

## Lab 2 - (Module 1) - Implement Azure Kubernetes Service

### Objectives

- Understand how to deploy an Azure Kubernetes Service cluster
- Understand how to deploy pods into the Azure Kubernetes Service cluster
- Understand how to scale containerized workloads in an Azure Kubernetes service cluster

### Task 0: Installing Google Chrome

- ☐ 1. Select Start and open PowerShell as administrator. Run the following command:

```
Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]
```

- ☐ 2. Run the next PowerShell command and when prompted type **Y** to run the script.

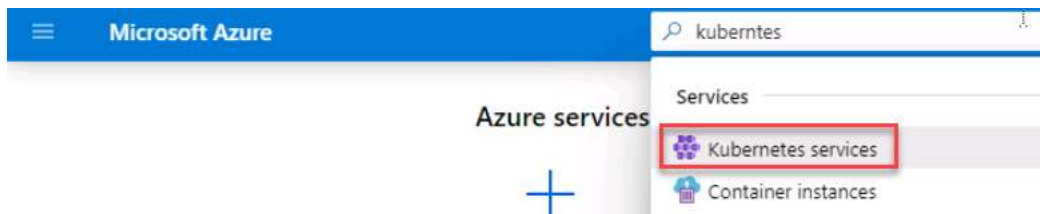
```
Choco install googlechrome
```

### Exercise 1: Deploy an Azure Kubernetes Service cluster

#### Task 1: Deploy an Azure Kubernetes Service cluster

In this task, you will deploy an Azure Kubernetes Services cluster by using the Azure portal.

- ☐ 1. Navigate to **Azure Portal** [portal.azure.com](https://portal.azure.com) and login with the username [sheikhnasir5RKFG@gdcs1.com](mailto:sheikhnasir5RKFG@gdcs1.com) and password [PYBV0xuK0D5r3Ttr](#)
- ☐ 2. In the **Azure portal**, search for and select [Kubernetes services](#) and then, on the **Kubernetes services** blade, click + **Create** > **Create a Kubernetes cluster**.



- ☐ 3. On the **Basics** tab of the **Create Kubernetes cluster** blade, specify the following settings (leave others with their default values):
- Resource group: **myResourceGroup-DVOMGA1TUZ**
  - Kubernetes cluster name: [myAKSCluster](#)
  - Region: **East US**
  - Kubernetes version: **Accept the default**
  - Node size: **Standard DS2 v2**
  - Scale Method: **Manual**
  - Node count: **1**

**Note:** The default node size should be **Standard DS2 v2**. If it is not, update the node size to **Standard DS2 v2**. The deployment will fail if the node size is not changed.

- ☐ 4. Click **Next: Node Pools** > and, on the **Node Pools** tab of the **Create Kubernetes cluster** blade, view the settings and leave with their default values.
- ☐ 5. Click **Next: Authentication** > and, on the **Authentication** tab of the **Create Kubernetes cluster** blade, specify the following settings (leave others with their default values):
- Authentication method: **System-assigned managed identity**
  - Role based access control (RBAC): **Enabled**
- ☐ 6. Click **Next: Networking** > and, on the **Networking** tab of the **Create Kubernetes cluster** blade, specify the following settings (leave others with their default values):
- Network configuration: **Kubenet**

- ☐ 7. Click **Next: Integrations** >, on the **Integrations** tab of the **Create Kubernetes cluster** blade, set **Container monitoring** to **Disabled**, click **Review + create** and then click **Create**.

**Note:** In production scenarios, you would want to enable monitoring. Monitoring is disabled in this case since it is not covered in the lab.

**Note:** Wait for the deployment to complete. This should take about 10 minutes.

✓ **Summary** In this exercise, you created a single node Kubernetes cluster using the Azure portal.

## Exercise 2: Deploy pods into the Azure Kubernetes Service cluster

### Overview

In this exercise, you will deploy a pod into the Azure Kubernetes Service cluster.

### Task 1: Deploy pods into the Azure Kubernetes Service cluster

- ☐ 1. On the deployment blade, click the **Go to resource** link.

Delete Cancel Redeploy Refresh

---

**Your deployment is complete**

Deployment name: microsoft.aks-  
Subscription:   
Resource group: myResourceGroup

Start time: 3/25/2020, 1:10:50 PM  
Correlation ID:

▼ Deployment details [\(Download\)](#)

^ Next steps

[Go to resource](#)

- ☐ 2. On the **myAKSCluster Kubernetes service** blade, in the **Settings** section, click **Node pools**.
- ☐ 3. On the **myAKSCluster - Node pools** blade, verify that the cluster consists of a single pool with one node.

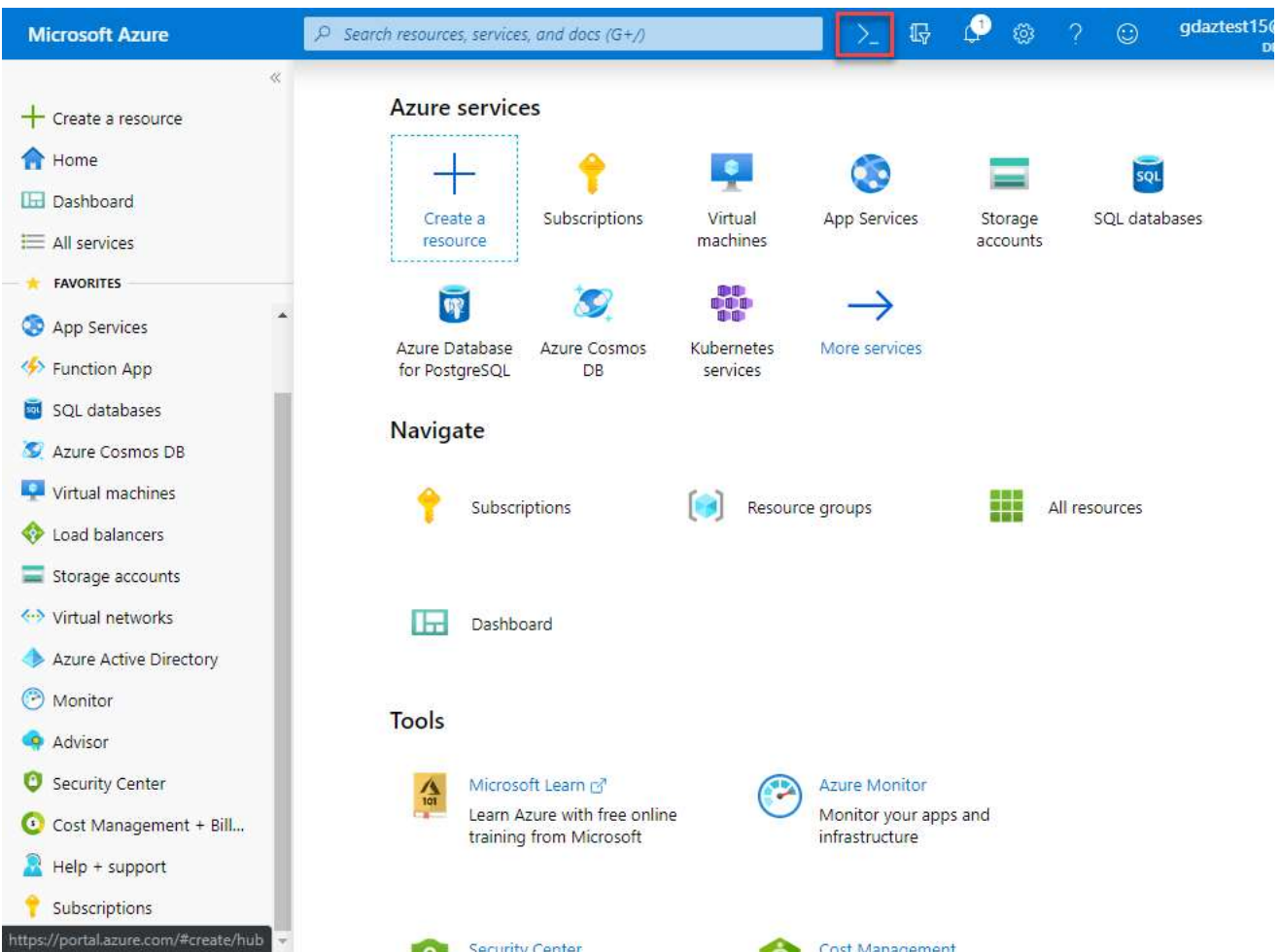
Add node pool Refresh

---


You can add node pools of different types to your cluster to handle a variety of workloads, scale and upgrade your existing node pools, or delete node pools that you no longer need. [Learn more about multiple node pools](#)

Name	Provisioning state	Kubernetes version	OS type	Node count
agentpool	Succeeded	1.15.10	Linux	1

- ☐ 4. Select **Cloud Shell** from the **Azure Portal** tool bar.



☐ 5. Select **Bash** on the Welcome screen.

- ☐ 6. In the **You have no storage mounted** pane, click **Show advanced settings**, perform the following tasks:
- Leave the **Subscription** drop-down list entry set to its default value.
  - In the **Cloud Shell region** drop-down list, select the Azure region matching or near the location where you intend to deploy resources in this exercise.
  - In the **Resource group** section, select the Resource Group that has been created for you.
  - In the **Storage account** section, ensure that the **Create new** option is selected and then, in the text box below, type a unique name consisting of a combination of between 3 and 24 characters and digits.
  - In the **File share** section, ensure that the **Create new** option is selected and then, in the text box below, type  **cloudshell**.
  - Click the **Create storage** button.

☐ 7. Wait for the **Cloud Shell** to finish its first-time setup procedures before you proceed to the next task.

☐ 8. From the **Cloud Shell** pane, run the following to retrieve the credentials to access the AKS cluster:

```
Resource_group=$(az group list --query [].name --output tsv)
AKS_CLUSTER='myAKSCluster'
az aks get-credentials --resource-group $Resource_group --name $AKS_CLUSTER
```

☐ 9. From the **Cloud Shell** pane, run the following to verify connectivity to the AKS cluster:

```
kubectl get nodes
```

☐ 10. In the **Cloud Shell** pane, review the output and verify that the one node which the cluster consists of at this point is reporting the **Ready** status.

☐ 11. From the **Cloud Shell** pane, run the following to deploy the **nginx** image from the Docker Hub:

```
kubectl create deployment nginx-deployment --image=nginx
```

 **Note:** Make sure to use lower case letters when typing the name of the deployment (nginx-deployment)

- ☐ 12. From the **Cloud Shell** pane, run the following to verify that a Kubernetes pod has been created:

```
kubectl get pods
```

- ☐ 13. From the **Cloud Shell** pane, run the following to identify the state of the deployment:

```
kubectl get deployment
```

- ☐ 14. From the **Cloud Shell** pane, run the following to make the pod available from Internet:

```
kubectl expose deployment nginx-deployment --port=80 --type=LoadBalancer
```

- ☐ 15. From the **Cloud Shell** pane, run the following to identify whether a public IP address has been provisioned:

```
kubectl get service
```

- ☐ 16. Re-run the command until the value in the **EXTERNAL-IP** column for the **nginx-deployment** entry changes from **<none>** to a public IP address. Note the public IP address in the **EXTERNAL-IP** column for **nginx-deployment**.

```
demo10qm0r@Azure:~$ kubectl get service
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
kubernetes	ClusterIP	10.0.0.1	<none>	443/TCP	9m53s
nginx-deployment	LoadBalancer	10.0.199.195	52.170.169.188	80:31393/TCP	66s

- ☐ 17. Open a browser window and navigate to the IP address you obtained in the previous step. Verify that the browser page displays the **Welcome to nginx!** message.

## Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](http://nginx.org).  
Commercial support is available at [nginx.com](http://nginx.com).

*Thank you for using nginx.*

### ✓ Summary

In this exercise, you deployed a nginx pod into the Azure Kubernetes Service cluster and verified the pod was running

### Exercise 3: Scale containerized workloads in the Azure Kubernetes service cluster

#### ? Overview

In this exercise, you will scale horizontally the number of pods and then the number of cluster nodes.

#### Task 1: Scale containerized workloads in the Azure Kubernetes service cluster

- ☐ 1. From the **Cloud Shell** pane, run the following to scale the deployment by increasing of the number of pods to 2:

```
kubectl scale --replicas=2 deployment/nginx-deployment
```

- ☐ 2. From the **Cloud Shell** pane, run the following to verify the outcome of scaling the deployment:

```
kubectl get pods
```

```
demo10qm0r@Azure:~$ kubectl get pods
```

NAME	READY	STATUS	RESTARTS	AGE
nginx-deployment-6f77f65499-jxvbm	1/1	Running	0	3m34s
nginx-deployment-6f77f65499-wt9n6	1/1	Running	0	5m22s

**⚠ Note:** Review the output of the command and verify that the number of pods increased to 2.

- ☐ 3. From the **Cloud Shell** pane, run the following to scale out the cluster by increasing the number of nodes to 2:

```
az aks scale --resource-group $Resource_group --name $AKS_CLUSTER --node-count 2
```

**Note:** Wait for the provisioning of the additional node to complete. This will take about 3 minutes. If it fails, rerun the `az aks scale` command.

**Note:** Do not scale the cluster beyond two nodes. This will stop your lab!

- ☐ 4. From the **Cloud Shell** pane, run the following to verify the outcome of scaling the cluster:

```
kubectl get nodes
```

```
demo10qm0r@Azure:~$ kubectl get nodes
NAME                                STATUS    ROLES    AGE      VERSION
aks-agentpool-19694923-vmss000000  Ready    agent    13m      v1.15.11
aks-agentpool-19694923-vmss000001  Ready    agent    2m29s    v1.15.11
```

**Note:** Review the output of the command and verify that the number of nodes increased to 2.

- ☐ 5. From the **Cloud Shell** pane, run the following to scale the deployment:

```
kubectl scale --replicas=10 deployment/nginx-deployment
```

- ☐ 6. From the **Cloud Shell** pane, run the following to verify the outcome of scaling the deployment:

```
kubectl get pods
```

```
demo10qm0r@Azure:~$ kubectl get pods
NAME                                READY    STATUS              RESTARTS   AGE
nginx-deployment-6f77f65499-75dxr  0/1      ContainerCreating   0           4s
nginx-deployment-6f77f65499-9n2n6  1/1      Running             0           4s
nginx-deployment-6f77f65499-cqkhq  0/1      ContainerCreating   0           4s
nginx-deployment-6f77f65499-hp2ts  0/1      ContainerCreating   0           4s
nginx-deployment-6f77f65499-jxvbm  1/1      Running             0           6m8s
nginx-deployment-6f77f65499-mf4pt  0/1      ContainerCreating   0           4s
nginx-deployment-6f77f65499-rbtgf  0/1      ContainerCreating   0           4s
nginx-deployment-6f77f65499-sgkxz  1/1      Running             0           4s
nginx-deployment-6f77f65499-vj69n  1/1      Running             0           4s
nginx-deployment-6f77f65499-wt9n6  1/1      Running             0           7m56s
```

**Note:** Review the output of the command and verify that the number of pods increased to 10.

- ☐ 7. From the **Cloud Shell** pane, run the following to review the pods distribution across cluster nodes:

```
kubectl get pod -o=custom-columns=NODE:.spec.nodeName,POD:.metadata.name
```

```
demo10qm0r@Azure:~$ kubectl get pod -o=custom-columns=NODE:.spec.nodeName,POD:.metadata.name
NODE                                POD
aks-agentpool-19694923-vmss000001  nginx-deployment-6f77f65499-75dxr
aks-agentpool-19694923-vmss000000  nginx-deployment-6f77f65499-9n2n6
aks-agentpool-19694923-vmss000001  nginx-deployment-6f77f65499-cqkhq
aks-agentpool-19694923-vmss000001  nginx-deployment-6f77f65499-hp2ts
aks-agentpool-19694923-vmss000000  nginx-deployment-6f77f65499-jxvbm
aks-agentpool-19694923-vmss000001  nginx-deployment-6f77f65499-mf4pt
aks-agentpool-19694923-vmss000001  nginx-deployment-6f77f65499-rbtgf
aks-agentpool-19694923-vmss000000  nginx-deployment-6f77f65499-sgkxz
aks-agentpool-19694923-vmss000000  nginx-deployment-6f77f65499-vj69n
aks-agentpool-19694923-vmss000000  nginx-deployment-6f77f65499-wt9n6
```

**Note:** Review the output of the command and verify that the pods are distributed across both nodes.

- ☐ 8. From the **Cloud Shell** pane, run the following to delete the deployment:

```
kubectl delete deployment nginx-deployment
```

- ☐ 9. Close the **Cloud Shell** pane.

### ✓ Summary

In this exercise, you scaled the number of pods and then the number of cluster nodes horizontally within your Kubernetes cluster.