Node-helloWorld Project

1- Deploy the Application

> Define your Deployment manifest for the Node.js app, apply it to the cluster, and confirm that your Pod comes up in the Ready state.
>
> > Deliverable
> > Deployment YAML and evidence that the Pod is running and healthy.

- Start Minikube using

```
minikube start --driver=docker
```

- Make sure that it is properly running

> ```
> minikube status
> ```
>
> the output should be
>
> ```
> minikube
> type: Control Plane
> host: Running
> kubelet: Running
> apiserver: Running
> kubeconfig: Configured
> ```

PROF

- Now, navigate to the project directory and apply the deployment

```
kubectl apply -f deployment.yaml
```

- To make sure that the deployment have been successfully deployed, run this command to only watch the labeled nodejs-hello pods

```
kubectl get pods -o wide -l app=nodejs-hello -w
```

- After checking that the pods are running, to test their reachability, we have to create a new pod on the same node to curl any of the deployed pods' ips and the port they are listening on `Port:3000`, this is a curl pod to use for the test run command

```
kubectl run curl-pod --image=curlimages/curl --rm -it -- /bin/sh
```

This will open sh in a curl pod, now curl one of the pods ips on the port 3000

```
curl {pod ip}:3000
```

the output should be

```
Hello Node!
```

2- Provision and Mount Storage

Create a PersistentVolumeClaim for your app's data. Update the Deployment manifest to mount the claim into the container's filesystem, then verify that writes persist.

> Deliverable
> PVC manifest, updated Deployment manifest, and proof of successful write to the mounted volume.

- First, apply the PersistentVolumeClaim:

```
kubectl apply -f pvc.yaml


- Now, update the `deployment.yaml` to mount the volume:

```yaml
volumeMounts:
  - mountPath: "/data"
    name: storage-volume

volumes:
  - name: storage-volume
    persistentVolumeClaim:
      claimName: nodejs-pvc
```

PROF

- Also, inside the container, write a file to `/data` to confirm persistence:

```
command: ["sh", "-c"]
args:
  - |
```
```

```
    echo 'Hello Persistent Storage' > /data/hello.txt && \
    tail -f /dev/null
```

- Re-apply the updated deployment:

```
kubectl apply -f deployment.yaml
```

- Verify that the file was written:

```
kubectl exec -it <nodejs-hello> -- cat /data/hello.txt
```

> The output should be:

```
Hello Persistent Storage
```