**Module 1**

**Topics to be covered**

**Common Services:**

Virtual machines

App service

MySQL

SQL

Cosmos DB

storage account

Application Gateway / Azure firewall

Vnet/ subnet

**Best practice Exercises:**

**Exercise1:**

1. Create a Vnet with local IP range (192.168.0.0/16) and 2 subnets.
2. Create a test resource group
3. Create a VM in central India region with B1Ms with Ubuntu OS with Public IP in one of the subnet
4. Attach Data disk to the VM 20 GB.
5. Install apache webserver in the VM (you can get the steps : <https://ubuntu.com/tutorials/install-and-configure-apache#1-overview>)
6. Access the site successfully from BCN using public IP of VM
7. Restrict access to the VM using NSG and allow only 22 and 80 Port.

**Exercise2:**

1. Create test storage account
2. Create blob container and file share container in storage account
3. Try copying data to the storage account via storage explorer. (You can use the access keys created with the storage account)

**Exercise3**

1. Create a Basic app service plan and Basic low config app service
2. Deploy basic hello world web page
3. Ref: <https://docs.microsoft.com/en-us/azure/developer/java/toolkit-for-eclipse/create-hello-world-web-app>)

Please complete AZ-900 course([Microsoft Certified: Azure Fundamentals - Learn | Microsoft Docs](https://docs.microsoft.com/en-us/learn/certifications/azure-fundamentals/)) and  you can use the link <https://docs.microsoft.com/en-us/learn/azure/> to learn.

Once we are through with the Azure portal and components we can move on to terraform and devops. Please find below link to create Azure devops test account.

<https://docs.microsoft.com/en-us/azure/devops/user-guide/sign-up-invite-teammates?view=azure-devops>

**Module 2**

**Topics to be covered**

**Session 1:**

Clarified the doubts on the key generation and access of VM and deploying the Apache server

Demo1: Generated Keys and converting .pem file to .ppk file and login to VM with key

              Deployed apache webserver and accessed it with public IP of VM.

**Session 2:**

Today we have started terraform and the below topics are covered. Please continue using terraform to perform other component deployments via terraform.

1. What is terraform
2. Need for terraform
3. Terraform declarative language
4. Terraform pre-requisites (Azure CLI, AZ login, set subscription)
5. Terraform file types and use (.tf and .tfvars)
6. Terraform flow (init, plan and apply)
7. Terraform sample scripts.
8. Sample codes on hashicorp site.

**Demo:**

We have covered Demo on terraform on basic deployments and also showed deploying resource group via terraform.

**Links to learn:**

<https://learn.hashicorp.com/terraform>

<https://learn.hashicorp.com/collections/terraform/azure-get-started>

<https://registry.terraform.io/>

**Module 3**

**Topics to be covered**

Azure Application Gateway:

1. Introduction, purpose, use-case
2. Front end - listener
3. Backend pools (backend VMs)
4. http settings
5. routing rules

Links for reference:

<https://aidanfinn.com/?p=21566>

<https://docs.microsoft.com/en-us/azure/application-gateway/>

<https://docs.microsoft.com/en-us/azure/application-gateway/quick-create-portal>

<https://docs.microsoft.com/en-us/azure/application-gateway/configuration-overview>

Azure Kubernetes Services (Basic concept and overview with small demo):

1. Introduction
2. Basic concepts like Docker
3. Configuration like nodes, size
4. PODs, nodes
5. Advantages over traditional VM based deployments

Links for reference:

<https://kubernetes.io/docs/tutorials/>

<https://kubernetes.io/docs/tutorials/kubernetes-basics/>

<https://docs.microsoft.com/en-us/azure/aks/>

<https://docs.microsoft.com/en-us/azure/aks/kubernetes-walkthrough-portal>

**Best practice Exercises:**

**Exercise1: (terraform based deployment)**

1. Create a Vnet with ip range 10.0.0/16
2. Create 2 subnets splitting the vnet in to 2
3. Create webserver in subnet1 with 2 vCPu and 4 RAM
4. Create NSG to allow only port 22 and 80
5. Access the application
6. Restrict port 80 and try accessing
7. Remove restriction and resize the VM to 1 vCPU and 2 GB RAM
8. Access the web server

**Exercise2: (Application gateway)**

1. Create a web server
2. Create application gateway
3. Create front end listener for port 80
4. Add the webserver to the backend pool of application gateway
5. Create http setting and routing rule to redirect traffic from application gateway to webserver
6. Access the webpage via application gateway IP

**Exercise3**

1. Deploy a basic AKS deployment with 1 node with less size
2. Deploy a basic hello world app in kubernetes as POD
3. Try accessing the application from internet

<https://kubernetes.io/docs/tutorials/kubernetes-basics/> - deploy an app section

<https://medium.com/@m.sedrowski/run-your-first-application-on-kubernetes-e54d5194e84b>

**Module 4**

**Topics to be covered**

1. CI/CD introduction
2. Need for devops
3. Azure Devops introduction
4. Azure Devops hierarchy (org à project à repos, boards, test plans, pipe lines, artifact)
5. Billing for Azure devops
6. Users creation and adding permissions in Org level, project level and repo level
7. Azure Repos (Branches, Commits, Pushes, Pull request, Tags)
8. Azure Pipelines
9. Azure Boards
10. Azure Test Plans
11. Azure Artifacts

**Reference Links:**

<https://azure.microsoft.com/en-us/services/devops/> (have sections for all the sub topics)

<https://docs.microsoft.com/en-us/azure/devops/user-guide/sign-up-invite-teammates?view=azure-devops> (test account creation)

**Exercise 1:**

1. Create a test organization name “TestORG”
2. Create 2 projects “project1” and “Project2”
3. Create 2 repos “testrepo1” and “testrepo2”
4. Create users (you can use each other’s ID and try accessing others project repos)
5. Try restricting the access to only repo and check the access

**Exercise 2:**

1. Create a test pipeline
2. Add a code to repo to trigger the pipeline
3. Validate the output

**Exercise 3:**

1. Create sample tasks using azure boards
2. Use each other’s ID to assign and update tasks
3. Validate and Track the task updates.