**Challenge 1:**

**Azure web app as Presentation Layer**

az login

az appservice plan create --name DemoAppPlan --resource-group SampleRSG --sku B1 --is-linux

az webapp create --name SampleAppService --resource-group SampleRSG --plan DemoAppPlan

**Web service as Logic Layer**

az webapp deployment source config-zip --src path/to/code.zip --name SampleAppService --resource-group myResourceGroup

az webapp start --name SampleAppService --resource-group myResourceGroup

az login

az group create --name SampleRSG --location westus

az vm create --resource-group SampleRSG --name myVM --image UbuntuLTS --admin-username azureuser --generate-ssh-keys

**Azure SQL as Data Layer**

az login

az group create --name SampleRSG --location westus

az sql server create --name DemoSQLServer --resource-group SampleRSG --location westus --admin-user myAdmin --admin-password myPassword

az sql db create --resource-group SampleRSG --server DemoSQLServer --name SampleDB --service-objective S0

**Configure web app to services**

az webapp config appsettings set --name SampleAppService --resource-group SampleRSG --settings WEBSITE\_LOAD\_BALANCER\_URL=http://myWebService:80

**Configure service to Azure SQL**

az webapp config connection-string set --name myWebService --resource-group SampleRSG --settings MyDbConnection="Server=tcp:DemoSQLServer.database.windows.net,1433;Initial Catalog= SampleDB;PersistSecurityInfo=False;User ID=myAdmin;Password=myPassword;MultipleActiveResultSets=False;Encrypt=True;TrustServerCertificate=False;Connection Timeout=30;" --connection-string-type SQLAzure

**Secure Communications**

az apim create --name myAPIManagement --resource-group SampleRSG --location eastus

az apim operation create --name getData --resource-group SampleRSG --api-management-name myAPIManagement --api-name myAPI --method get --url-template /api/data

az apim operation create --name getData --resource-group SampleRSG --api-management-name myAPIManagement --api-name myAPI --method get --url-template /api/data

az apim api set-authentication --name myAPI --resource-group SampleRSG --api-management-name myAPIManagement --authentication-type AAD --aad-tenant-id <your-tenant-id> --aad-client-id <your-client-id> --aad-secret <your-secret>

az apim api set-policies --name myAPI --resource-group SampleRSG --api-management-name myAPIManagement --policy '{ "rate-limit": { "rate": 10, "unit": "second" } }'

az apim api publish --name myAPI --resource-group SampleRSG --api-management-name myAPIManagement

az network vnet create --name myVnet --resource-group SampleRSG --subnet-name mySubnet

az webapp vnet-integration add -g SampleRSG -n SampleAppService --vnet myVnet --subnet mySubnet

az network vnet peering create --name myVnetPeering --resource-group SampleRSG --vnet-name myVnet --remote-vnet myRemoteVnet --allow-vnet-access

az network nsg create --name myNSG --resource-group SampleRSG

az network firewall create -g SampleRSG -n myFirewall

**Monitoring**

az resource create --name myLogAnalyticsWorkspace --resource-group SampleRSG --location eastus --resource-type Microsoft.OperationalInsights/workspaces

az webapp log config --name SampleAppService --resource-group SampleRSG --log-analytics-workspace myLogAnalyticsWorkspace

az monitor metrics alert create --resource SampleAppService --condition "Requests/sec > 10" --name myAlert --resource-group SampleRSG

az monitor log-analytics query --workspace myLogAnalyticsWorkspace --query "requests | summarize count() by bin(timestamp, 1h)"

az webapp log config --name SampleAppService --resource-group SampleRSG --log-analytics-workspace myLogAnalyticsWorkspace

az monitor metrics alert create --resource SampleAppService --condition "Requests/sec > 10" --name myAlert --resource-group SampleRSG

az monitor log-analytics query --workspace myLogAnalyticsWorkspace --query "requests | summarize count() by bin(timestamp, 1h)"

**Challenge 2:**

**To get the metadata of the Azure instance**

az vm show --name myVm --resource-group SampleRSG --query "{name: name, location: location, tags: tags, type: type, provisioningState: provisioningState}" --out json

**C# Challenge 3:**

public static string GetValuesByKey(string json, string key)

{

var jObject = JSON.Parse(json);

var values = jObject.SelectTokens(key).Select(token => token.Value<string>());

return String.Join(",", values);

}