Challenge #1 Ans

A three-tier architecture is a software architecture pattern where the application is broken down into three logical tiers: the presentation layer, the business/application logic layer, and the database layer. This architecture is used in a client-server application such as a web application that has the frontend, the backend, and the database. Each of these layers or tiers does a specific task and can be managed independently of each other.

**Presentation Layer:** The presentation layer is responsible for user interaction and provides the user interface.

**Application Layer:** The application layer contains the business logic and processes user requests received from the presentation layer.

Azure VM or Virtual Machine Scale Set

**Data Layer:** The data layer stores and manages the application’s data. It can include databases, file storage, or other data storage mechanisms.

Azure SQL database

Azure is a cloud platform that provides different cloud computing services.

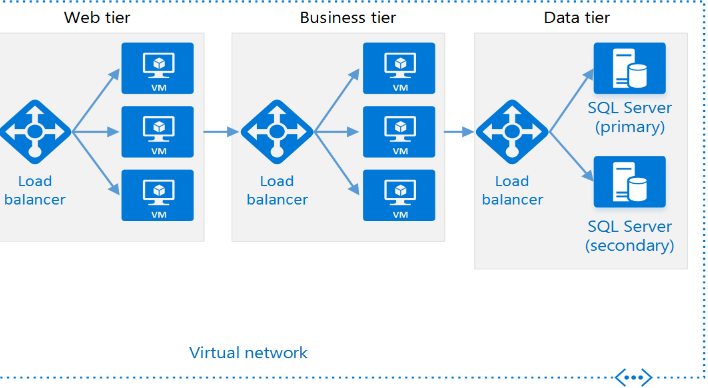
We use the following Azure services to design and build a three-tier cloud infrastructure

VNet

Azure VM

Load Balancer

SQL Server

Internet 

Architecture Diagram

Challenge #2 Ans

provider "azurerm" {

features {}

}

data "azurerm\_virtual\_machine" "example" {

name = "my-vm"

resource\_group\_name = "RG1"

}

output "instance\_metadata" {

value = jsonencode({

vm\_id = data.azurerm\_virtual\_machine.example.id

vm\_name = data.azurerm\_virtual\_machine.example.name

location = data.azurerm\_virtual\_machine.example.location

resource\_group = data.azurerm\_virtual\_machine.example.resource\_group\_name

tags = data.azurerm\_virtual\_machine.example.tags

})

}

Challenge #3 Ans

def get\_value(obj, key):

keys = key.split('/')

value = obj

try:

for k in keys:

value = value[k]

return value

except (KeyError, TypeError):

return None

#Example usage

object = {"a": {"b": {"c": "d"}}}

key = "a/b/c"

value = get\_value(object, key)

print(value) # Output: d