**TASK 4** Jaswanth S  
 22CDR035

**PROCEDURE**

**1. Initialize Minikube with Docker Driver**

Start Minikube with the Docker driver and force its execution to ensure the Kubernetes cluster is up and running.

**2. Deploy the Web Application**

Create a Kubernetes deployment for the nginx web server, which will act as the application inside the cluster. This ensures the pod is managed and maintained automatically.

**3. Expose the Application via NodePort**

Define a NodePort service to expose the web application outside the cluster, allowing access from an external source. The service maps the internal container port (80) to a NodePort on the host machine.

**4. Check the Status of Running Pods**

List all the running pods to confirm that the deployment was successful and the application is running inside the cluster.

**5. Retrieve Service Information**

Get details about the services running in Kubernetes to identify the assigned NodePort, which will be used to access the web application externally.

**6. Access the Application Using Minikube**

Obtain the Minikube service URL and open it in a browser or use curl to verify the application is accessible.

**7. Continuously Monitor Pod Status**

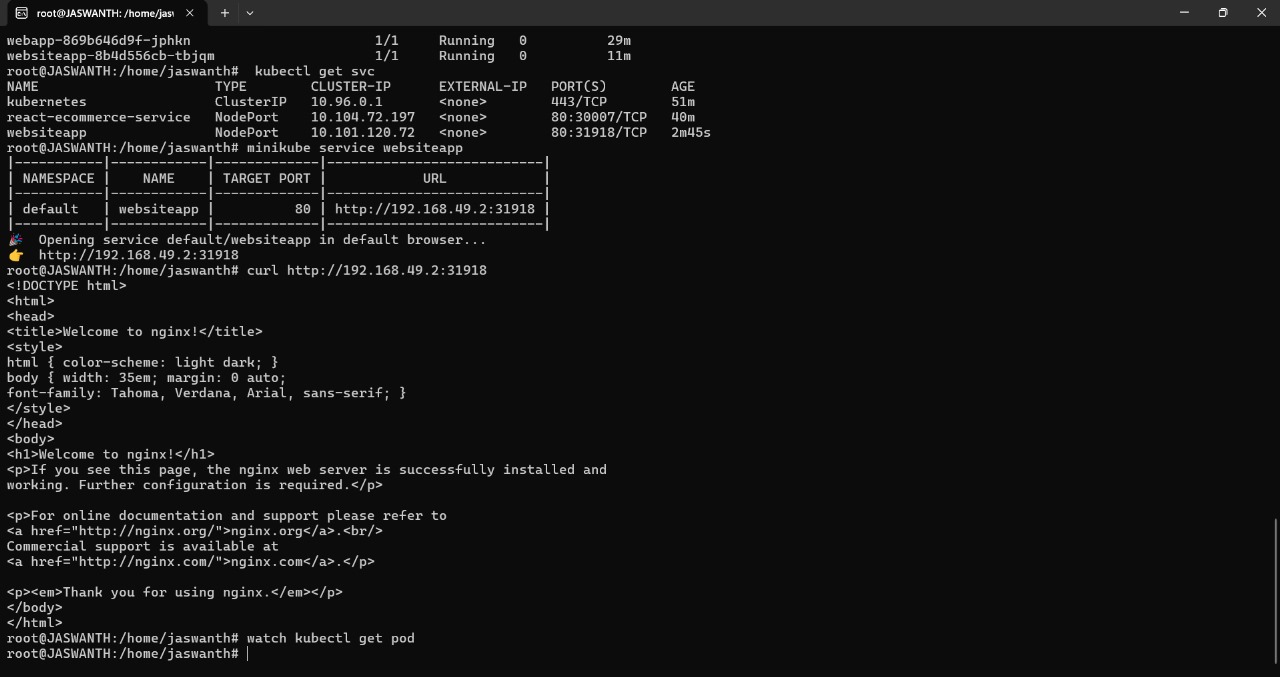
Use the watch command to keep an eye on the status of the pods in real-time and ensure they remain in a running state.

**8. View Application Logs**

Check the logs of the running web application to track its activity and identify any potential issues.

**9. Display Executed Commands**

Retrieve the command history to review previously executed commands, useful for debugging or reference purposes.

**OUTPUT  
  
**