

Application Deployment with Docker, Kubernetes, and CI/CD

Github repo: <https://github.com/asnashameel/spring-boot-hello-world>

DockerHub repo: <https://hub.docker.com/repository/docker/asnashameel/spring-boot-hello-world/>

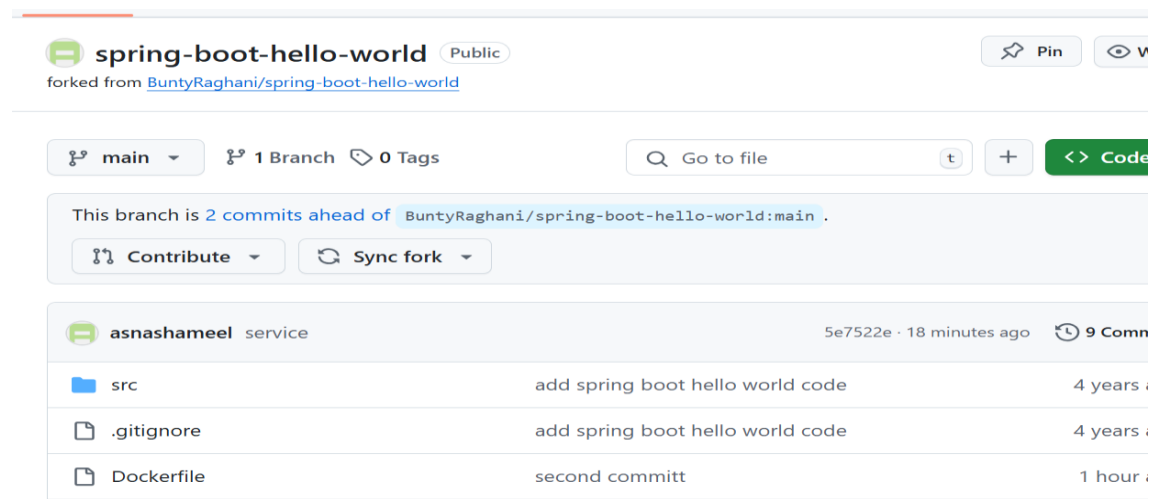
Service.yaml – <https://github.com/asnashameel/spring-boot-hello-world/blob/main/service.yaml>

deploy.yaml - <https://github.com/asnashameel/spring-boot-hello-world/blob/main/deploy.yaml>

Task 1

1) Fork and Clone java repository

Forked spring-boot-hello-world repository and cloned it



2) create a Dockerfile

```
FROM maven:3.8.6-eclipse-temurin-17 AS build
WORKDIR /app
COPY . .

RUN mvn clean package -DskipTests

FROM eclipse-temurin:17-jdk
WORKDIR /app
COPY --from=build /app/target/*.jar app.jar
EXPOSE 8080

ENTRYPOINT ["java", "-jar", "app.jar"]
```

3) Build docker image

```

=> => extracting sha256:0b4fff2c206cae13745f200704b38134105ac3a4bcf862dc0d04d08f6aa9d2ce 0.0s
=> => extracting sha256:3d2700d3f64bcfb1b67ccf65a715330aa34cb16c0c1e8c9146c15011233d6fc4 0.0s
=> [internal] load build context 0.1s
=> => transferring context: 46.30kB 0.1s
=> [build 1/4] FROM docker.io/library/maven:3.8.6-eclipse-temurin-17@sha256:5092873778f0495464c1151df8f5c2e01a09ba37d931be71 92.1s
=> => resolve docker.io/library/maven:3.8.6-eclipse-temurin-17@sha256:5092873778f0495464c1151df8f5c2e01a09ba37d931be71 0.0s
=> => sha256:6e3729cf69e0ce2de9e779575a1fec8b7fb5efdfa822829290ab6d5d1bc3e797 30.43MB / 30.43MB 1.7s
=> => sha256:0f909120a5780e04d8cb0aa2f4bd04de2579ff60528719fffb09e3611423534fb 8.48kB / 8.48kB 0.0s
=> => sha256:d9035c93956cdaf1067a6fe5d120c6c3575830dd651802e72c8e9376b78563b9 2.00kB / 2.00kB 0.0s
=> => sha256:4d9d923227d8dc300e1ddab1b65c83c0eff7f4a7105d958420872f7138b79735 16.97MB / 16.97MB 3.1s
=> => sha256:eda8241fd25fe52c584da502f21cef5b5067551f7bcfcca58e9a630238f7f558 192.44MB / 192.44MB 22.1s
=> => sha256:5092873778f0495464c1151df8f5c2e01a09ba37d931be719c9c1fc9f4559a07 1.21kB / 1.21kB 0.0s
=> => sha256:35dccabde73de58115ed8db691bd335b82d91b6961c0d1af77691b368b0a8253 175B / 175B 2.3s
=> => extracting sha256:6e3729cf69e0ce2de9e779575a1fec8b7fb5efdfa822829290ab6d5d1bc3e797 10.9s
=> => sha256:d7207820eacfa26927d03b44854489e8e2a9052aacb83a3bd19d6ab059e3dbad 21.82MB / 21.82MB 5.7s
=> => sha256:ed84b45b63bdf4e0e24ee349ecca664d2b61ef0c1a6812b7a7dab051475eff6a 8.74MB / 8.74MB 5.8s
=> => sha256:4097965eadc4b93852c18df33b24f9917da48c19b9ebd19498e69c6b5af7cf6e 852B / 852B 6.2s
=> => sha256:d713a6b46ea0e75b76bbc3073acbf8c82526abdc3f09109a5642320742681e9 361B / 361B 6.7s
=> => extracting sha256:4d8d923227d8dc300e1ddab1b65c83c0eff7f4a7105d958420872f7138b79735 9.8s
=> => extracting sha256:eda8241fd25fe52c584da502f21cef5b5067551f7bcfcca58e9a630238f7f558 41.7s
=> => extracting sha256:35dccabde73de58115ed8db691bd335b82d91b6961c0d1af77691b368b0a8253 0.1s
=> => extracting sha256:d7207820eacfa26927d03b44854489e8e2a9052aacb83a3bd19d6ab059e3dbad 24.1s
=> => extracting sha256:ed84b45b63bdf4e0e24ee349ecca664d2b61ef0c1a6812b7a7dab051475eff6a 0.3s
=> => extracting sha256:4097965eadc4b93852c18df33b24f9917da48c19b9ebd19498e69c6b5af7cf6e 0.0s
=> => extracting sha256:d713a6b46ea0e75b76bbc3073acbf8c82526abdc3f09109a5642320742681e9 0.0s
=> [stage-1 2/3] WORKDIR /app 3.8s
=> [build 2/4] WORKDIR /app 0.3s
=> [build 3/4] COPY . . 0.2s
=> [build 4/4] RUN mvn clean package -DskipTests 345.0s
=> [stage-1 3/3] COPY --from=build /app/target/*.jar app.jar 0.2s
=> exporting to image 0.2s
=> => exporting layers 0.2s
=> => writing image sha256:6bd24f8ce38e0d4bf8862fc83a9bf277375a892efa63de9363841b25f0e73f4c 0.0s
=> => naming to docker.io/library/spring-boot-hello-world 0.0s

```

Check container is running :

```

47e35b8d4202812a5f74c8d81cc47c75bef25e273316ebd77c57a3dd9477346c
root@kmaster:~/spring-boot-hello-world# curl http://localhost:8081/hello
Hello, World!root@kmaster:~/spring-boot-hello-world#

```

4)push docker image

```

The push refers to repository [docker.io/asnashameel/spring-boot-hello-world]
d770b9b8f84e: Pushed
f744257d467a: Pushed
c25422af2aa3: Mounted from library/eclipse-temurin
d69a85e978ba: Mounted from library/eclipse-temurin
e11ffcce38cd: Mounted from library/eclipse-temurin
d79a1d8f9a9b: Mounted from library/eclipse-temurin
4b7c01ed0534: Mounted from library/eclipse-temurin
latest: digest: sha256:d14ab111946df8656d069db4a0dc51cd0c610ea4620dd3d4840b4aac5b36eea4 size: 1787
root@kmaster:~/spring-boot-hello-world#

```

asnashameel

Docker Personal

Repositories

Settings

Default privacy

Notifications

Billing

Usage

Repositories

All repositories within the **asnashameel** namespace.

All content

Create a repository

Name	Last Pushed	Contains	Visibility	Scout
asnashameel/spring-boot-hello-world	1 minute ago	IMAGE	Public	Inactive

Task 2

1)creates kubernetes deployment

deployment.yaml

```

GNU nano 4.8
apiVersion: apps/v1
kind: Deployment
metadata:
  creationTimestamp: null
  labels:
    app: spring-boot-app
  name: spring-boot-app
spec:
  replicas: 1
  selector:
    matchLabels:
      app: spring-boot-app
  strategy: {}
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: spring-boot-app
    spec:
      containers:
      - image: ashashameel/spring-boot-hello-world:latest
        name: spring-boot-hello-world
        resources: {}
status: {}

```

Kubectl get deploy,po

```

Normal Created 20s kubelet Created container spring-boot-hello-world
Normal Started 19s kubelet Started container spring-boot-hello-world
root@kmaster:~/spring-boot-hello-world# kubectl get po
NAME                                READY   STATUS    RESTARTS   AGE
mypod                               1/1     Running   0           138m
spring-boot-app-6bb67d5cb9-kqtvb    1/1     Running   0           64s
root@kmaster:~/spring-boot-hello-world# kubectl get deploy
NAME    READY   UP-TO-DATE   AVAILABLE   AGE
spring-boot-app  1/1     1             1           91s
root@kmaster:~/spring-boot-hello-world#

```

2)scale up and scale down deployment

```

root@kmaster:~/spring-boot-hello-world# kubectl scale deployment spring-boot-app --replicas=3
deployment.apps/spring-boot-app scaled
root@kmaster:~/spring-boot-hello-world# kubectl get deploy,po
NAME                                READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/spring-boot-app    3/3     3             3           6m33s

NAME                                READY   STATUS    RESTARTS   AGE
pod/mypod                           1/1     Running   0           143m
pod/spring-boot-app-6bb67d5cb9-258l6 1/1     Running   0           8s
pod/spring-boot-app-6bb67d5cb9-d9z7j 1/1     Running   0           9s
pod/spring-boot-app-6bb67d5cb9-kqtvb 1/1     Running   0           6m33s
root@kmaster:~/spring-boot-hello-world# kubectl scale deployment spring-boot-app --replicas=2
deployment.apps/spring-boot-app scaled
root@kmaster:~/spring-boot-hello-world# kubectl get deploy,po
NAME                                READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/spring-boot-app    2/2     2             2           7m32s

NAME                                READY   STATUS    RESTARTS   AGE
pod/mypod                           1/1     Running   0           144m
pod/spring-boot-app-6bb67d5cb9-d9z7j 1/1     Running   0           68s
pod/spring-boot-app-6bb67d5cb9-kqtvb 1/1     Running   0           7m32s

```

Task 3

1)Expose service using Nodeport and access the application

```

root@kmaster:~/spring-boot-hello-world# kubectl get svc
NAME            TYPE          CLUSTER-IP      EXTERNAL-IP  PORT(S)          AGE
kubernetes      ClusterIP     10.96.0.1       <none>       443/TCP          3h52m
spring-boot-app NodePort      10.107.67.160   <none>       8080:32680/TCP   80m
root@kmaster:~/spring-boot-hello-world# curl localhost:32680/hello
Hello, World!root@kmaster:~/spring-boot-hello-world#

root@kmaster:~/spring-boot-hello-world# curl http://192.168.38.128:32680/hello
Hello, World!root@kmaster:~/spring-boot-hello-world#

```

Service.yaml

```
apiVersion: v1
kind: Service
metadata:
  creationTimestamp: "2025-04-03T09:49:34Z"
  labels:
    app: spring-boot-app
    name: spring-boot-app
    namespace: default
  resourceVersion: "13179"
  uid: e895a061-ba06-41b0-a2f8-1085ca518799
spec:
  clusterIP: 10.107.67.160
  clusterIPs:
    - 10.107.67.160
  externalTrafficPolicy: Cluster
  internalTrafficPolicy: Cluster
  ipFamilies:
    - IPv4
  ipFamilyPolicy: SingleStack
  ports:
    - nodePort: 32680
      port: 8080
      protocol: TCP
      targetPort: 8080
  selector:
    app: spring-boot-app
  sessionAffinity: None
```

Read 34 lines

Activate Windows

Task -4

Installed Jenkins and created freestyle job

```
#!/bin/bash
sudo apt update && sudo apt install mvn -y
DOCKER_IMAGE="asnashameel/spring-boot-hello-world:latest"
docker build -t $DOCKER_IMAGE .
echo "Logging into Docker Hub..."
docker login -u "asnashameel" -p "Asna@2001"
docker push $DOCKER_IMAGE
kubectl apply -f deploy.yaml
kubectl apply -f service.yaml
kubectl get pods
kubectl get svc
```

Console Output

Download

Copy

View as plain text

```
Started by user admin
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/maven
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/maven/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/asnashameel/spring-boot-hello-world #
timeout=10
Fetching upstream changes from https://github.com/asnashameel/spring-boot-hello-world
> git --version # timeout=10
> git --version # 'git version 2.25.1'
> git fetch --tags --force --progress -- https://github.com/asnashameel/spring-boot-hello-
world +refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision bfa8a88ec6aab1f43333fd2c9587e584f87a1e2 (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
```

```
Login Succeeded
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/
docker.sock: Post "http://%2Fvar%2Frun%2Fdocker.sock/v1.48/images/asnashameel/spring-boot-
hello-world/push?tag=latest": dial unix /var/run/docker.sock: connect: permission denied
Error from server (Forbidden): <html><head><meta http-equiv='refresh' content='1;url=/login?
from=%2Fswagger-2.0.0.pb-v1%3Ftimeout%3D32s' /><script id='redirect' data-redirect-url='/
login?from=%2Fswagger-2.0.0.pb-v1%3Ftimeout%3D32s' src='/static/f99721da/scripts/
redirect.js'></script></head><body style='background-color:white; color:white;'>
Authentication required
<!--
-->

</body></html>
Deployment Successful!
Finished: SUCCESS
```