|  |
| --- |
| **Azure Kubernetes Service with Azure DevOps and Terraform**  **Duration: 10 Days (2 Week)** |
|  |
| 1. Create Azure AKS Cluster using Azure Portal |
| 2. Docker Fundamentals |
| 3. Imperative Method: Kubernetes Fundamentals using kubectl |
| 4. Declarative Method: Kubernetes Fundamentals using YAML |
| 5. Azure Disks for AKS Storage |
| 6. Custom Storage Class, PVC, and PV |
| 7. AKS default Storage class, PVC and PV |
| 8. User Management Web Application Deployment with MySQL as storage using Storage Class, PVC, and PV |
| 9. Azure MySQL for AKS Storage |
| 10. Kubernetes Secrets |
| 11. Azure Files for AKS Storage |
| 12. Ingress Basics |
| 13. Ingress Context path based Routing |
| 14. Azure DNS Zones - Delegate domain from AWS to Azure |
| 15. Ingress and External DNS with Azure DNS Zones |
| 16. Ingress Domain Name based Routing with External DNS |
| 17. Ingress SSL with LetsEncrypt |
| 18. Kubernetes Requests & Limits |
| 19. Kubernetes Namespaces |
| 20. Kubernetes Namespaces - Imperative |
| 21. Kubernetes Namespaces - Limit Range |
| 22. Kubernetes Namespaces - Resource Quota |
| 23. Azure Virtual Nodes for AKS |
| 24. Azure Virtual Nodes Basics |
| 25. Azure AKS Virtual Nodes Mixed Mode Deployments |
| 26. Azure Container Registry for AKS |
| 27. Integrate Azure Container Registry ACR with AKS |
| 28. Azure AKS Pull Docker Images from ACR using Service Principal |
| 29. Pull Docker Images from ACR using Service Principal and Run on Azure Virtual Nodes |
| 30. Azure DevOps with AKS Cluster |
| 31. Azure DevOps - Build and Push Docker Image to Azure Container Registry |
| 32. Azure DevOps - Build, Push to ACR and Deploy to AKS |
| 33. Azure DevOps - Create Starter Pipeline |
| 34. Azure DevOps - Release Pipelines |
| 35. Azure AKS - Enable HTTP Application Routing AddOn |
| 36. Azure AKS Authentication with Azure AD and Kubernetes RBAC |
| 37. Azure AKS Cluster Access with Multiple Clusters |
| 38. Azure AD Integration with Azure AKS for Authentication |
| 39. Kubernetes RBAC Role & Role Binding with Azure AD on AKS |
| 40. Kubernetes RBAC Cluster Role & Role Binding with AD on AKS |
| 41. Azure AKS Cluster Autoscaling |
| 42. Azure AKS - Cluster Autoscaler |
| 43. Azure AKS - Horizontal Pod Autoscaler HPA |
| 44. Azure AKS Production Grade Cluster Design using AZ AKS CLI |
| 45. Create Azure AKS Cluster using AZ AKS CLI |
| 46. Create Azure AKS Linux, Windows, and Virtual Node Pools |
| 47. Deploy Apps to Azure AKS Linux, Windows, and Virtual Node Pools |
| 48. Provision Azure AKS Clusters using Terraform |
| 49. Terraform Command Basics |
| 50. Terraform Language Basics |
| 51. Provision AKS Cluster using Terraform |
| 52. Create AKS Cluster Linux and Windows Node Pools |
| 53. Create an Azure AKS Cluster using Custom Virtual Network |
| 54. Provision Azure AKS using Terraform & Azure DevOps |
|  |
| **Azure Services Covered** |
| 1. Azure Kubernetes Service |
| 2. Azure Disks |
| 3. Azure Files |
| 4. Azure MySQL Database |
| 5. Azure Storage Accounts |
| 6. Azure Cloud Shell |
| 7. Azure Load Balancer |
| 8. Azure DNS Zones |
| 9. Azure Container Registries ACR |
| 10. Azure Container Registries ACR with Azure Service Principal |
| 11. Azure DevOps - Build Pipelines with ACR & Github Repositories |
| 12. Azure DevOps - Release Pipelines with AKS |
| 13. Azure Public IP Address |
| 14. Azure Standard Load Balancer |
| 15. Azure Virtual Networks |
| 16. Azure Active Directory |
| 17. Azure Container Instances - Virtual Nodes |
| 18. Azure AKS Windows and Linux User NodePools |
| 19. Azure Managed Service Identity - MSI |
| 20. Azure Virtual Machine Scale Sets |
| 21. Azure Log Analytics Workspaces for Azure Monitor |
|  |
|  |
| **Kubernetes Concepts Covered** |
| 1. Kubernetes Architecture |
| 2. Pods |
| 3. ReplicaSets |
| 4. Deployments |
| 5. Services - Load Balancer Service |
| 6. Services - Cluster IP Service |
| 7. Services - External Name Service |
| 8. Services - Ingress Service |
| 9. Services - Ingress SSL & SSL Redirect |
| 10. Services - Ingress & External DNS |
| 11. Services - Domain Name based Routing |
| 12. Imperative - with kubectl |
| 13. Declarative - Declarative with YAML |
| 14. Secrets |
| 15. Init Containers |
| 16. Requests & Limits |
| 17. Namespaces - Imperative |
| 18. Namespaces - Limit Range |
| 19. Namespaces - Resource Quota |
| 20. Storage Classes |
| 21. Persistent Volumes |
| 22. Persistent Volume Claims |
| 23. Services - Load Balancers |
| 24. Annotations |
| 25. HPA - Horizontal Pod Autoscaler |
| 26. CA - Cluster Autoscaler |
| 27. Config Maps |
| 28. RBAC - Role & Role Bindings |
| 29. RBAC - Cluster Role & Cluster Role Bindings |
| 30. Virtual Kubelet |
| 31. Secrets - Image Pull Secrets |

az login

Resource Groupname: testvm16may

AKS Cluster name : Cluster29June

az aks get-credentials --resource-group testvm16may --name Cluster29June

Demo Steps:

# Template

az aks get-credentials --resource-group <Resource-Group-Name> --name <Cluster-Name>

# Replace Resource Group & Cluster Name

az aks get-credentials --resource-group aks-rg1 --name aksdemo1

# List Kubernetes Worker Nodes

kubectl get nodes

kubectl get nodes -o wide

* apiVersion - Which version of the Kubernetes API you're using to create this object
* kind - What kind of object you want to create
* metadata - Data that helps uniquely identify the object, including a name string, UID, and optional namespace
* spec - What state you desire for the object

**apiVersion**: apps/v1

**kind**: Deployment

**metadata**:

**name**: nginx-deployment

**spec**:

**selector**:

**matchLabels**:

**app**: nginx

**replicas**: 2 *# tells deployment to run 2 pods matching the template*

**template**:

**metadata**:

**labels**:

**app**: nginx

**spec**:

**containers**:

- **name**: nginx

**image**: nginx:1.14.2

**ports**:

- **containerPort**: 80

<https://kubernetes.io/docs/concepts/overview/components/>



