Assignment – 1

Tasks:

1. Install minikube on your local machine and run the few commands of kubectl.
2. Try to give alias to kubectl as "k".
3. Try out "kubectl get pods" and "kubectl get nodes" and check what are the information that console is giving back.
4. Optional - Try to create EKS configuration (Note that this will cause you cost. Also remember to delete all services which you created for EKS)

Steps:

1. Go to windows powershell and run following commands to create a new directory named ‘minikube’ and install it.

New-Item -Path 'c:\' -Name 'minikube' -ItemType Directory -Force

Invoke-WebRequest -OutFile 'c:\minikube\minikube.exe' -Uri 'https://github.com/kubernetes/minikube/releases/latest/download/minikube-windows-amd64.exe' -UseBasicParsing

2. Now to add minikube.exe to our environment variable path run this command.

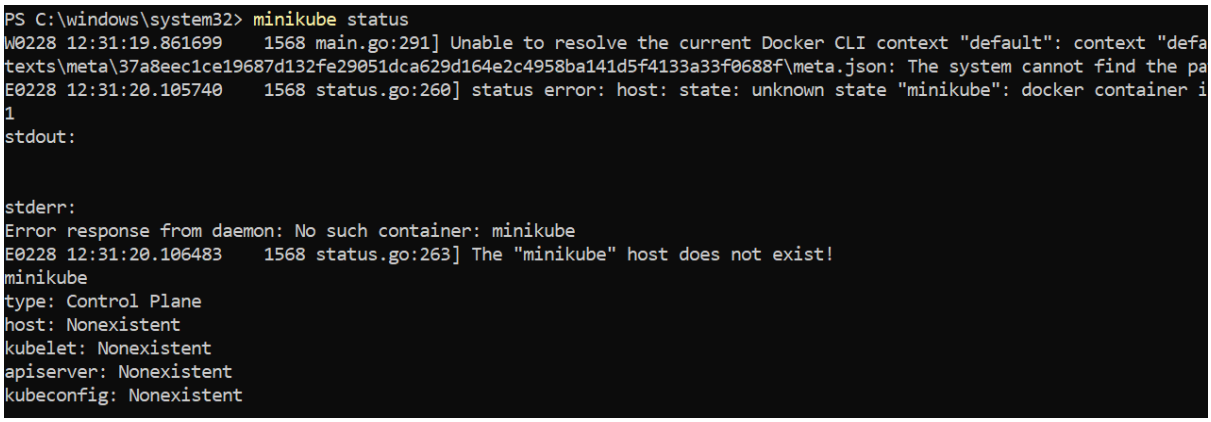
$oldPath = [Environment]::GetEnvironmentVariable('Path', [EnvironmentVariableTarget]::Machine)

if ($oldPath.Split(';') -inotcontains 'C:\minikube'){

[Environment]::SetEnvironmentVariable('Path', $('{0};C:\minikube' -f $oldPath), [EnvironmentVariableTarget]::Machine)

}

3. Now to check minikube status, use the command : minikube status
Here we can see that kubelet, apiserver, kubeconfig are Nonexistent.



4. Now to start the minikube, run the command: *minikube start*

```
PS C:\windows\system32> minikube start
W0228 12:31:42.481354    3616 main.go:291] Unable to resolve the current Docker CLI context "default
texts\meta\37a8eec1ce19687d132fe29051dca629d164e2c4958ba141d5f4133a33f0688f\meta.json: The system ca
* minikube v1.32.0 on Microsoft Windows 11 Home Single Language 10.0.22631.3155 Build 22631.3155
* Using the docker driver based on existing profile
* Starting control plane node minikube in cluster minikube
* Pulling base image ...
* docker "minikube" container is missing, will recreate.
* Creating docker container (CPUs=2, Memory=3900MB) ...
* Preparing Kubernetes v1.28.3 on Docker 24.0.7 ...
  - Generating certificates and keys ...
  - Booting up control plane ...
  - Configuring RBAC rules ...
* Configuring bridge CNI (Container Networking Interface) ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: storage-provisioner, default-storageclass
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
PS C:\windows\system32>
```

5. Now here we can see that kubelet and api-server are running and kubeconfig is configured.

```
PS C:\windows\system32> minikube status
W0228 12:37:10.975128   25784 main.go:291] Unable to resolve the current Docker CLI context "
texts\meta\37a8eec1ce19687d132fe29051dca629d164e2c4958ba141d5f4133a33f0688f\meta.json: The sy
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured

PS C:\windows\system32>
```

6. Now to give alias to kubectl as “k” run the command : *New-Alias -Name “k” kubectl*

```
PS C:\Users\venis> New-Alias -Name "k" kubectl
```

7. Now lets try-out commands like “kubectl get pods” and “kubectl get nodes”.
   Here we can see that as we have not created any pods, we are getting no resources and there is one node which has been created and that is “minikube”.

```
PS C:\Users\venis> k get pods
No resources found in default namespace.
PS C:\Users\venis> k get nodes
NAME       STATUS   ROLES           AGE   VERSION
minikube   Ready    control-plane   26m   v1.28.3
PS C:\Users\venis>
```

Some other commands of kubectl:

```
PS C:\Users\venis> k get namespaces
NAME              STATUS   AGE
default           Active   77m
kube-node-lease   Active   77m
kube-public       Active   77m
kube-system       Active   77m
PS C:\Users\venis>
```

```
PS C:\Users\venis> k get services
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP   77m
PS C:\Users\venis>
```

```
PS C:\Users\venis> k get deployments
No resources found in default namespace.
```