



*Terraform: Read, Generate, and Modify
Configuration*

Terraform : Deployment Automation

- **Section 8 : Read, Generate, and Modify Configuration**
 - Demonstrate use of Variables and Outputs
 - Describe Secure Secret Injection best practice
 - Understand the use of Collection and Structural types
 - Create and differentiate resource and data configuration

Terraform : Deployment Automation

- **Demonstrate use of Variables and Outputs :**
- User can create the variable Specific terraform Configuration File.

```
variable "region" {  
  default = "us-west-2"  
}
```

- User can define Default value in variable or can pass explicit value as well.
- Assign Value from command line.

```
terraform apply -var 'region=us-east-2'
```

Terraform : Deployment Automation

- Demonstrate use of Variables and Outputs :
- Mention Secrets in variable files are not recommend. As it will commit the secrets in VCS.
- User can create a local file with a name like **secret.tfvars** and use **-var-file flag** to load it.
- User can use multiple -var-file arguments in a single command.

```
terraform apply \  
-var-file="secret.tfvars" \  
-var-file="production.tfvars"
```

Terraform : Deployment Automation

- **Describe Secure Secret Injection Best Practice :**
- Multiple way to manage the secrets in Terraform. Few possible ways are -
 - AWS Secret Manager
 - GCP Secret Manager
 - Azure Creds
 - HashiCorp Vault
- Mention Secrets are in Terraform Config, Statefiles & Env Variables are not recommended.

Terraform : Deployment Automation

- Understand the use of Collection and Structural types :
- Collection type allows multiple values of one other type to be grouped together as a single value.
- Terraform have 3 Collection Types-
 - list() - A sequence of values.
 - map() - A collection of key-value pair.
 - set() - A collection of unique values

Terraform : Deployment Automation

- Understand the use of Collection and Structural types :
- Structural Type allows multiple values of several distinct types to be grouped together as a single value.
- Terraform have 2 Structural Types-
- object() - A pair of curly braces containing a comma-separated series of <KEY> = <TYPE> pairs.

```
{  
  name = "John"  
  age  = 52  
}
```

- tuple() - A pair of square brackets containing a comma-separated series of types.

```
["a", 15, true]
```

Will see you in Next Lecture...

Thank you!

A close-up photograph of a hand holding a black marker, completing the cursive word 'Thank you!' on a white surface. The hand is positioned on the right side of the frame, with the index and thumb fingers visible, holding the marker. The marker's tip is just finishing the exclamation mark. The background is a plain, light-colored surface.

See you in next lecture ...