Pods

This lab consists of a list of exercises to demonstrate and understand the most commonly used kubernetes commands and concepts to ramp up your kubernetes competency skills in the area of `Pods` .

Learning Outcomes

After completing the lab, you will be able to understand and use Kubernetes concepts based on use-case scenarios in the following domains:

1. Advanced Configuration & Custom Commands
2. Environment Variables
3. Probes and Observability
4. Volumes
5. SecurityContexts

Start the minikube

1. Start minikube locally `minikube start --driver=virtualbox`
2. Verify the kubectl context `kubectl config get-contexts` is set to minikube. If not, set it to minikube `kubectl config use-context minikube`

Setup the working directory

- Create all manifest resources in the directory `~/workspace/kubernetes-manifests/competencies` . Watch out for the right file names in the solution section.

  ```
  mkdir -p ~/workspace/kubernetes-manifests/competencies
  cd ~/workspace/kubernetes-manifests
  ```

Advanced Configuration & Custom Commands

1. Create the directory `pods` inside `competencies`

   ```
   mkdir ~/workspace/kubernetes-manifests/competencies/pods
   ```

2. Create a busybox pod which prints "Hello from pages Yellow Pages" on the console using manifest file.

   ▶ Click to see solution

3. Create a busybox pod which prints "Hello from pages Yellow Pages" on the console, for every 10 seconds infinitely using manifest file.

   ▶ Click to see solution
4. Create an nginx pod and test it in the browser

   ▶ Click to see solution
5. Create a busybox pod which prints "Hello from pages Yellow Pages" on the console imperatively

   ▶ Click to see solution
6. Create a busybox pod which prints `current date` on the console, for every 10 seconds infinitely

   ▶ Click to see solution

Environment variables:

1. Create environment variables for busybox pod with key= `course` and value= `K8s` and also prints "Hello!" on the console, for every 10 seconds infinitely using manifest file. Verify the environment variables are set properly

   ▶ Click to see solution
2. Create environment variables for pages application with key= `PAGE_CONTENT` and value= `Hello from pages environment` Verify the environment variables are set properly

   ▶ Click to see solution

Probes and Observability:

1. Create a pod that runs a web server. Configure readiness probe within the pod so that Kubernetes sends the first request after 5 seconds and check if port 80 is accessible. Hint: use nginx image

   ▶ Click to see solution
2. In the previous scenario add a liveness probe of type exec command to check if index.html exists whose exit code determines the success of the probe every 10 seconds.

   ▶ Click to see solution
3. Configure liveness and readiness probe for pages to check if the application is accessible at port 8080

   ▶ Click to see solution

Volumes

1. Create the directory `volumes` inside `competencies`

```
mkdir ~/workspace/kubernetes-manifests/competencies/volumes
```

2. Mount a directory `/var/output` to the container which writes the current date to a file called `log.txt` inside the mounted directory `/var/output` for every `5 seconds`. The directory should reside inside the pod and should cease to exist once the pod terminates.

   HINT

   Use `EmptyDir` type of volume

   ▶ Click to see solution

3. Mount a directory `/var/output` to the container which writes the current date to a file called `log.txt` inside the mounted directory `/var/output` for every `5 seconds`. The directory should reside external to the pod and should exist even if the pod terminates or restarts.

   HINT

   Use `HostPath` type of volume

   ▶ Click to see solution

SecurityContext

1. Create a pod with a container that writes the current date to a file in a mounted volume at /etc/kal-directory/date-file.txt every five seconds. The created file should have the user ID 45 and the group ID 231 which will be useful for auditing and tracking files system updates. Use alpine image.

   ▶ Click to see solution