**Data Sources**

Data sources in Terraform are used to get information about resources external to Terraform and use them to set up your Terraform resources. For example, a list of IP addresses a cloud provider exposes.

Each provider may offer data sources alongside its set of resource types.

**Using Data Sources**

A data source is accessed via a special kind of resource known as a *data resource*, declared using a data block:

**Note:** While managed resources cause Terraform to create, update, and delete infrastructure objects, data resources cause Terraform only to **read** objects.

**Creating and Storage Account in Existing Resource Group Demo-rg:**

#Fetch Existing Resource Group Information

**data** "azurerm\_resource\_group" "DemoRG" {

  name = "Demo-rg"

}

resource "azurerm\_storage\_account" "example" {

  name                     = "sandeepstorageacc123"

  resource\_group\_name     = **data.azurerm\_resource\_group.DemoRG.name**

  location                 = **data.azurerm\_resource\_group.DemoRG.location**

  account\_tier             = "Standard"

  account\_replication\_type = "GRS"

}

**Assignment:**

Update your configuration files to create VM in existing Resource Group and VNet (manually created in portal)

**data** "azurerm\_virtual\_network" "my" {

  name                = "Demo-vnet"

  resource\_group\_name = "Demo-rg"

}

**data** "azurerm\_subnet" "my" {

  name                 = "default"

  virtual\_network\_name = data.azurerm\_virtual\_network.my.name

  resource\_group\_name  = data.azurerm\_resource\_group.DemoRG.name

}

# Create network interface

resource "azurerm\_network\_interface" "myNIC" {

  count               = 2

  name                = "myNIC${count.index}"

  location            = data.azurerm\_resource\_group.DemoRG.location

  resource\_group\_name = data.azurerm\_resource\_group.DemoRG.name

  ip\_configuration {

    name                          = "myNIC"

    subnet\_id                     = data.azurerm\_subnet.my.id

    private\_ip\_address\_allocation = "Dynamic"

    public\_ip\_address\_id          = azurerm\_public\_ip.myVMPublicIP[count.index].id

  }

}

**Terraform Commands**

* terraform init
* terraform validate
* terraform plan -out=tfplan
* terraform apply tfplan
* terraform destroy --auto-approve

**Assignment:**

Update your configuration files to create Azure App Service in existing App Service Plan (manually created in portal)