Terraform: Variables in Terraform

Terraform : Deployment Automation

- Variables are necessary to parameterizing our deployments using Terraform.
- Input Variables enable user to pass in configuration values at the time of deployment.
- Using Terraform Input Variables is one way of deploying development, staging, or production environments using the same Terraform resource declarations, with slightly different configuration.

Terraform : Deployment Automation

- **Define Input Variables -**
- Terraform Input Variables are defined using a variable block that defines the variable name and other option parameters for the variable. These variable blocks can be place in any **.tf** file within your Terraform project, but a common location used is within a file named **variables.tf**.
- **Input Variable Definition**
- The variable block is defined with the variable keyword followed by the name of the Input Variable. Also, the required type property must be added to define the data type of the variable; such as string, number, object, or other supported data types.

Terraform : Deployment Automation

```
variable "environment" {
  type = string
}

variable "location" {
  type        = string
  default     = "eastus"
  description = "The Azure Region to deploy resources."
}
```

Terraform : Deployment Automation

- **Input Variable Conditions -**
- When defining Input Variables for a Terraform project, the Input Variable can have a custom validation rules defined. The custom validation rules for an Input Variable are defined by adding a validation block within the variable block for the Input Variable.

```
variable "location" {
  type        = string
  description = "The Azure Region to deploy resources."

  validation {
    condition     = contains(["eastus", "westus"], lower(var.location))
    error_message = "Unsupported Azure Region specified. Supported regions
include: eastus, westus"
  }
}
```

Terraform : Deployment Automation

- **Input Variable Types -**
- Primitive Types - Terraform supports three primitive types for Input Variables. **string, number, bool**
- Complex Types - Complex types allow you to group multiple values together into a single variable. Collection types, Structural types.
- Collection Types - A collection of multiple values grouped together as a single value.
  - list(...) – A sequence of values identified by an index starting with zero.
  - map(...) – A collection of values; each with a string label identifier.
  - set(...) – A collection of unique values without any secondary identifiers or ordering.

- **Input Variable Types -**
- Structural Types - A collection of multiple values of several distinct types grouped together as a single value.
  - object(…) – A collection of values each with their own type.
  - tuple(…) – A sequence of values each with their own type.

Will see you in Next Lecture…

See you in next lecture …