

What is Kubernetes?

Kubernetes (often abbreviated as **K8s**) is an **orchestration tool** that helps manage **containers** automatically. Think of it as a **traffic controller** for applications running in containers, making sure they are running **smoothly, efficiently, and reliably**.

Why Do We Need Kubernetes?

Imagine you have an app that runs inside **containers** (small, portable environments that include everything needed to run software). Managing one or two containers manually is easy, but what if your app needs **hundreds or thousands** of containers across multiple machines? That's where Kubernetes helps!

Kubernetes **automates**:

- ✓ **Deployment** – Launching new containers automatically.
 - ✓ **Scaling** – Increasing or decreasing the number of running containers based on demand.
 - ✓ **Load Balancing** – Distributing traffic evenly so no single part gets overloaded.
 - ✓ **Self-Healing** – Restarting failed containers automatically.
 - ✓ **Resource Management** – Optimizing memory and CPU usage.
-

How Does Kubernetes Work?

Kubernetes has different parts that work together:

1. **Cluster** 🏢 – A collection of machines where your app runs.
2. **Nodes** 💻 – Each machine (physical or virtual) in the cluster is a **node**.
3. **Pods** 📦 – The smallest unit in Kubernetes. A **pod** holds one or more containers.
4. **Master Node (Control Plane)** 🧠 – The brain of Kubernetes, responsible for managing everything. It includes:
 - **API Server** – The entry point for commands and communication.
 - **Scheduler** – Decides which node runs which pod.
 - **Controller Manager** – Ensures the system is running as expected.
5. **Worker Nodes** 👤 – The machines where the actual work happens, running the containers inside pods.
6. **Kubelet** 🛡️ – Runs on each worker node and talks to the master node.
7. **Kube-Proxy** 🔄 – Manages networking between pods.

Key Features of Kubernetes

- ✓ **Self-Healing** – If a container crashes, Kubernetes restarts it automatically.
 - ✓ **Auto-Scaling** – Kubernetes can increase or decrease the number of running containers based on traffic.
 - ✓ **Rolling Updates** – Deploy a new version of an app without downtime.
 - ✓ **Service Discovery & Load Balancing** – Manages network traffic between different parts of an application.
-

Real-World Example

Imagine you run an **online shopping website**. During normal days, 10 containers handle traffic. But on **Black Friday**, traffic increases 10x. Instead of manually starting more servers, Kubernetes automatically **scales up** new containers. Once the sale is over, it scales them **down** to save costs.























Why Use Kubernetes?

- 🚀 **Efficient** – Saves time and effort in managing containers.
- 💡 **Smart** – Can heal itself and optimize resources.
- 🔗 **Flexible** – Works with any cloud provider (AWS, Google Cloud, Azure) or even on-premises.
- 🔄 **Continuous Deployment** – Easily update applications without downtime.

Demo

<https://www.youtube.com/watch?v=SzbeDqBSRkc>

Refer this vide for the demo

- ✓  DockerKubernetes-Demo [boot] [devtools]
 - ✓  src/main/java
 - ✓  com.wipro
 - >  DockerKubernetesDemoApplication.java
 - ✓  com.wipro.controller
 - >  MyController.java
 - ✓  src/main/resources
 -  static
 -  templates
 -  application.properties
 - ✓  src/test/java
 -  com.wipro
 - >  JRE System Library [JavaSE-17]
 - >  Maven Dependencies
 - >  src
 -  target
 -  HELP.md
 -  mvnw
 -  mvnw.cmd
 -  pom.xml
- >  DurgaSoftContent
- >  employee-organization [boot] [devtools]

```

1 package com.wipro;
2
3 import org.springframework.boot.SpringApplication;
4
5
6 @SpringBootApplication
7 public class DockerKubernetesDemoApplication {
8
9     public static void main(String[] args) {
10         SpringApplication.run(DockerKubernetesDemoApplication.class, args);
11     }
12 }
13
14
```

```

1 package com.wipro.controller;
2
3 import org.springframework.web.bind.annotation.GetMapping;
4 import org.springframework.web.bind.annotation.RestController;
5
6 @RestController
7 public class MyController {
8
9     @GetMapping("/test-docker")
10     public String getData() {
11         return "in docker tutorial project of play java";
12     }
13 }
14
15
```

Dependencies

ADD DEPENDENCIES... CTRL + B

Spring Web

WEB

Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.



Spring Boot Actuator

OPS

Supports built in (or custom) endpoints that let you monitor and manage your application - such as application health, metrics, sessions, etc.



Spring Boot DevTools

DEVELOPER TOOLS

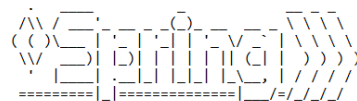
Provides fast application restarts, LiveReload, and configurations for enhanced development experience.



These are the dev tools i have added

Now run the application

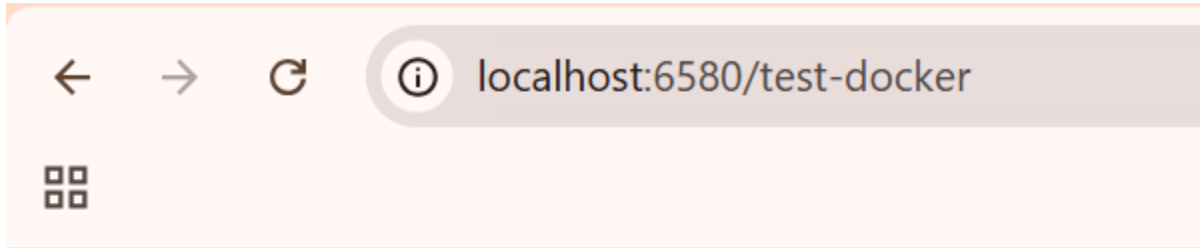
DockerKubernetes-Demo - DockerKubernetesDemoApplication [Spring Boot App] C:\Program Files\Java\jdk-17\bin\javaw.exe (12-Mar-2025, 4:52:16 pm) [pid: 25056]



:: Spring Boot :: (v3.4.3)

```
2025-03-12T16:52:18.095+05:30 INFO 25056 --- [DockerKubernetes-Demo] [ restartedMain] c.wipro.DockerKubernetesDemoApplication : Starting Dock
2025-03-12T16:52:18.097+05:30 INFO 25056 --- [DockerKubernetes-Demo] [ restartedMain] c.wipro.DockerKubernetesDemoApplication : No active pro
2025-03-12T16:52:18.147+05:30 INFO 25056 --- [DockerKubernetes-Demo] [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : Devtools prop
2025-03-12T16:52:18.147+05:30 INFO 25056 --- [DockerKubernetes-Demo] [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : For additiona
2025-03-12T16:52:19.062+05:30 INFO 25056 --- [DockerKubernetes-Demo] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initia
2025-03-12T16:52:19.073+05:30 INFO 25056 --- [DockerKubernetes-Demo] [ restartedMain] o.apache.catalina.core.StandardService : Starting serv
2025-03-12T16:52:19.073+05:30 INFO 25056 --- [DockerKubernetes-Demo] [ restartedMain] o.apache.catalina.core.StandardEngine : Starting Serv
2025-03-12T16:52:19.105+05:30 INFO 25056 --- [DockerKubernetes-Demo] [ restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing
2025-03-12T16:52:19.105+05:30 INFO 25056 --- [DockerKubernetes-Demo] [ restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebAppli
2025-03-12T16:52:19.538+05:30 INFO 25056 --- [DockerKubernetes-Demo] [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : LiveReload se
2025-03-12T16:52:19.543+05:30 INFO 25056 --- [DockerKubernetes-Demo] [ restartedMain] o.s.b.a.e.web.EndpointLinksResolver : Exposing 1 en
2025-03-12T16:52:19.626+05:30 INFO 25056 --- [DockerKubernetes-Demo] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat starte
2025-03-12T16:52:19.652+05:30 INFO 25056 --- [DockerKubernetes-Demo] [ restartedMain] c.wipro.DockerKubernetesDemoApplication : Started Docke
2025-03-12T16:52:20.119+05:30 INFO 25056 --- [DockerKubernetes-Demo] [on(4)-127.0.0.1] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing
2025-03-12T16:52:20.119+05:30 INFO 25056 --- [DockerKubernetes-Demo] [on(4)-127.0.0.1] o.s.web.servlet.DispatcherServlet : Initializing
2025-03-12T16:52:20.120+05:30 INFO 25056 --- [DockerKubernetes-Demo] [on(4)-127.0.0.1] o.s.web.servlet.DispatcherServlet : Completed ini
```

Now check this in the local browser whether it is working or not



in docker tutorial project of play java
























Now stop the server

Now we r going to create the docker file

```
DockerKubernetes-Demo/pom.xml  myController.java  application.pr
1 FROM java:8
2
3 EXPOSE 6580
4
5 ADD target/DockerKubernetes-Demo.jar DockerKubernetes-Demo.jar
6
7 ENTRYPOINT ["java", "-jar", "DockerKubernetes-Demo.jar"]
8
```

Now we need to create the jar file

RightClick>>runas>maven build>clean package>apply

- ✓  DockerKubernetes-Demo [boot] [devtools]
 - ✓  src/main/java
 - ✓  com.wipro
 - >  DockerKubernetesDemoApplication.java
 - ✓  com.wipro.controller
 - >  MyController.java
 - >  src/main/resources
 - ✓  src/test/java
 -  com.wipro
 - >  JRE System Library [JavaSE-17]
 - >  Maven Dependencies
 -  target/generated-sources/annotations
 - >  src
 - ✓  target
 -  generated-sources
 - >  maven-archiver
 - >  maven-status
 -  DockerKubernetes-Demo.jar
 -  DockerKubernetes-Demo.jar.original
 -  Dockerfile
 -  HELP.md
 -  mvnw
 -  mvnw.cmd

target - DockerKubernetes-Demo

Close the server

Now create the repository in the docker hub

dockerhub Explore **Repositories** Organizations Usage

Search Docker Hub

pavanprem / [Repositories](#) / [dockub-demo](#) / [General](#) Using 0 of 1 private repositories

pavanprem/dockub-demo

Created less than a minute ago

Add a description

Add a category

Docker commands [Public view](#)

To push a new tag to this repository:

```
docker push pavanprem/dockub-demo:tagname
```

General Tags Image Management **BETA** Collaborators Webhooks Settings

Now open the cmd and enter “docker login”

```
C:\Users\miniMiracle>docker login
Authenticating with existing credentials...
Login Succeeded

C:\Users\miniMiracle>
```

Now go to the dockerfile destination and open the command prompt

```
C:\Users\miniMiracle\eclipse-workspace\DockeKubernetes-Demo\DockeKubernetes-Demo>docker build -t docker-demo .
[+] Building 43.0s (6/8)
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 415B
=> [internal] load metadata for docker.io/library/openjdk:17
=> [auth] library/openjdk:pull token for registry-1.docker.io
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [1/3] FROM docker.io/library/openjdk:17@sha256:528707081fdb9562eb819128a9f85ae7fe000e2fbaeaf9f87662e7b3f38cb
=> => resolve docker.io/library/openjdk:17@sha256:528707081fdb9562eb819128a9f85ae7fe000e2fbaeaf9f87662e7b3f38cb7
=> => sha256:a7203ca35e75e068651c9907d659adc721dba823441b78639fde66fc988f042f 187.53MB / 187.53MB
=> => sha256:de849f1cfbe60b1c06a1db83a3129ab0ea397c4852b98e3e4300b12ee57ba111 13.53MB / 13.53MB
=> => sha256:38a980f2cc8accf69c23dae6743d42a87eb34a54f02396f3fcfd7c2d06e2c5b 42.11MB / 42.11MB
=> => extracting sha256:38a980f2cc8accf69c23dae6743d42a87eb34a54f02396f3fcfd7c2d06e2c5b
=> => extracting sha256:de849f1cfbe60b1c06a1db83a3129ab0ea397c4852b98e3e4300b12ee57ba111
=> => extracting sha256:a7203ca35e75e068651c9907d659adc721dba823441b78639fde66fc988f042f
=> [internal] load build context
=> => transferring context: 23.36MB
```

Now check whether the docker images are created are not

```
C:\Users\miniMiracle\eclipse-workspace\DockeKubernetes-Demo\DockeKubernetes-Demo>docker images
REPOSITORY          TAG          IMAGE ID          CREATED          SIZE
docker-demo         latest       e508119f50d3      19 seconds ago  771MB
pavanprem/dockerkubernetes-demo latest       598ec528eb1b      28 minutes ago  848MB
pavanprem/myapp      latest       8abbbdf968a15      2 days ago      681MB
springboot-restful-webservices 0.1.RELEASE f71027d27a51      3 days ago      855MB
springboot-docker-demo 0.1.RELEASE e264a36fd66c      6 days ago      778MB
pavanprem/springboot-docker-demo 0.1.RELEASE e264a36fd66c      6 days ago      778MB
openzipkin/zipkin    latest       d9316e7fff757      3 weeks ago     377MB
mysql                latest       146682692a3a      6 weeks ago     1.09GB
mongo                6.0         453e114d98c3      7 weeks ago     1.01GB
rabbitmq             3.13.7-management-alpine d759525efd68      5 months ago    279MB
```


Now push the image to the docker hub

```
C:\Users\miniMiracle\eclipse-workspace\DockerKubernetes-Demo\DockerKubernetes-Demo>docker tag docker-demo pavanprem/dockub-demo

C:\Users\miniMiracle\eclipse-workspace\DockerKubernetes-Demo\DockerKubernetes-Demo>docker push pavanprem/dockub-demo
Using default tag: latest
The push refers to repository [docker.io/pavanprem/dockub-demo]
de849f1cfbe6: Pushing [=====>] 13.53MB/13.53MB
9bcb3b002419: Pushed
b18e50bd113a: Pushed
a15a60594b7e: Pushed
38a980f2cc8a: Pushing [=====>] 19.92MB/42.11MB
a7203ca35e75: Pushing [=====>] 3.146MB/187.5MB
```

Now reload the docker hub..it will show the new images which we pushed just now

pavanprem/dockub-demo ⓘ

Last pushed less than a minute ago

[Add a description](#) ⓘ

[Add a category](#) ⓘ

General | Tags | Image Management BETA | Collaborators | Webhooks | Settings

Docker commands [Public view](#)

To push a new tag to this repository:

```
docker push pavanprem/dockub-demo:tagname
```

Tags

This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
latest		Image	less than 1 day	less than a minute

[See all](#)

Automated builds

Manually pushing images to Docker Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.

Available with Pro, Team and Business subscriptions. [Read more about automated builds](#) ↗.

[Upgrade](#)

```
C:\Users\miniMiracle\eclipse-workspace\DockerKubernetes-Demo\DockerKubernetes-Demo>docker images
REPOSITORY          TAG          IMAGE ID       CREATED        SIZE
docker-demo         latest       e508119f50d3   4 minutes ago  771MB
pavanprem/dockub-demo latest       e508119f50d3   4 minutes ago  771MB
pavanprem/dockerkubernetes-demo latest       598ec528eb1b   32 minutes ago 848MB
pavanprem/myapp      latest       8abbdf968a15   2 days ago    681MB
springboot-restful-webservices 0.1.RELEASE f71027d27a51   3 days ago    855MB
springboot-docker-demo 0.1.RELEASE e264a36fd66c   6 days ago    778MB
pavanprem/springboot-docker-demo 0.1.RELEASE e264a36fd66c   6 days ago    778MB
openzipkin/zipkin    latest       d9316e7ff757   3 weeks ago   377MB
mysql                latest       146682692a3a   6 weeks ago   1.09GB
mongo                6.0         453e114d98c3   7 weeks ago   1.01GB
rabbitmq             3.13.7-management-alpine d759525efd68   5 months ago  279MB
```

Now we can see 2 images ie docker-remo and dockub-demo which is created in hub and other one is local one

Now pull the image from the docker hub before that we need to remove them

Now we will remove the images which we had created

~~C:\H... \... \... \D... \D... \D... \D...~~

```
C:\Users\miniMiracle\eclipse-workspace\DockerKubernetes-Demo\DockerKubernetes-Demo>
```

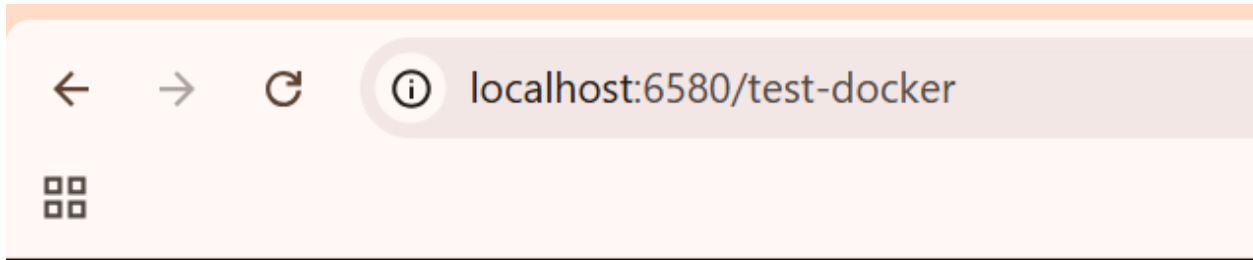
Now we will pull from the docker hub

```

  \  /-----\  /-----\  /-----\  /-----\  /-----\  /-----\
 (  (  \-----\  \-----\  \-----\  \-----\  \-----\  \-----\
  \  /-----\  /-----\  /-----\  /-----\  /-----\  /-----\
  ==-----\  \-----\  \-----\  \-----\  \-----\  \-----\

:: Spring Boot ::                                (v3.4.3)

```



in docker tutorial project of play java

Now open the “google cloud platform” to deploy our service

Now go to the kubernetes tab after successfully created acc and payment method

Watch the video for cluster creation

Filter Enter property name or value								?	
<input type="checkbox"/> Status	Name ↑	Location	Tier ?	Number of nodes	Total vCPUs	Total memory	Notif		
<input type="checkbox"/>	C	kbs-docker	us-central1	Standard	0	0 GB			

Now create the yml file in our eclipse

```

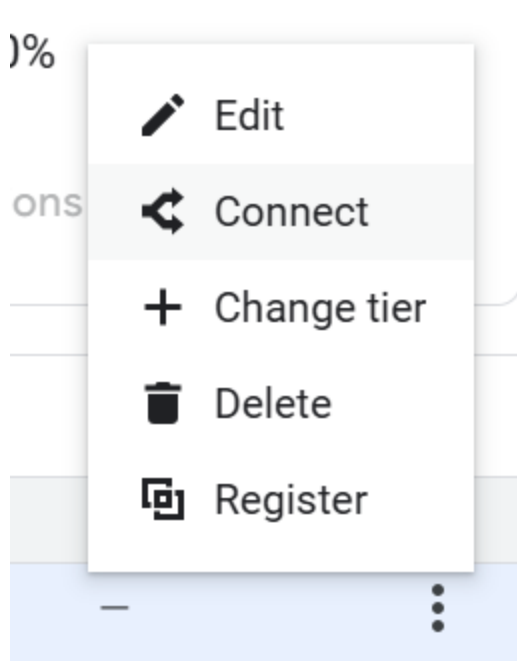
1 apiVersion: apps/v1
2 kind: Deployment
3 metadata:
4   name: docker-k8s-demo-deployment
5   labels:
6     app: docker-k8s-demo
7 spec:
8   replicas: 1
9   selector:
10    matchLabels:
11      app: docker-k8s-demo
12   template:
13     metadata:
14       labels:
15         app: docker-k8s-demo
16     spec:
17       containers:
18         - name: docker-k8s-demo
19           image: pavanprem/dockub-demo
20         ports:
21           - containerPort: 6580
22

```

Filter Enter property name or value							
<input type="checkbox"/>	Status	Name ↑	Location	Tier ?	Number of nodes	Total vCPUs	Total memory
<input type="checkbox"/>	✓	kbs-docker	us-central1	Standard		0	0 GB

Our cluster is created now

Now click on



Connect to the cluster

You can connect to your cluster via command-line or using a dashboard.

Command-line access

Configure [kubectl](#) command line access by running the following command:

```
$ gcloud container clusters get-credentials kbs-docker --region us-central1 --project galvanic-circle-453512-q6
```

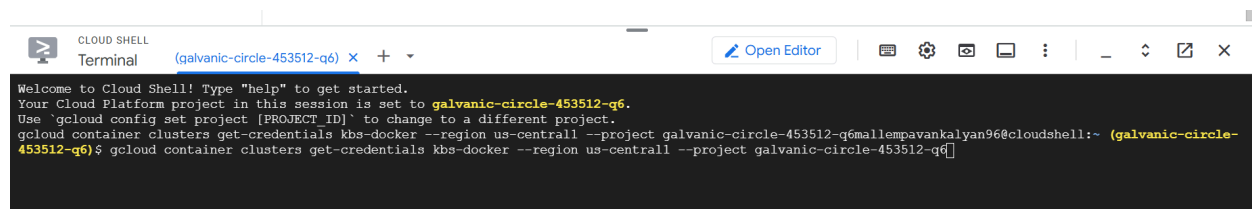
[Run in Cloud Shell](#)

Cloud Console dashboard

You can view the workloads running in your cluster in the Cloud Console [Workloads dashboard](#).

[Open Workloads dashboard](#)

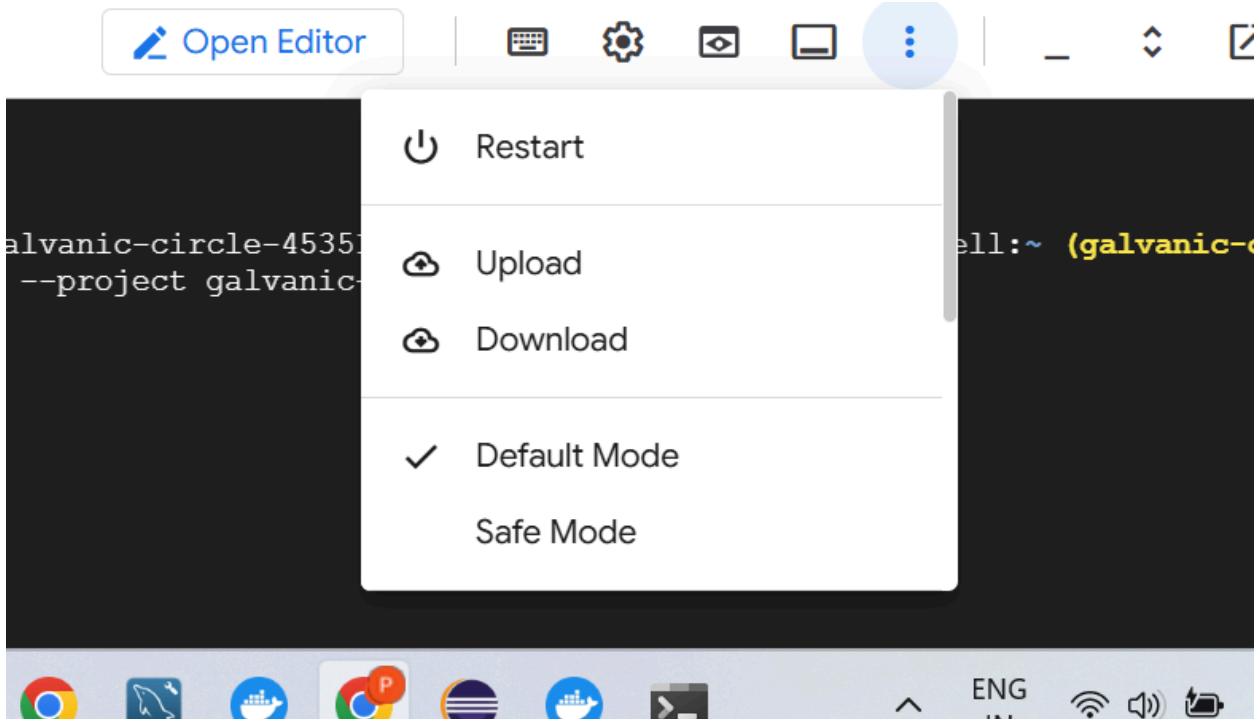
Click on cloud shell



Now hit enter

```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to galvanic-circle-453512-q6.
Use 'gcloud config set project [PROJECT ID]' to change to a different project.
galvanic-circle-453512-q6$ gcloud container clusters get-credentials kbs-docker --region us-central1 --project galvanic-circle-453512-q6
Fetching cluster endpoint and auth data.
kubeconfig entry generated for kbs-docker.
mallempavankalyan96@cloudshell:~ (galvanic-circle-453512-q6) $
```

Now upload the yml file



Check whether the file is submitted or not

```
mallempavankalyan96@cloudshell:~ (galvanic-circle-453512-q6) $ ls
dockerdeployment.yml  README-cloudshell.txt
mallempavankalyan96@cloudshell:~ (galvanic-circle-453512-q6) $

mallempavankalyan96@cloudshell:~ (galvanic-circle-453512-q6) $ kubectl apply -f .
mallempavankalyan96@cloudshell:~ (galvanic-circle-453512-q6) $ kubectl apply -f dockerdeployment.yml
Warning: autopilot-default-resources-mutator:Autopilot updated Deployment default/docker-k8s-demo-deployment: defaulted unspecified 'cpu' resource for container [docker-k8s-demo] (see http://g.co/gke/autopilot-defaults).
deployment.apps/docker-k8s-demo-deployment created
mallempavankalyan96@cloudshell:~ (galvanic-circle-453512-q6) $
```

Now go to the workloads and refresh

Overview Observability Cost Optimization						
Filter	Is system object: False		Filter workloads			
<input type="checkbox"/>	Name ↑	Status	Type	Pods	Namespace	Cluster
<input type="checkbox"/>	docker-k8s-demo-deployment	OK	Deployment	1/1	default	kbs-docker

Now we need to expose it

Port mapping

Port 1 80	Target port 1 6580	Protocol 1 TCP
--------------	-----------------------	-------------------

+ Add port mapping

Service type
Load balancer

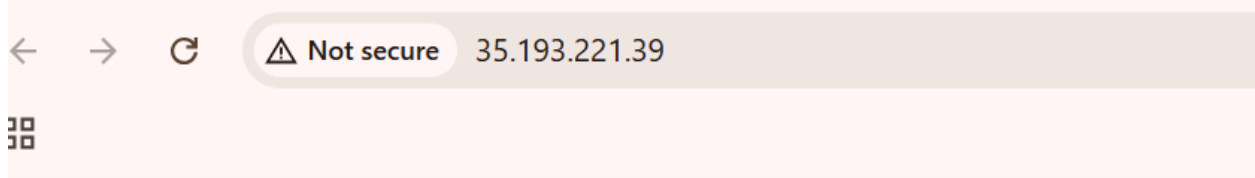
Service name
docker-k8s-demo-deployment-service

Expose

View YAML

* Indicates required field

Once it is done....click on the external end point



Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

Wed Mar 12 13:15:28 UTC 2025

There was an unexpected error (type=Not Found, status=404).

Use our endpoint which is exist in the program



in docker tutorial project of play java

Now finally our service is deployed on the cloud

1 Kubernetes cluster selected

[Delete](#) [Labels](#) [Register cluster to fleet](#)

[Overview](#) [Observability](#) [Cost Optimization](#)

Health ⓘ

100% healthy

No recommendations

Upgrade ⓘ

100% up to date

No recommendations

Estimated monthly cost ⓘ

\$0.00 / month · 0%

No recommendations

[Filter](#) Enter property name or value ⓘ

<input checked="" type="checkbox"/>	Status	Name ↑	Location	Tier ⓘ	Number of nodes	Total vCPUs	Total memory	Nodes
<input checked="" type="checkbox"/>		kbs-docker	us-central1	Standard		0.5	2 GB	

To avoid extra billing we need to delete the cluster