Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 15

Aim : To understand the basics of Kubernetes

Scenario : Your company and its application are scaling, and you hence need to make use of tools and services that would help you manage multiple containers and its management easily. On that note, understand the basics of Kubernetes and perform the following.

Tasks and Screenshots containing commands and output :

1. Learn what a Kubernetes cluster is.

Kubernetes coordinates a highly available cluster of computers that are connected to work as a single unit. The abstractions in Kubernetes allow you to deploy containerized applications to a cluster without tying them specifically to individual machines. To make use of this new model of deployment, applications need to be packaged in a way that decouples them from individual hosts: they need to be containerized. Containerized applications are more flexible and available than in past deployment models, where applications were installed directly onto specific machines as packages deeply integrated into the host. Kubernetes automates the distribution and scheduling of application containers across a cluster in a more efficient way. Kubernetes is an open-source platform and is production-ready.
A Kubernetes cluster consists of two types of resources:

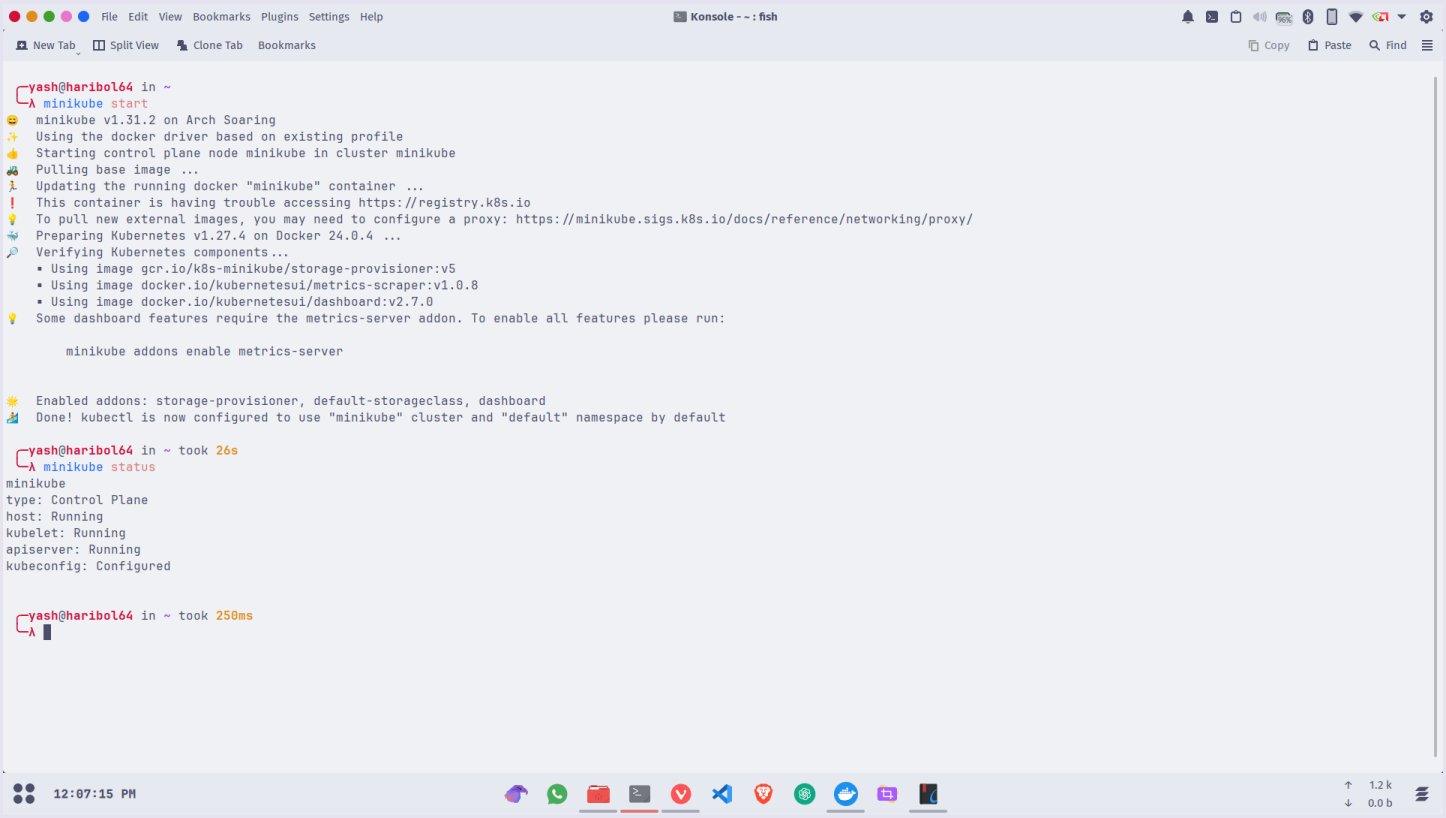
- The Control Plane coordinates the cluster.
- Nodes are the workers that run applications.

2. Learn what Minikube is.

Minikube is a lightweight tool that enables you to set up and run a single-node Kubernetes cluster locally, making it easier for developers to develop and test Kubernetes applications on their own machines.

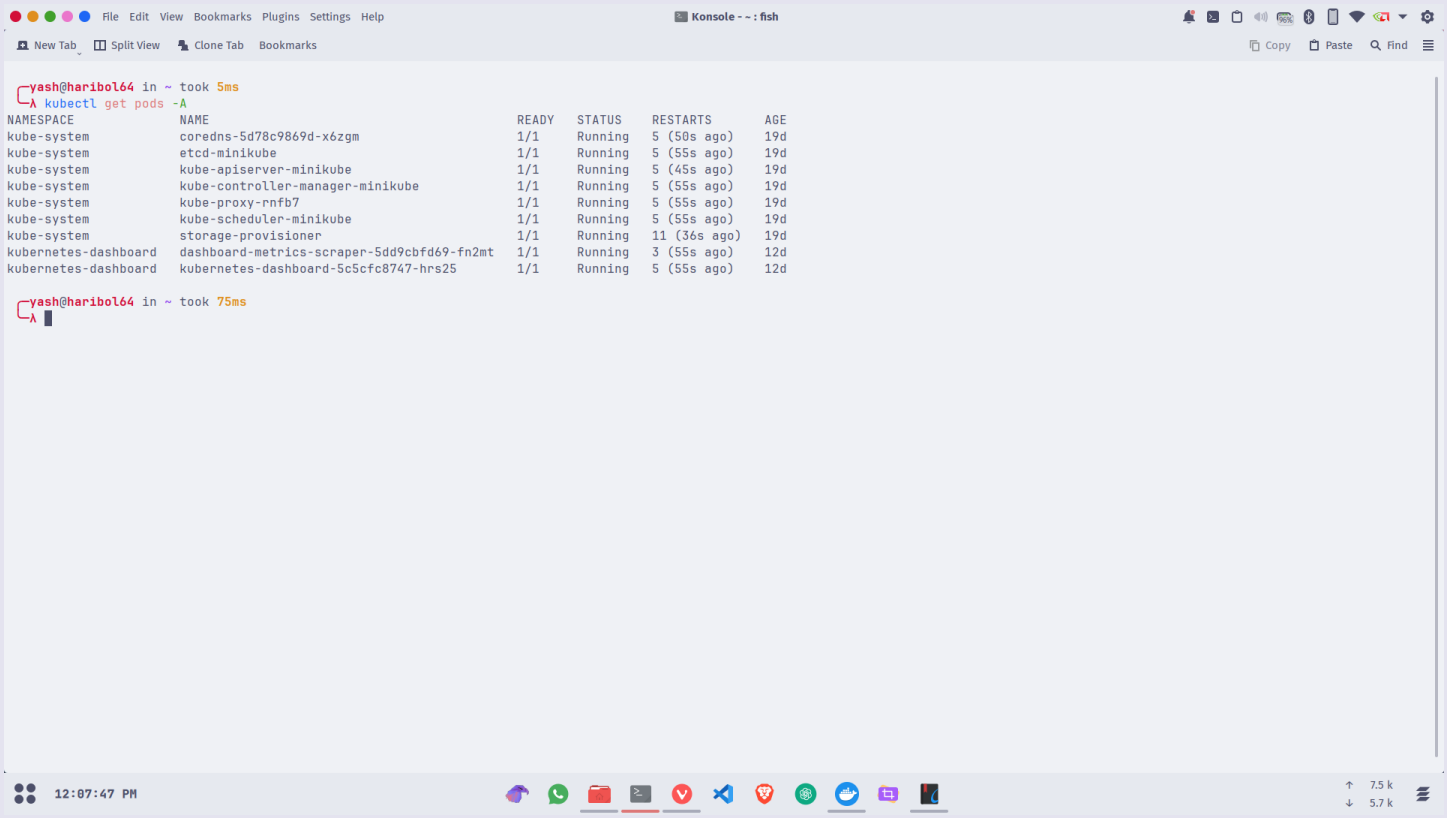
Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 15

3. Start a Kubernetes cluster using a terminal.



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 15

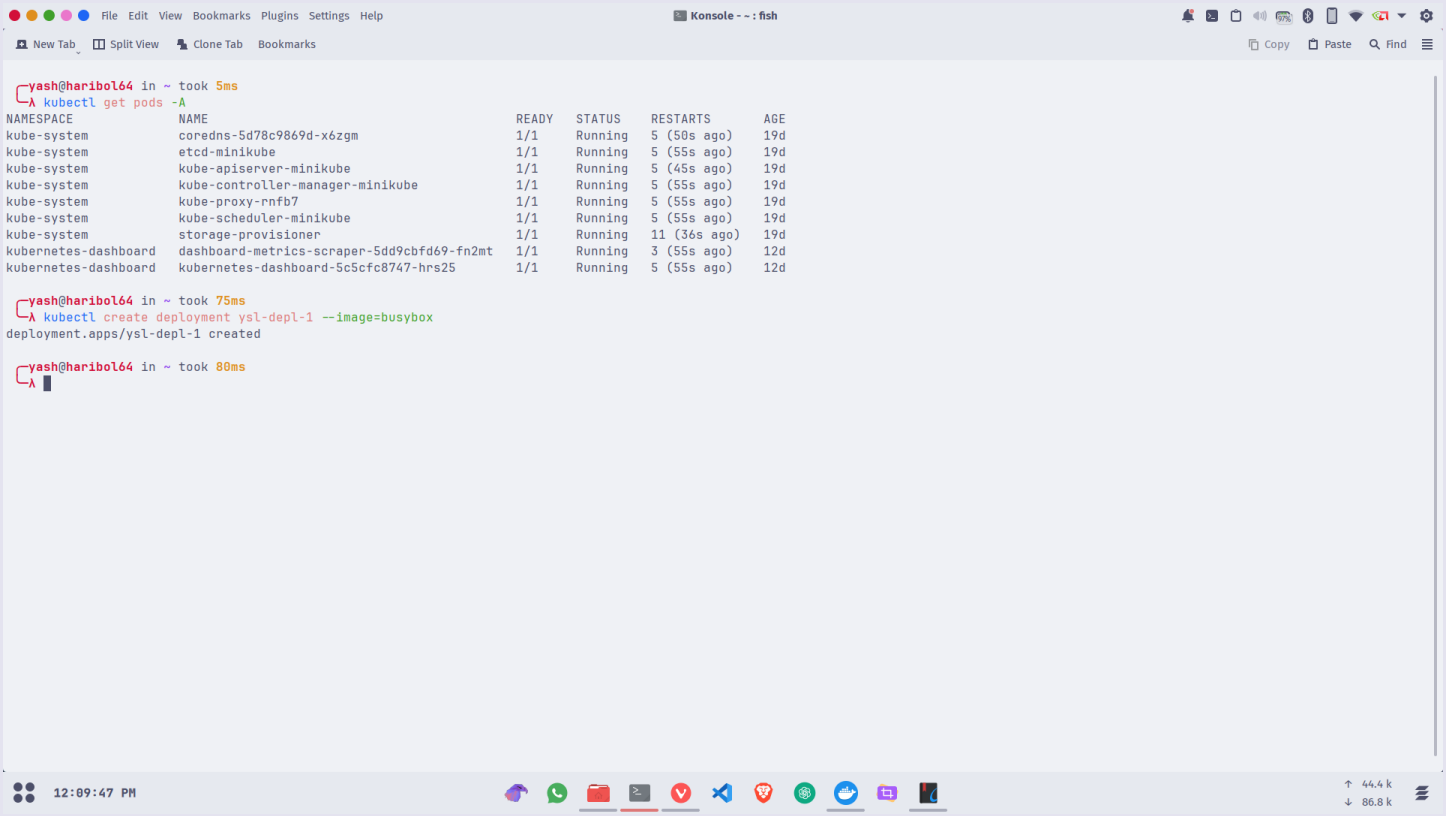
The command `kubectl get pods -A` is used to list all the pods in all namespaces within a Kubernetes cluster. It provides a comprehensive view of all running pods, regardless of the namespace they belong to, making it a valuable tool for monitoring and managing the entire cluster's pod resources.



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 15

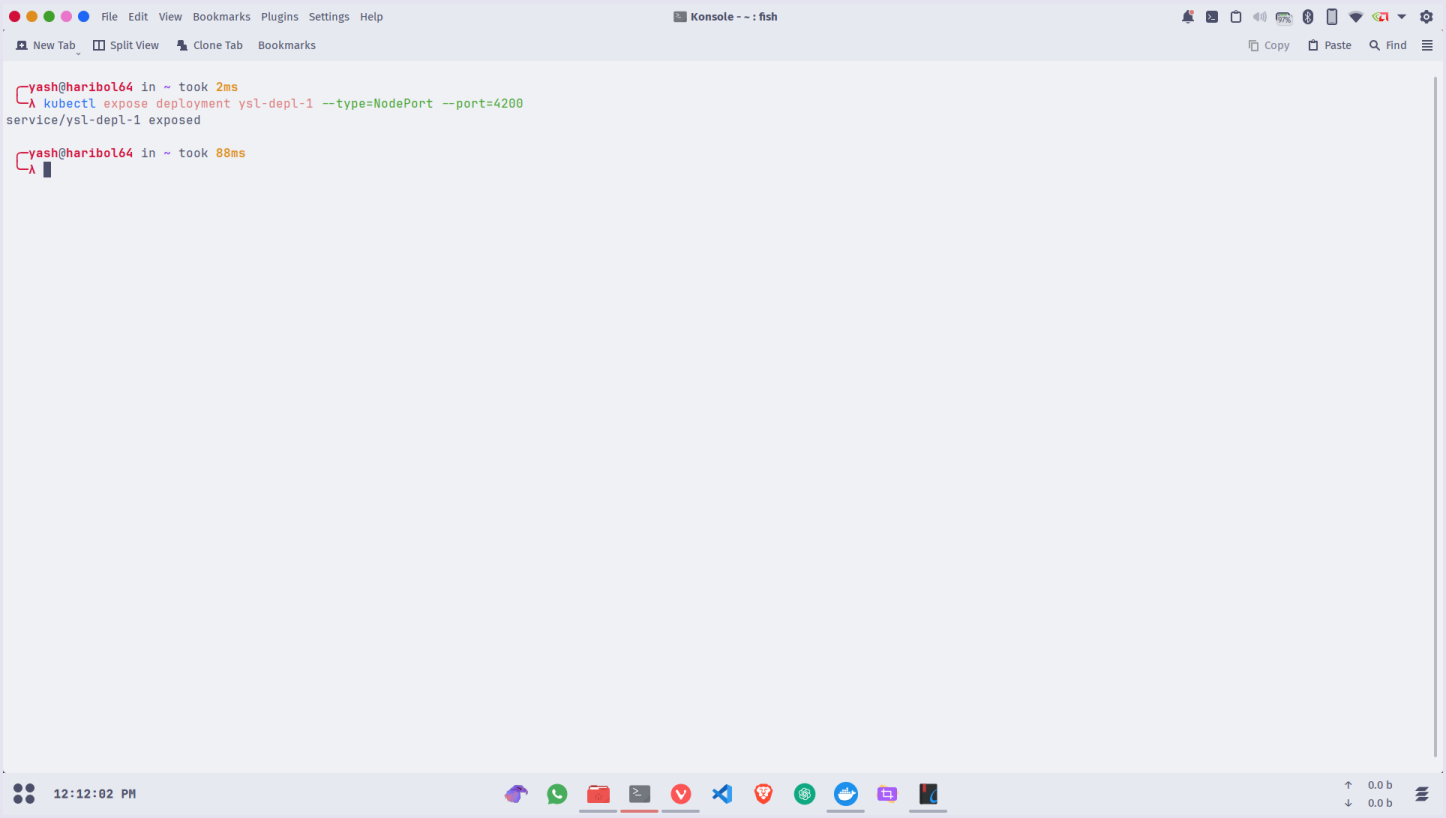
4. Learn about application Deployments.

a) Create deployment with image specified in command

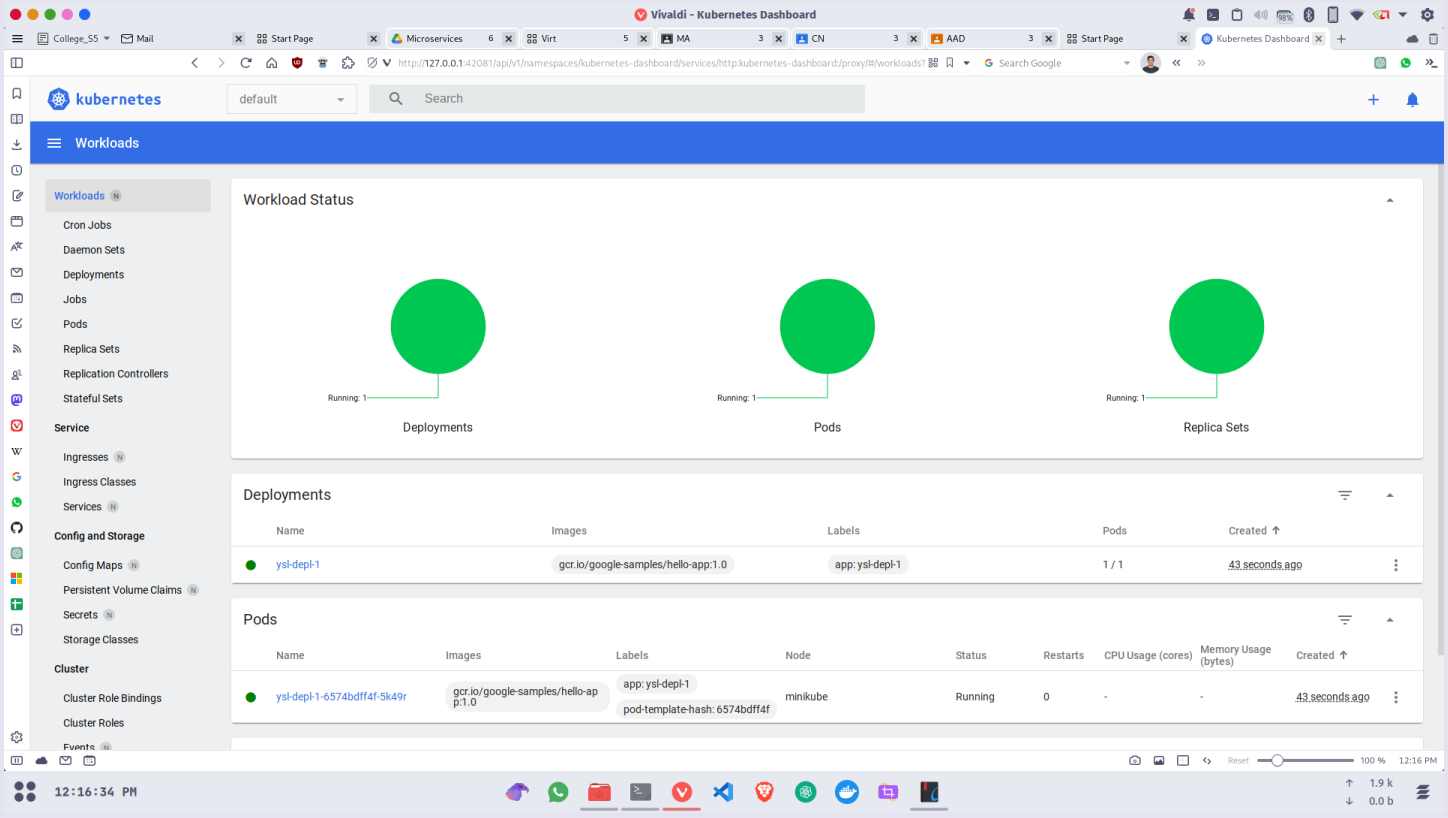


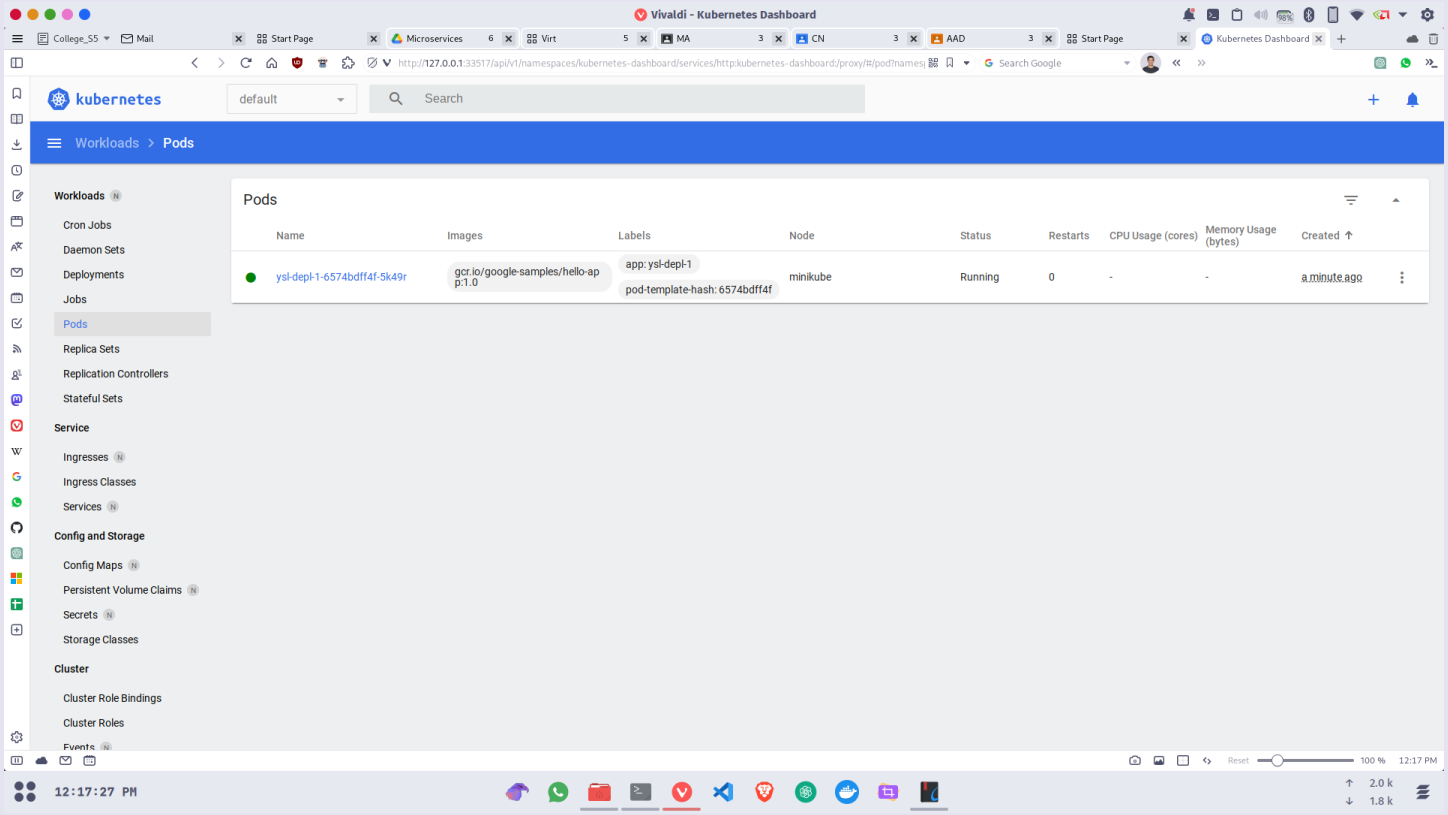
Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 15

b) Expose the deployment by creating a service of type and port specified.



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 15

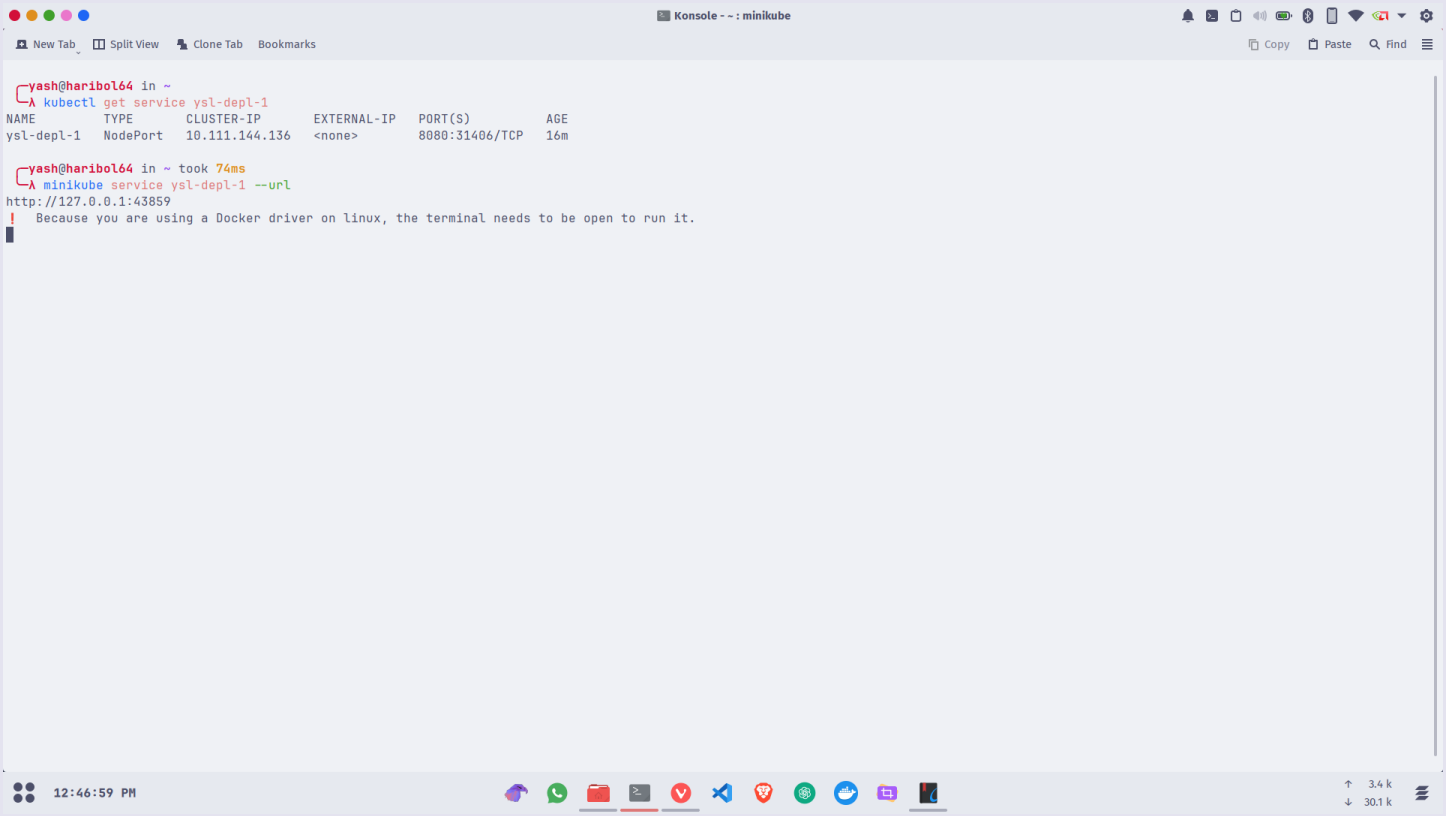




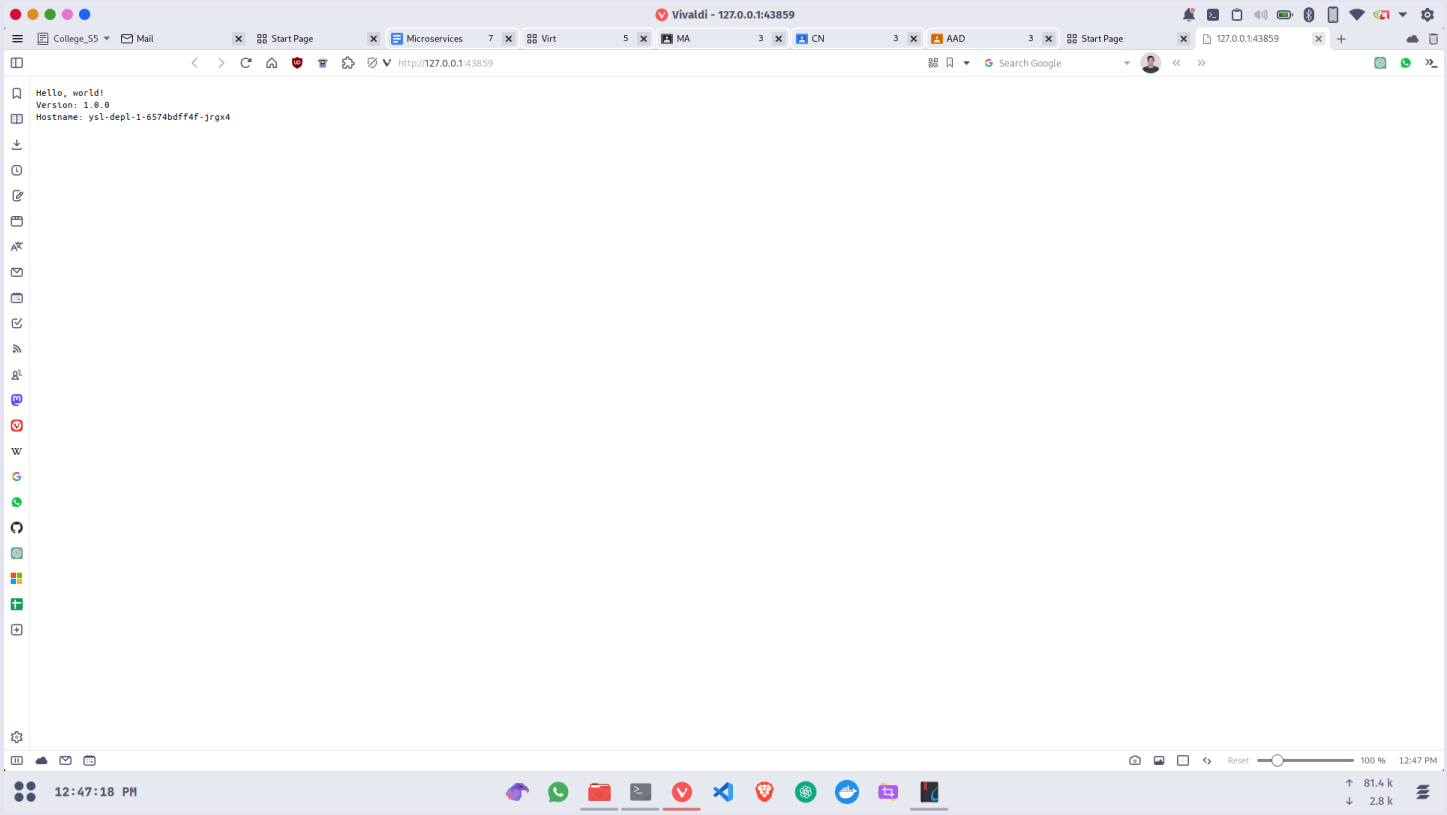
Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 15

5. Deploy your first app on Kubernetes with kubectl.

Get service info using kubectl and access it using minikube



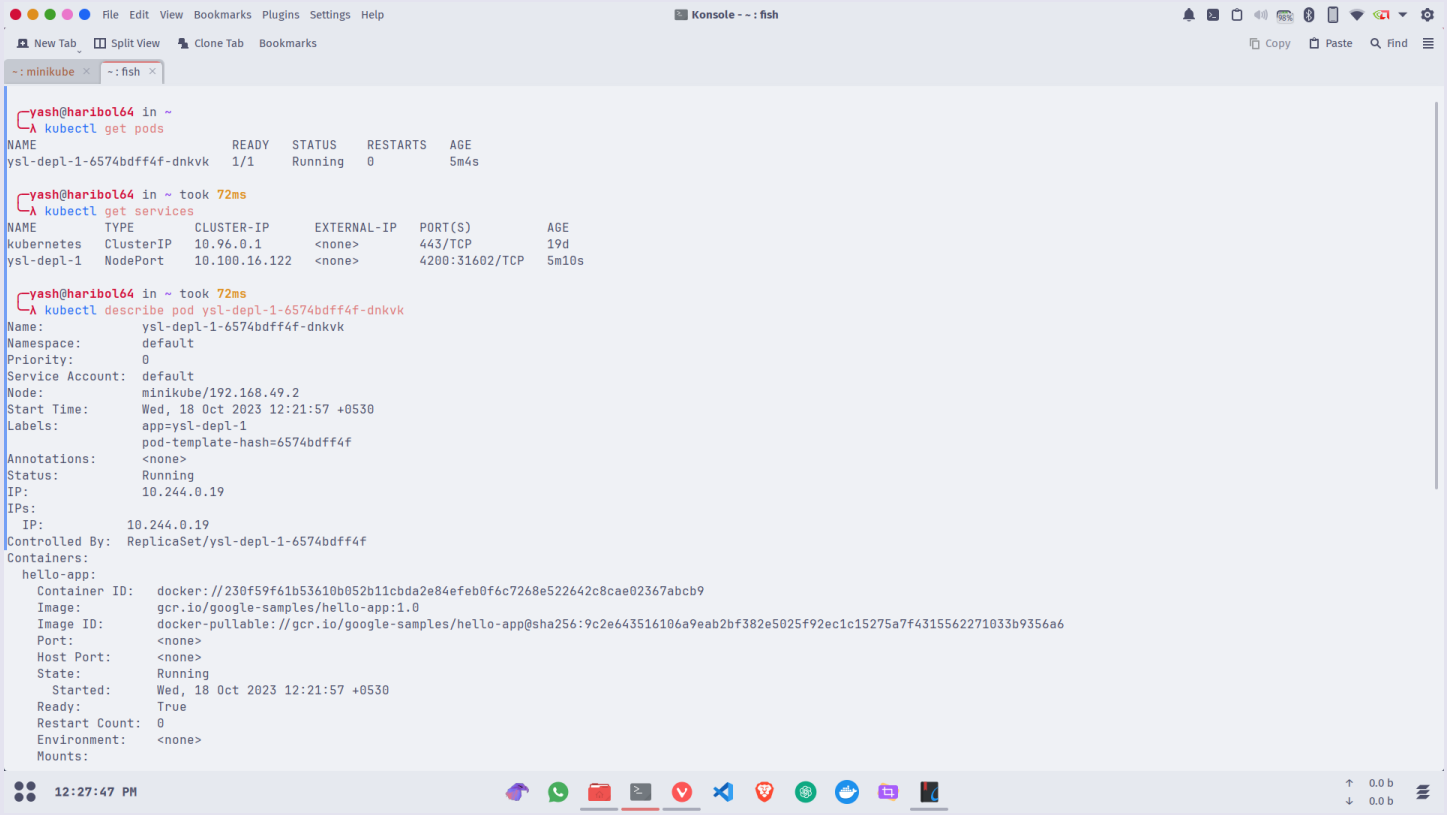
Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 15



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 15

6. Learn how to troubleshoot Kubernetes applications using the kubectl get, describe, logs and exec commands

Use **kubectl get** to list resources, then **kubectl describe** to get detailed information of a specific Kubernetes resource, also **kubectl logs** can be used to get logs of pods. Optionally, **kubectl exec** to execute command inside a container (useful for interactive shell apps or containers).



Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 51
Microservices Practical 15