Lab Exercise 8- Create POD in Kubernetes

NAME : JIYA TYAGI
SAP ID: 500119743
BATCH 2 DEVOPS

Objective:

- Understand the basic structure and syntax of a Kubernetes Pod definition file (YAML).
- Learn to create, inspect, and delete a Pod in a Kubernetes cluster.

Prerequisites

- Kubernetes Cluster: You need a running Kubernetes cluster. You can set up a local cluster using tools like Minikube or kind, or use a cloud-based Kubernetes service.
- kubectl: Install and configure kubectl to interact with your Kubernetes cluster.
- Basic Knowledge of YAML: Familiarity with YAML format will be helpful as Kubernetes resource definitions are written in YAML.

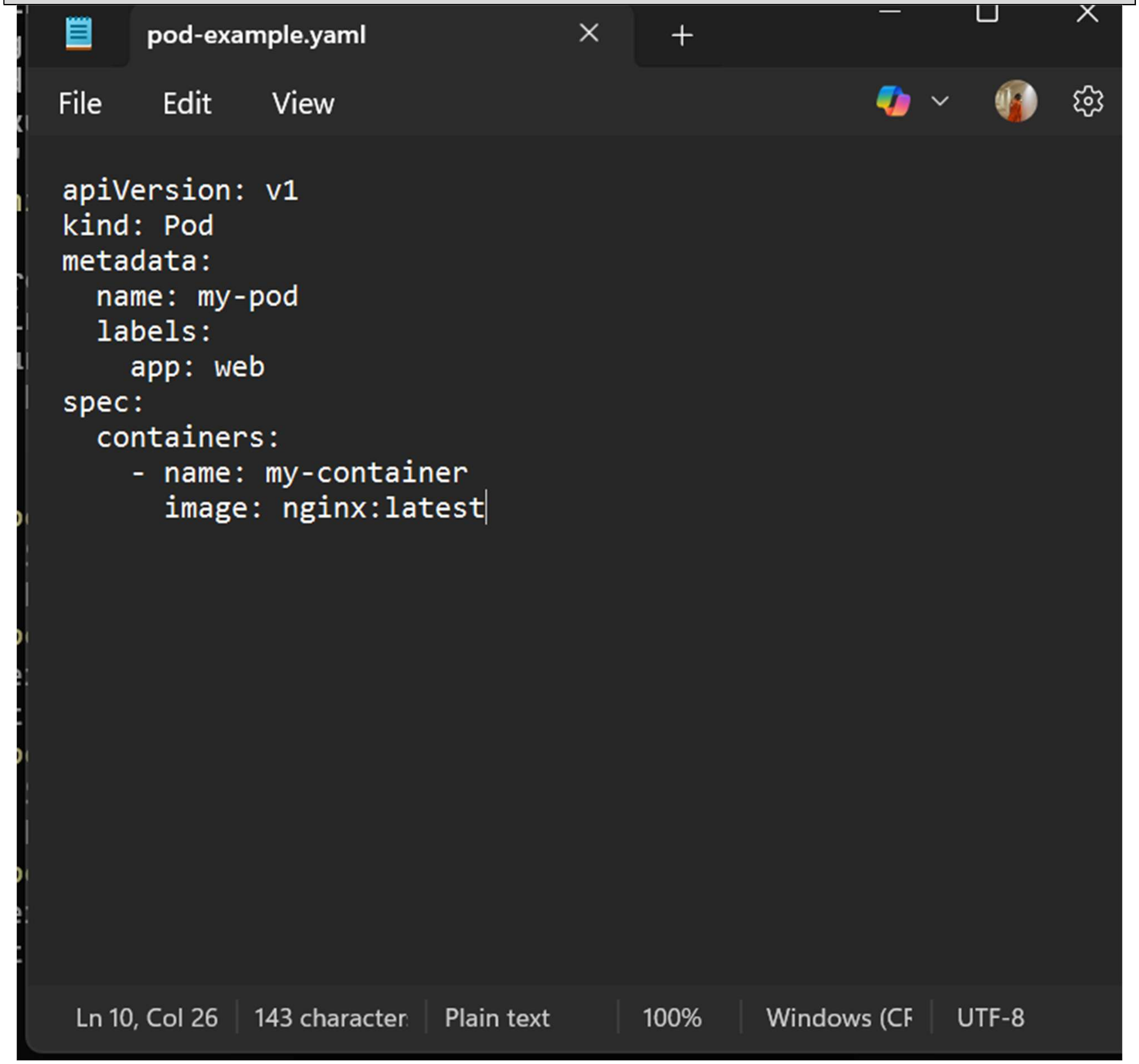
Step-by-Step Guide

Step 1: Create a YAML File for the Pod

We'll create a Pod configuration file named **pod-example.yaml**

```
apiVersion: v1
kind: Pod
metadata:
  name: my-pod
  labels:
    app: web
```

```
spec:
  containers:
    - name: my-container
      image: nginx:latest
```



```
apiVersion: v1
kind: Pod
metadata:
  name: my-pod
  labels:
    app: web
spec:
  containers:
    - name: my-container
      image: nginx:latest
```

Explanation of the YAML File

- apiVersion: Specifies the version of the Kubernetes API to use. For Pods, it's typically v1.
- kind: The type of object being created. Here it's a Pod.

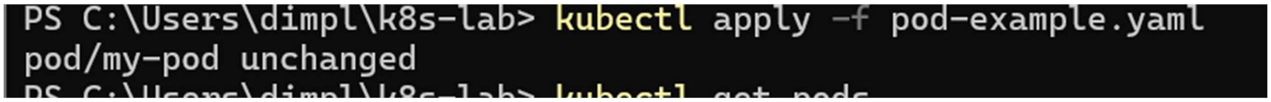
- metadata: Provides metadata about the object, including name and labels. The name must be unique within the namespace, and labels help in identifying and organizing Pods.
- spec: Contains the specifications of the Pod, including:
    - containers: Lists all containers that will run inside the Pod. Each container needs:
        - name: A unique name within the Pod.
        - image: The Docker image to use for the container.
        - ports: The ports that this container exposes.
        - env: Environment variables passed to the container.

Step 2: Apply the YAML File to Create the Pod

Use the kubectl apply command to create the Pod based on the YAML configuration file.

```
kubectl apply -f pod-example.yaml
```

This command tells Kubernetes to create a Pod as specified in the pod-example.yaml file.



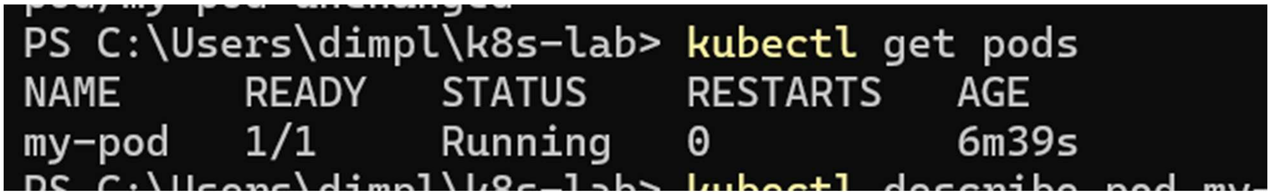
```
PS C:\Users\dimpl\k8s-lab> kubectl apply -f pod-example.yaml
pod/my-pod unchanged
PS C:\Users\dimpl\k8s-lab> kubectl get pods
```

Step 3: Verify the Pod Creation

To check the status of the Pod and ensure it's running, use:

```
kubectl get pods
```

This command lists all the Pods in the current namespace, showing their status, restart count, and other details.

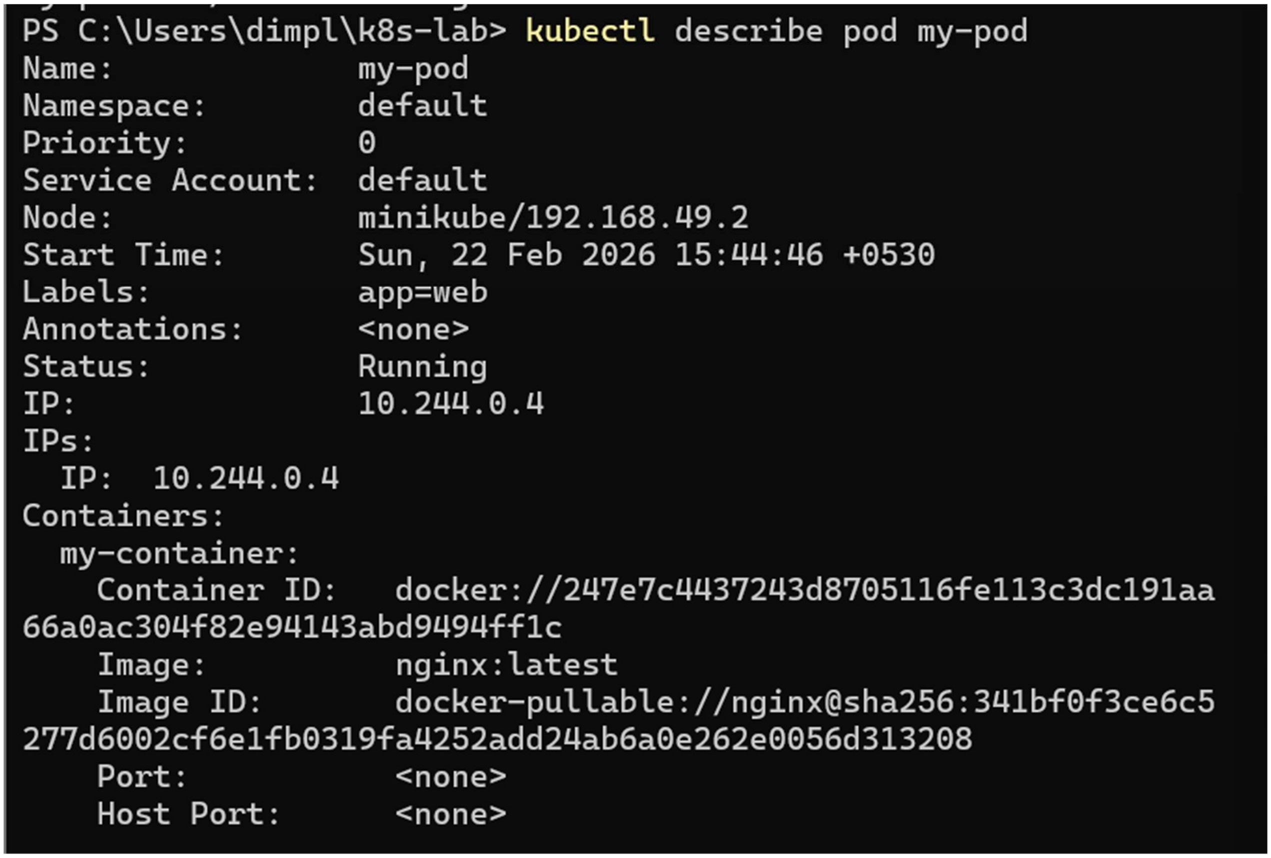


```
PS C:\Users\dimpl\k8s-lab> kubectl get pods
NAME     READY   STATUS    RESTARTS   AGE
my-pod   1/1     Running   0          6m39s
```

You can get detailed information about the Pod using:

```
kubectl describe pod my-pod
```

This command provides detailed information about the Pod, including its events, container specifications, and resource usage.



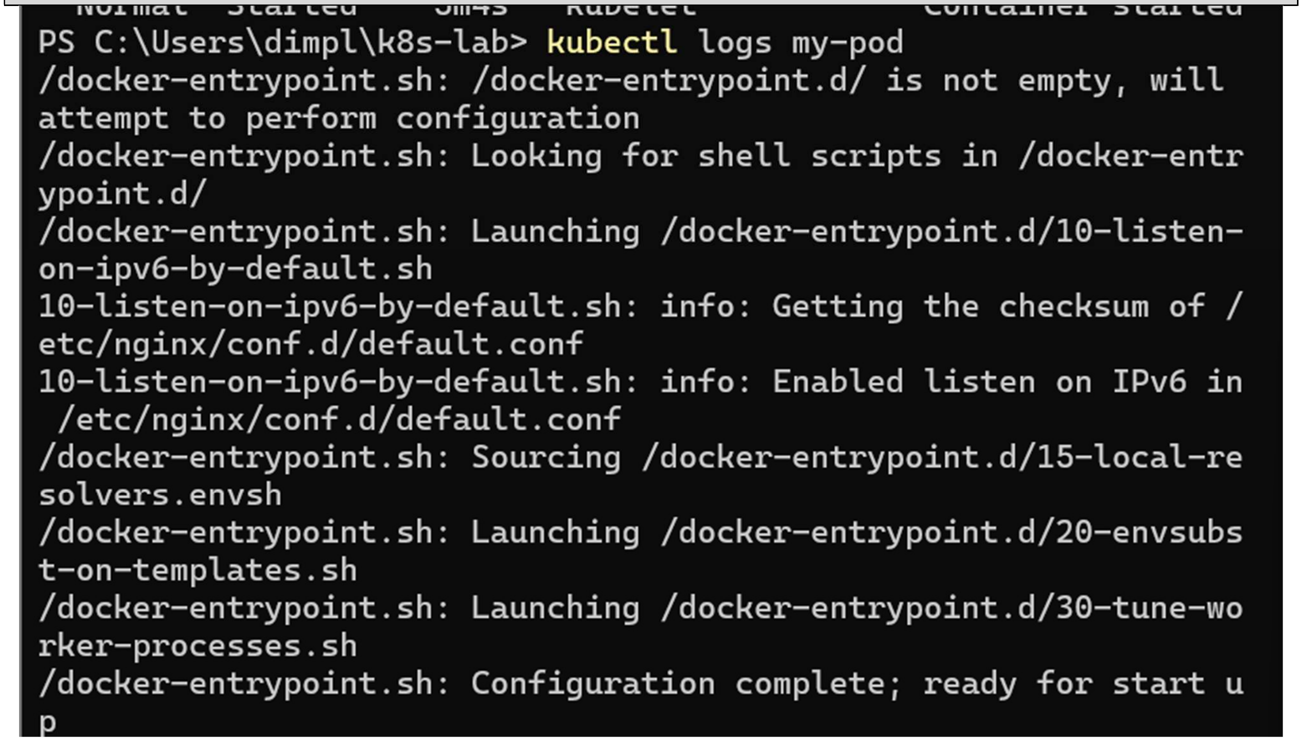
```
PS C:\Users\dimpl\k8s-lab> kubectl describe pod my-pod
Name:             my-pod
Namespace:        default
Priority:         0
Service Account:  default
Node:             minikube/192.168.49.2
Start Time:       Sun, 22 Feb 2026 15:44:46 +0530
Labels:           app=web
Annotations:      <none>
Status:           Running
IP:               10.244.0.4
IPs:
  IP:  10.244.0.4
Containers:
  my-container:
    Container ID:   docker://247e7c4437243d8705116fe113c3dc191aa
66a0ac304f82e94143abd9494ff1c
    Image:          nginx:latest
    Image ID:       docker-pullable://nginx@sha256:341bf0f3ce6c5
277d6002cf6e1fb0319fa4252add24ab6a0e262e0056d313208
    Port:           <none>
    Host Port:      <none>
```

Step 4: Interact with the Pod

You can interact with the running Pod in various ways, such as accessing the logs or executing commands inside the container.

View Logs: To view the logs of the container in the Pod:

```
kubectl logs my-pod
```

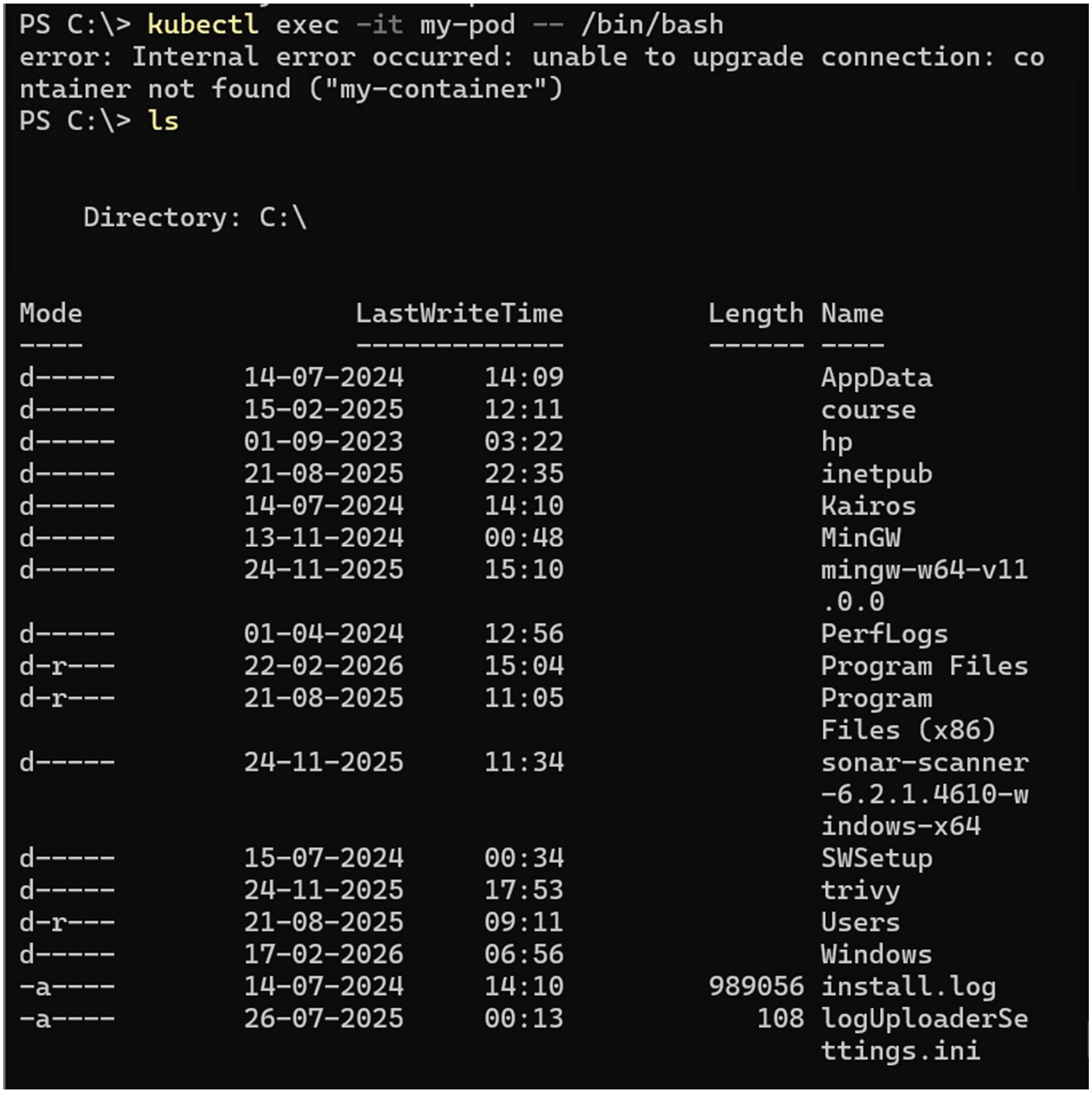


```
  Normal  Started    3m4s   kubelet            Container started
PS C:\Users\dimpl\k8s-lab> kubectl logs my-pod
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will
attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entr
ypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-
on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /
etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in
 /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-re
solvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubs
t-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-wo
rker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start u
p
```

Execute a Command: To run a command inside the container:

```
kubectl exec -it my-pod -- /bin/bash
```

The -it flag opens an interactive terminal session inside the container, allowing you to run commands.



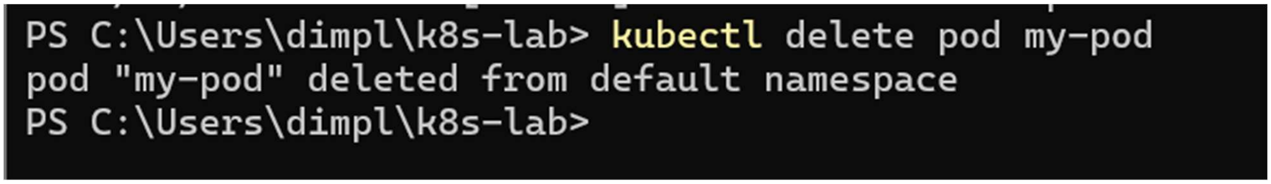
```
PS C:\> kubectl exec -it my-pod -- /bin/bash
error: Internal error occurred: unable to upgrade connection: co
ntainer not found ("my-container")
PS C:\> ls


    Directory: C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        14-07-2024     14:09                AppData
d-----        15-02-2025     12:11                course
d-----        01-09-2023     03:22                hp
d-----        21-08-2025     22:35                inetpub
d-----        14-07-2024     14:10                Kairos
d-----        13-11-2024     00:48                MinGW
d-----        24-11-2025     15:10                mingw-w64-v11
                                                  .0.0
d-----        01-04-2024     12:56                PerfLogs
d-r---        22-02-2026     15:04                Program Files
d-r---        21-08-2025     11:05                Program
                                                  Files (x86)
d-----        24-11-2025     11:34                sonar-scanner
                                                  -6.2.1.4610-w
                                                  indows-x64
d-----        15-07-2024     00:34                SWSetup
d-----        24-11-2025     17:53                trivy
d-r---        21-08-2025     09:11                Users
d-----        17-02-2026     06:56                Windows
-a----        14-07-2024     14:10         989056 install.log
-a----        26-07-2025     00:13            108 logUploaderSe
                                                  ttings.ini
```

Step 5: Delete the Pod

To clean up and remove the Pod when you're done, use the following command:

```
kubectl delete pod my-pod
```

This command deletes the specified Pod from the cluster.



```
PS C:\Users\dimpl\k8s-lab> kubectl delete pod my-pod
pod "my-pod" deleted from default namespace
PS C:\Users\dimpl\k8s-lab>
```