UNIVERSITY OF PETROLEUM & ENERGY STUDIES
Dehradun

ACO LAB

NAME- YADRISHI DIXIT
BRANCH- COMPUTER SCIENCE ENGINEERING
BATCH- B-4 DEVOPS
SAP ID- 500097959
ROLL NO- R2142211468

SUBMITTED TO- Dr. Hitesh Kumar Sharma

EXPERIMENT – 7

AIM: Creating Pods in Kubernetes

Below is a lab exercise that will help you understand and practice creating pods in Kubernetes:

Task 1: Start Kubernetes in Docker-Desktop

- Start Kubernetes service in Docker-Desktop

Task 2: Creating a Simple Pod

- Create a simple YAML manifest file named pod.yaml to define a basic Pod in Kubernetes. An example of the file content is as follows:

```
apiVersion: v1

kind: Pod

metadata:

  name: my-nginx-pod

  labels:

    app: lbnginx

spec:

containers:

  - name: nginx-container
```

- Apply the Pod configuration using the following command:

   " kubectl apply -f pod.yaml "

```
91983@DELL MINGW64 ~/OneDrive/Desktop/SEM5/Kubernetes (main)
$ kubectl config use-context docker-desktop
Switched to context "docker-desktop".

91983@DELL MINGW64 ~/OneDrive/Desktop/SEM5/Kubernetes (main)
$ kubectl config current-context
docker-desktop

91983@DELL MINGW64 ~/OneDrive/Desktop/SEM5/Kubernetes (main)
$ kubectl apply -f pod.yaml
pod/my-nginx-pod created
```

- Check the status of the Pod using the following command:

   " kubectl get pods "

```
91983@DELL MINGW64 ~/OneDrive/Desktop/SEM5/Kubernetes (main)
$ kubectl get pods
NAME            READY   STATUS     RESTARTS   AGE
my-nginx-pod    1/1     Running    0          2m
```

Task 3: Accessing the Pod

- Access the Pod by using *port forwarding* to the container. Run the following command:

   " kubectl port-forward my-nginx-pod 8081:80 "

```
91983@DELL MINGW64 ~/OneDrive/Desktop/SEM5/Kubernetes (main)
$ kubectl port-forward my-nginx-pod 8081:80
Forwarding from 127.0.0.1:8081 -> 80
Forwarding from [::1]:8081 -> 80
```

- Access the Nginx server running in the Pod by opening a web browser and navigating to http://localhost:8081
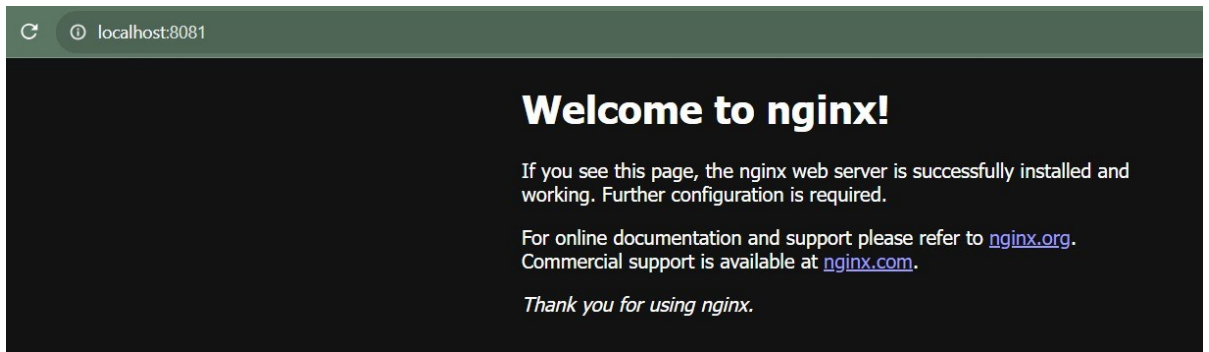
Task 4: Exploring Pod Details

- Retrieve detailed information about the Pod using the following command:

" kubectl describe pod my-nginx-pod "

```
Conditions:
  Type              Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  kube-api-access-bxhch:
    Type:                      Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:    3607
    ConfigMapName:             kube-root-ca.crt
    ConfigMapOptional:         <nil>
    DownwardAPI:               true
QoS Class:                     BestEffort
Node-Selectors:                <none>
Tolerations:                   node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                               node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type    Reason     Age    From                Message
  ----    ------     ----   ----                -------
  Normal  Scheduled  10m    default-scheduler   Successfully assigned default/my-nginx-pod to docker-desktop
  Normal  Pulling    10m    kubelet             Pulling image "nginx"
  Normal  Pulled     9m3s   kubelet             Successfully pulled image "nginx" in 57.368s (57.368s including waiting)
  Normal  Created    9m2s   kubelet             Created container nginx-container
  Normal  Started    9m1s   kubelet             Started container nginx-container
```

- Check the logs of the Pod to understand its behavior using the following command:

" kubectl logs my-nginx-pod "

```
91983@DELL MINGW64 ~/OneDrive/Desktop/SEM5/Kubernetes (main)
$ kubectl logs my-nginx-pod
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.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-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2023/11/13 12:45:39 [notice] 1#1: using the "epoll" event method
2023/11/13 12:45:39 [notice] 1#1: nginx/1.25.3
2023/11/13 12:45:39 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2023/11/13 12:45:39 [notice] 1#1: OS: Linux 5.10.102.1-microsoft-standard-WSL2
2023/11/13 12:45:39 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2023/11/13 12:45:39 [notice] 1#1: start worker processes
2023/11/13 12:45:39 [notice] 1#1: start worker process 29
2023/11/13 12:45:39 [notice] 1#1: start worker process 30
2023/11/13 12:45:39 [notice] 1#1: start worker process 31
2023/11/13 12:45:39 [notice] 1#1: start worker process 32
2023/11/13 12:45:39 [notice] 1#1: start worker process 33
2023/11/13 12:45:39 [notice] 1#1: start worker process 34
2023/11/13 12:45:39 [notice] 1#1: start worker process 35
2023/11/13 12:45:39 [notice] 1#1: start worker process 36
127.0.0.1 - - [13/Nov/2023:12:51:15 +0000] "GET / HTTP/1.1" 200 615 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Geck
o) Chrome/119.0.0.0 Safari/537.36" "-"
127.0.0.1 - - [13/Nov/2023:12:51:15 +0000] "GET /favicon.ico HTTP/1.1" 404 555 "http://localhost:8081/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) Appl
eWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36" "-"
2023/11/13 12:51:15 [error] 30#30: *1 open() "/usr/share/nginx/html/favicon.ico" failed (2: No such file or directory), client: 127.0.0.1, server: loca
lhost, request: "GET /favicon.ico HTTP/1.1", host: "localhost:8081", referrer: "http://localhost:8081/"
```

Task 5: Deleting the Pod

- Delete the Pod using the following command:

" kubectl delete pod my-nginx-pod "

```
91983@DELL MINGW64 ~/OneDrive/Desktop/SEM5/Kubernetes (main)
$ kubectl delete pod my-nginx-pod
pod "my-nginx-pod" deleted
```

- Verify that the Pod has been deleted by running the **"kubectl get pods"** command.

```
91983@DELL MINGW64 ~/OneDrive/Desktop/SEM5/Kubernetes (main)
$ kubectl get pods
No resources found in default namespace.
```

Task 6: Advanced Pod Configuration

- Experiment with advanced Pod configuration options such as environment variables, volume mounts, resource limits, and labels.
- Update the Pod manifest file and apply the changes to the Kubernetes cluster.

Task 7: Cleanup

Delete any remaining Pods, services, and deployments created during the exercise using the appropriate kubectl delete commands.

Task 8: Documentation and Best Practices

Document your findings and the best practices for creating and managing Pods in Kubernetes.

Through this exercise, you'll gain a better understanding of how to create, manage, and interact with Pods in Kubernetes. Adjust the exercise based on your specific use case and requirements.