**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:**

**D:\Data Example>Main.tf**

terraform {

   required\_providers {

    azurerm = {

      source = "hashicorp/azurerm"

      version = "3.73.0"

    }

    tls = {

      source = "hashicorp/tls"

      version = "4.0.4"

    }

    local = {

      source = "hashicorp/local"

      version = "2.4.0"

    }

  }

}

provider "azurerm" {

    subscription\_id = "<your subscription here>"

    client\_id = "c259f715-168c-4ad5-a3b5-7a3f4a290af7"

    client\_secret = "oB68Q~eLq\_QfyzNOmwCxEG~I9fqNYhayL"

    tenant\_id = "82d8af3b-d3f9-465c-b724-0fb186cc28c7"

    features {}

}

**D:\Data Example>ResourceGroup.tf**

#Fetch Existing Resource Group Information

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

  name = "Demo-rg"

}

**D:\Data Example>StorageAccount.tf**

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"

}

**Example2: Create an App Service in already existing Resource Group (Sandeep-terraform-rg) and App Service Plan (DemoAppService-plan) which are created using Portal**

**D:\Data Example>AppService.tf**

# resource "azurerm\_service\_plan" "demo-appservice-plan" {

#   name                = "sandeep-demo-appservice-plan"

#   location            = azurerm\_resource\_group.terraform-rg.location

#   resource\_group\_name = azurerm\_resource\_group.terraform-rg.name

#   os\_type             = "Linux"

#   sku\_name         = "B2"

# }

data "azurerm\_service\_plan" "demo-appservice-plan" {

  name                = "DemoAppService-plan"

  resource\_group\_name = data.azurerm\_resource\_group.terraform-rg.name

}

resource "azurerm\_linux\_web\_app" "demo-appservice" {

  name                = "sandeep-demo-appservice"

  location            = data.azurerm\_resource\_group.terraform-rg.location

  resource\_group\_name = data.azurerm\_resource\_group.terraform-rg.name

  service\_plan\_id = data.azurerm\_service\_plan.demo-appservice-plan.id

  site\_config {

    application\_stack {

      dotnet\_version = "7.0"

    }

    always\_on = false

  }

  app\_settings = {

    "Setting1" = "Value1"

    "Setting2" = "Value2"

  }

  connection\_string {

    name  = "Database"

    type  = "SQLServer"

    value = "Server=some-server.mydomain.com;Integrated Security=SSPI"

  }

}

**~~Assignment:~~**

**~~Step1: Go to Azure Portal and Create Resource Group (Demo-rg) and VNet + Subnet~~**

**~~Step2: Update your configuration files to create VM in existing Resource Group and VNet~~**

**~~VNet.tf~~**

**~~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~~