

Fork the code below

<https://github.com/devops-experience/k8s-helloworld>

Github repo link: <https://github.com/vinayz7/hello-world-java-docker>

devops-experience / k8s-helloworld

Create a new fork

A fork is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

Required fields are marked with an asterisk (*).

Owner * Repository name *

vinayz7 / k8s-helloworld

k8s-helloworld is available.

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

A simple Python Helloworld application designed to demonstrate Kubernetes features.

☒ Copy the master branch only

Contribute back to devops-experience/k8s-helloworld by adding your own branch. [Learn more.](#)

Build a Dockerfile for a simple Python Flask app

```
root@kmaster:/home/ubuntu/k8s-helloworld# docker build -t helloworld .
[+] Building 5.3s (10/10) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 419B
=> [internal] load metadata for docker.io/library/python:3.9-slim
=> [auth] library/python:pull token for registry-1.docker.io
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [1/4] FROM docker.io/library/python:3.9-slim@sha256:40007fe18a72a2e7166be350d52dab86b9fe18f2de08e6a38e26422fb247e81e
=> [internal] load build context
=> => transferring context: 38.45kB
=> CACHED [2/4] WORKDIR /app
=> [3/4] COPY . .
=> [4/4] RUN pip install --no-cache-dir -r requirements.txt
=> exporting to image
=> => exporting layers
=> => writing image sha256:1f05d93baa49b40bfabc6d473349af0a5de8383b9e2fc9b1e6b6bf31ccb526b4
=> => naming to docker.io/library/helloworld
root@kmaster:/home/ubuntu/k8s-helloworld# docker images
REPOSITORY          TAG         IMAGE ID      CREATED        SIZE
helloworld          latest     1f05d93baa49  18 seconds ago 136MB
vinayz7/hello1springboot v1         b902924f72f4  5 hours ago   428MB
vinayz7/my-flask-app latest     ea9295df8dc0  5 hours ago   139MB
root@kmaster:/home/ubuntu/k8s-helloworld#
```

- After running the container, display:

```
root@kmaster:/home/ubuntu/k8s-helloworld# docker run -d --name flask-container helloworld
dc9dfca710d35f9f3b54987d8ba7ba64b87ca5d826a051873a94b15f0682da89
root@kmaster:/home/ubuntu/k8s-helloworld# docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
dc9dfca710d3   helloworld    "python app.py"         7 seconds ago Up 6 seconds  5000/tcp       flask-container
```

- The container ID

```
root@kmaster:/home/ubuntu/k8s-helloworld# docker ps -aqf "name=flask-container"
dc9dfca710d3
```

- The working directory inside the container

```
root@kmaster:/home/ubuntu/k8s-helloworld# docker exec flask-container pwd
/app
```

- The read-write layer directory on the host

```
root@kmaster:/home/ubuntu/k8s-helloworld# docker inspect flask-container --format='{{.GraphDriver.Data.UpperDir}}'
/var/lib/docker/overlay2/71e76bece42d23bde29a0eaf7364e45e107471f5378a12185f6f1935ff79d46a/diff
root@kmaster:/home/ubuntu/k8s-helloworld#
```