

- image: jayanthroyg/hello-world

imagePullPolicy: Always

name: hello-world

ports:

- containerPort: 9091 # Update the containerPort to 9091

protocol: TCP

resources: {}

terminationMessagePath: /dev/termination-log

terminationMessagePolicy: File

dnsPolicy: ClusterFirst

restartPolicy: Always

schedulerName: default-scheduler

securityContext: {}

terminationGracePeriodSeconds: 30

status:

availableReplicas: 1

conditions:

- lastTransitionTime: "2025-03-29T07:08:47Z"

lastUpdateTime: "2025-03-29T07:08:47Z"

message: Deployment has minimum availability.

reason: MinimumReplicasAvailable

status: "True"

type: Available

- lastTransitionTime: "2025-03-29T07:08:42Z"

lastUpdateTime: "2025-03-29T07:08:47Z"

message: ReplicaSet "hello-world-deployment-5cd6fb998d" has successfully progressed.

reason: NewReplicaSetAvailable

status: "True"

type: Progressing

observedGeneration: 1

readyReplicas: 1

replicas: 1

updatedReplicas: 1

```
root@kmaster:/home/ubuntu# kubectl get all
NAME                                READY   STATUS    RESTARTS   AGE
pod/hello-world-deployment-7c44dd8f9f-jgvf7   1/1     Running   0          4m28s

NAME                                TYPE               CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
service/hello-world-service          NodePort           10.103.207.175  <none>            9091:30007/TCP   2m20s
service/kubernetes                   ClusterIP           10.96.0.1       <none>            443/TCP          154m

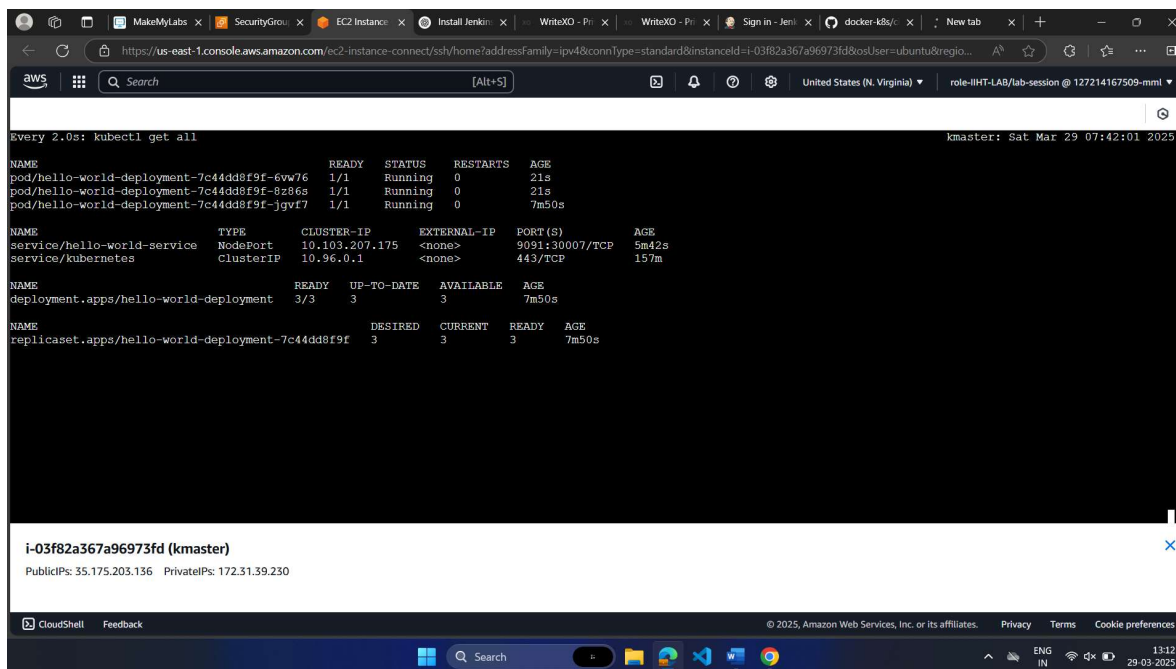
NAME                                READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/hello-world-deployment  1/1     1            1          4m28s

NAME                                DESIRED   CURRENT   READY   AGE
replicaset.apps/hello-world-deployment-7c44dd8f9f  1         1         1       4m28s
root@kmaster:/home/ubuntu#
```

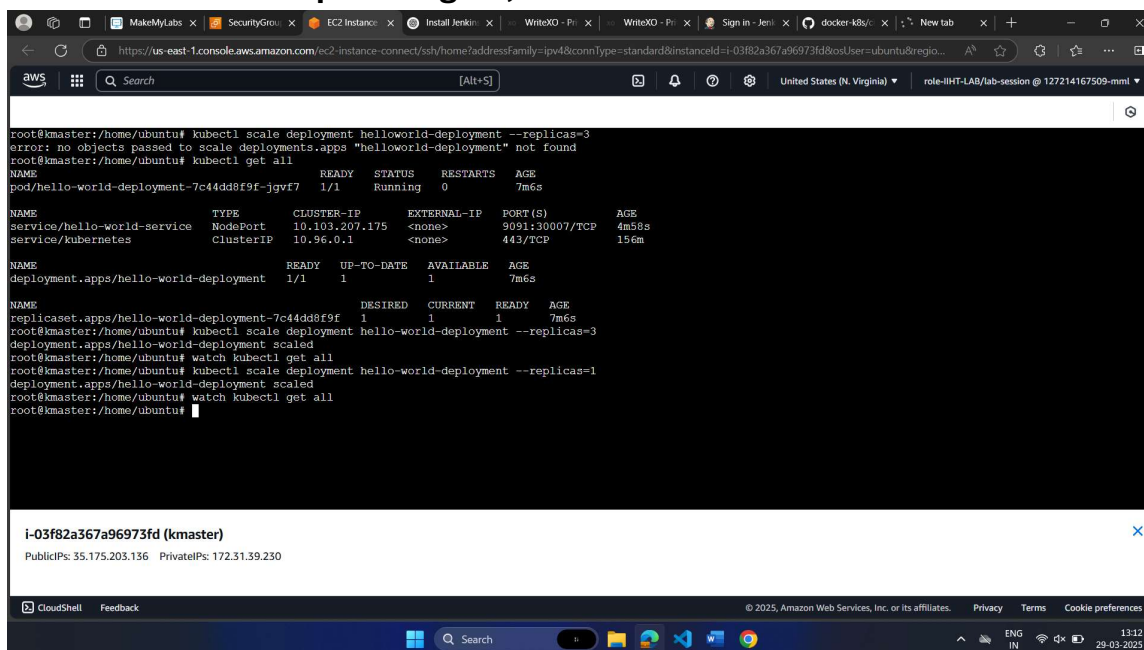
i-03f82a367a96973fd (kmaster)  
PublicIPs: 35.175.203.136 PrivateIPs: 172.31.39.230

Scaled the application replicas to 3

Below is the screenshot



## Scaled down to 1 replicas again, below is the screenshot



## Task 6: Expose the Service Using NodePort

apiVersion: v1

kind: Service

metadata:

name: hello-world-service

namespace: default

spec:

type: NodePort

selector:

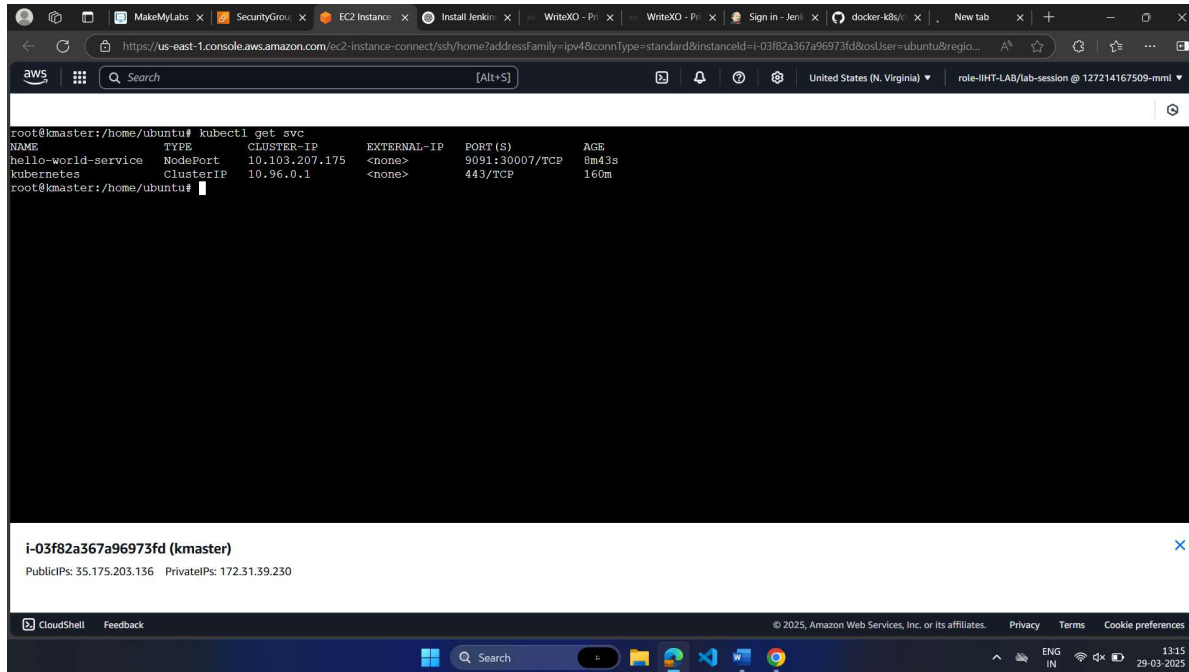
app: hello-world-deployment # This matches the label in your deployment YAML  
ports:

- protocol: TCP

port: 9091 # The internal port the service will listen on (match containerPort)

targetPort: 9091 # The target port in the container

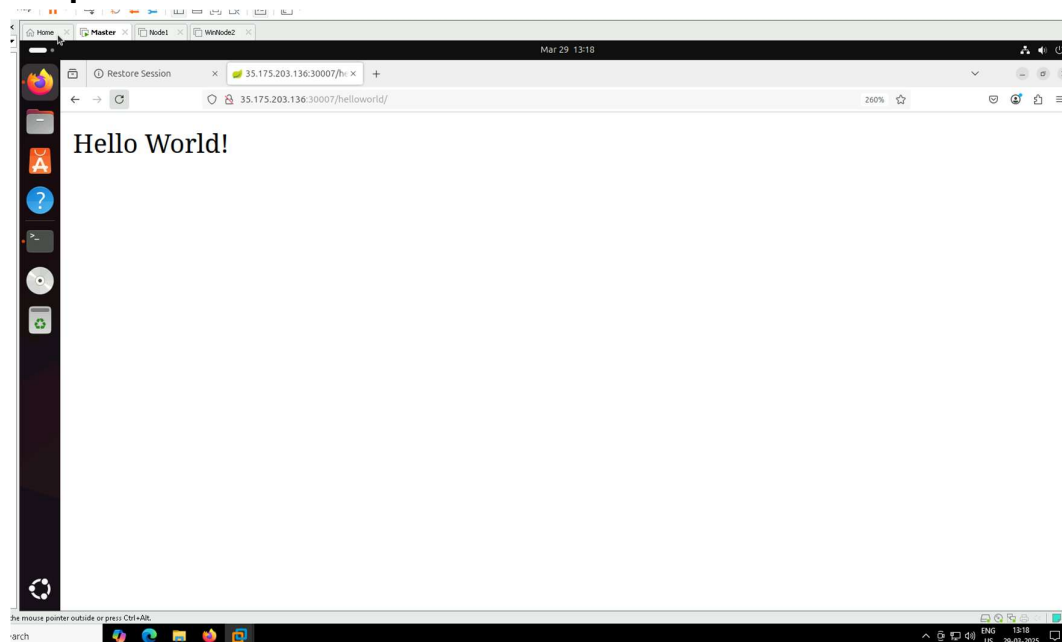
nodePort: 30007 # The external port on which the service will be exposed



<http://35.175.203.136:30007/helloworld/>

The above is the application which returns Helloworld! output

**Output**



## Task 7: Automate Deployment Using Jenkins

Dashboard > java-hello-world >

Status

Changes

Workspace

Build Now

Configure

Delete Project

Rename

Builds

Filter

Today

#2 5:40 AM

java-hello-world

Add description

Permalinks

- Last build (#2), 2 hr 2 min ago
- Last stable build (#2), 2 hr 2 min ago
- Last successful build (#2), 2 hr 2 min ago
- Last failed build (#1), 2 hr 3 min ago
- Last unsuccessful build (#1), 2 hr 3 min ago
- Last completed build (#2), 2 hr 2 min ago

18.232.83.123/job/java-hello-world/2/console

Update

Dashboard > java-hello-world > #2 > Console Output

```
+ docker push vinayz7/hello-world-java-docker
Using default tag: latest
The push refers to repository [docker.io/vinayz7/hello-world-java-docker]
b111145cb0d4: Preparing
2b6a5d9b3ab0: Preparing
889a085eab4b: Preparing
cb36459aaa32: Preparing
f1ccf0afcb5e: Preparing
235cb1df51fd: Preparing
235cb1df51fd: Waiting
f1ccf0afcb5e: Pushed
889a085eab4b: Pushed
b111145cb0d4: Pushed
2b6a5d9b3ab0: Pushed
235cb1df51fd: Pushed
cb36459aaa32: Pushed
latest: digest: sha256:ccb59f1f1b38434a61be53cd1cc07032302ab798dd811d147c704c4b941ff6b6 size: 1575
Finished: SUCCESS
```

<> Code Pull requests Actions Projects Wiki Security Insights Settings

Okay, that hook was successfully created. We sent a ping payload to test it out! Read more about it at <https://docs.github.com/webhooks/#ping-event>.

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Rules

Actions

Webhooks

Webhooks

Add webhook

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

✓ http://18.232.83.123/github-webho... (push)

Edit Delete

Last delivery was successful.