Practice1: Simple pods operations

1 .Login to Azure and connect to your AKS cluster

```
PS /home/andrej> az account set --subscription 6e58e49b-f4f5-4019-9ff0-b815383a0405
PS /home/andrej> az aks get-credentials --resource-group andrejRSG1 --name andrejKC1
Merged "andrejKC1" as current context in /home/andrej/.kube/config
PS /home/andrej>
```

2. Check how many pods run under the default namespace

```
PS /home/andrej> kubectl get pods
No resources found in default namespace.
PS /home/andrej>
```

3. You should not see any pod under the default namespace. Now check all namespaces

```
PS /home/andrej> kubectl get pods --all-namespaces
NAMESPACE     NAME                                READY   STATUS    RESTARTS   AGE
kube-system   ama-logs-6jp6w                      2/2     Running   0          10m
kube-system   ama-logs-flvpk                      2/2     Running   0          11m
kube-system   ama-logs-rs-575ffbcd66-h8pbw        1/1     Running   0          11m
kube-system   ama-logs-zcjn8                      2/2     Running   0          11m
kube-system   azure-ip-masq-agent-c9w72           1/1     Running   0          11m
kube-system   azure-ip-masq-agent-fn8x8           1/1     Running   0          11m
kube-system   azure-ip-masq-agent-nz92v           1/1     Running   0          10m
kube-system   cloud-node-manager-kqnjx            1/1     Running   0          11m
kube-system   cloud-node-manager-mphrd            1/1     Running   0          10m
kube-system   cloud-node-manager-r8qnf            1/1     Running   0          11m
kube-system   coredns-59b6bf8b4f-kzwjg            1/1     Running   0          11m
kube-system   coredns-59b6bf8b4f-mw2rg            1/1     Running   0          10m
kube-system   coredns-autoscaler-64b6477b8b-xgf9w 1/1     Running   0          11m
kube-system   csi-azuredisk-node-56xpm            3/3     Running   0          11m
kube-system   csi-azuredisk-node-gpmg5            3/3     Running   0          10m
kube-system   csi-azuredisk-node-jmnxl            3/3     Running   0          11m
kube-system   csi-azurefile-node-5bt4b            3/3     Running   0          10m
kube-system   csi-azurefile-node-c8lw8            3/3     Running   0          11m
kube-system   csi-azurefile-node-gv6ng            3/3     Running   0          11m
kube-system   konnectivity-agent-689ddbbf4f-hqqhn 1/1     Running   0          11m
kube-system   konnectivity-agent-689ddbbf4f-prjxq 1/1     Running   0          11m
kube-system   kube-proxy-25cdz                    1/1     Running   0          11m
kube-system   kube-proxy-459vl                    1/1     Running   0          11m
```

4. How many pods do you see? Who deployed these pods? Why are they deployed?

- The kube-system namespace in Kubernetes is reserved for managing essential system components like the API server, etcd, kube-proxy, CoreDNS, and controller manager. It is automatically created and managed by the Kubernetes control plane or the tools used for cluster setup.

5. Now deploy you first pod using the imperative approach.

```
PS /home/andrej> kubectl run nginx --image=nginx
pod/nginx created
PS /home/andrej>
```

6. Validate if the pods has been created. What is the status of your pod?

```
PS /home/andrej> kubectl get pods
NAME    READY   STATUS    RESTARTS   AGE
nginx   1/1     Running   0          84s
PS /home/andrej>
```

7. Check the logs coming out of your pod.

```
PS /home/andrej> kubectl logs nginx
/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: 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
2023/04/09 16:20:20 [notice] 1#1: using the "epoll" event method
2023/04/09 16:20:20 [notice] 1#1: nginx/1.23.4
2023/04/09 16:20:20 [notice] 1#1: built by gcc 10.2.1 20210110 (Debian 10.2.1-6)
2023/04/09 16:20:20 [notice] 1#1: OS: Linux 5.4.0-1104-azure
2023/04/09 16:20:20 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2023/04/09 16:20:20 [notice] 1#1: start worker processes
2023/04/09 16:20:20 [notice] 1#1: start worker process 28
2023/04/09 16:20:20 [notice] 1#1: start worker process 29
PS /home/andrej>
```

8. Run following command to check current resource consumption of your pod: **kubectl top pod nginx**.

```
PS /home/andrej> kubectl top pod nginx
NAME    CPU(cores)   MEMORY(bytes)
nginx   0m           3Mi
PS /home/andrej>
```

9. Check on which Node your pods has been scheduled.

```
PS /home/andrej> kubectl get pods -o wide
NAME    READY   STATUS    RESTARTS   AGE    IP          NODE                                NOMINATED NODE   READINESS GATES
nginx   1/1     Running   0          7m3s   10.244.0.5   aks-agentpool-37021570-vmss000002   <none>           <none>
PS /home/andrej>
```

10. Try to find the same information but this time running **kubectl describe pod nginx**.

```
PS /home/andrej> kubectl describe pod nginx
Name:             nginx
Namespace:        default
Priority:         0
Service Account:  default
Node:             aks-agentpool-37021570-vmss000002/10.224.0.6
Start Time:       Sun, 09 Apr 2023 16:20:17 +0000
Labels:           run=nginx
Annotations:      <none>
Status:           Running
IP:               10.244.0.5
IPs:
  IP:  10.244.0.5
```

Terminal container button

11. Delete your pod using **kubectl delete pod nginx**

```
PS /home/andrej> kubectl delete pod nginx
pod "nginx" deleted
PS /home/andrej>
```

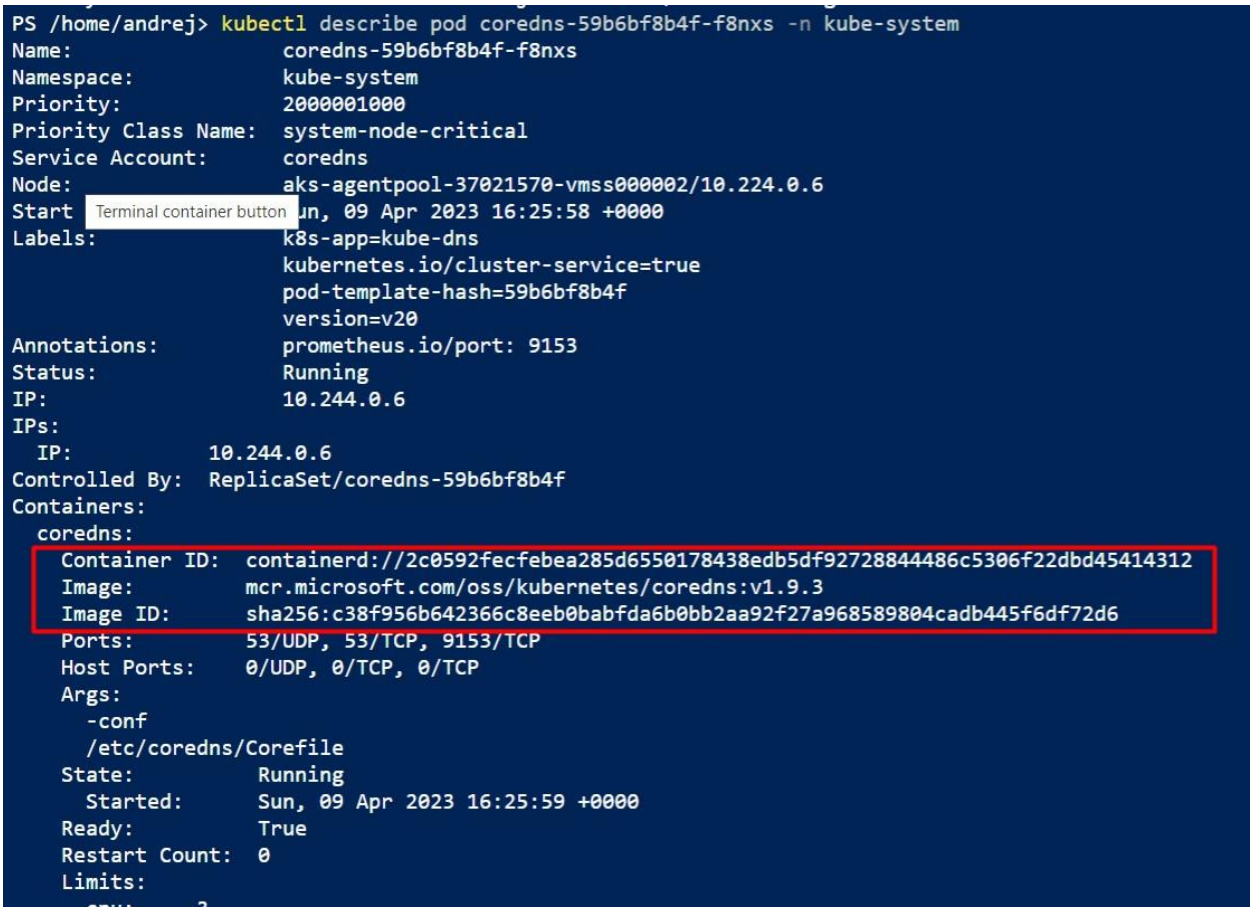
12. Let's find the image used on one of the **coredns** pods under **the kube-system** namespace.

```
PS /home/andrej> kubectl get pods --all-namespaces
NAMESPACE     NAME                                  READY   STATUS    RESTARTS   AGE
kube-system   ama-logs-rs-575ffbcd66-75z2q          1/1     Running   0          7m6s
kube-system   ama-logs-zcjn8                        2/2     Running   0          35m
kube-system   azure-ip-masq-agent-fn8x8             1/1     Running   0          35m
kube-system   cloud-node-manager-r8qnf              1/1     Running   0          35m
kube-system   coredns-59b6bf8b4f-f8nxs              1/1     Running   0          7m7s
kube-system   coredns-59b6bf8b4f-gfdhd              1/1     Running   0          6m57s
kube-system   coredns-autoscaler-64b6477b8b-h8n7c   1/1     Running   0          7m6s
kube-system   csi-azuredisk-node-jmnxl              3/3     Running   0          35m
kube-system   csi-azurefile-node-gv6ng              3/3     Running   0          35m
kube-system   konnectivity-agent-689ddbbf4f-4rjm5   1/1     Running   0          6m57s
kube-system   konnectivity-agent-689ddbbf4f-7xk6p   1/1     Running   0          7m7s
kube-system   kube-proxy-459vl                      1/1     Running   0          35m
kube-system   metrics-server-7dd74d8758-7s62d       2/2     Running   0          4m18s
kube-system   metrics-server-7dd74d8758-8jfl5       2/2     Running   0          4m18s
```

14. Note one of the **coredns** pods. Now run **kubectl describe pod -n kube-system**.

```
PS /home/andrej> kubectl describe pod coredns-59b6bf8b4f-f8nxs -n kube-system
Name:                 coredns-59b6bf8b4f-f8nxs
Namespace:            kube-system
Priority:             2000001000
Priority Class Name:  system-node-critical
Service Account:      coredns
Node:                 aks-agentpool-37021570-vmss000002/10.224.0.6
Start Time:           Sun, 09 Apr 2023 16:25:58 +0000
Labels:               k8s-app=kube-dns
                      kubernetes.io/cluster-service=true
                      pod-template-hash=59b6bf8b4f
                      version=v20
Annotations:          prometheus.io/port: 9153
Status:               Running
IP:                   10.244.0.6
IPs:
  IP:           10.244.0.6
Controlled By:  ReplicaSet/coredns-59b6bf8b4f
Containers:
  coredns:
    Container ID:  containerd://2c0592fecfebea285d6550178438edb5df92728844486c5306f22dbd45414312
    Image:         mcr.microsoft.com/oss/kubernetes/coredns:v1.9.3
    Image ID:      sha256:c38f956b642366c8eeb0babfda6b0bb2aa92f27a968589804cadb445f6df72d6
    Ports:         53/UDP, 53/TCP, 9153/TCP
    Host Ports:    0/UDP, 0/TCP, 0/TCP
```

15. Inspect the output and locate the image information.



16. Now let us check the logs of the metrics-server pod. Run the same command as in step 7