

Terraform Certified Associate - IBM Cloud

Duration: 4 days (8hrs/day)

Prerequisites: Basic knowledge of IBM Cloud.

Course Objective: Equip participants with the essential skills to proficiently leverage Terraform for Infrastructure as Code (IaC) on IBM Cloud, covering setup, configuration management, modularization, remote state handling, and utilization of Terraform Cloud for scalable and secure infrastructure deployment, ultimately preparing them for success in obtaining the Terraform Certified Associate (003) certification.

Cloud Platform: IBM Cloud, Terraform Version: Latest

Module 1 - Getting Started & Setting Up Labs

Introduction to Infrastructure as Code and Terraform
Lab: Installation of Terraform on Windows
Comparison between Terraform and Ansible
Introduction to IBM Cloud CLI
Understanding Terraform Providers
Authenticate IBM Cloud with Terraform
Lab: Setting Up Terraform on Windows and IBM Cloud Authentication
Basic Terraform commands: init, plan, apply
Lab: Defining Provider & Using Basic Terraform commands

Module 2 – Building Cloud Infrastructure with Terraform

Lab: Creating Resource Groups in IBM Cloud
Lab: Provisioning Virtual Networks, Subnets, Floating IPs, and Network Interfaces of VPC Infra
Lab: Deploying Windows and Linux Virtual Servers in VPC infra
Lab: Configuring Cloud Storage, Security Groups, and Load Balancers in VPC infra
Understanding Terraform State file
Understanding Working of State file – Desired State & Current State
Terraform Provider Versioning
Lab: Methods to define Terraform Provider Versions



Module 3 - Read, Generate, Modify Configurations

Understanding Attributes and Output Values in Terraform Lab: Handling Terraform attributes and output values Lab: Referencing attributes across resources Understanding Terraform Variables and Data Types – (String, Number, Boolean, List, Map) Lab: Methods to Define Variables & Variable Arguments Lab: Fetching Data from List & Map in Variables Understanding Meta-Arguments – (for each, count, depends on) Lab: Using Meta-Arguments **Understanding Terraform Functions** Lab: Using element, zipmap, lookup function in terraform Understanding conditional expression and locals Lab: Using Conditional expression and Locals Understanding Expressions – for & Splat expression Lab: Using for and Splat expression **Understanding Data Sources & Dynamic Blocks** Lab: Using Data Sources Lab: Using Dynamic Blocks Lab: Exploring debugging techniques in Terraform Terraform Commands – validate, fmt Lab: Using terraform validate and terraform fmt Lab: Replacing Resource in terraform manually – taint and replace Lab: Using Terraform Graph utility Lab: Saving Terraform Plan to a file and apply from plan file

Module 4 - Terraform Provisioners

Understanding provisioners in Terraform

Understanding Connection Block

Types of provisioners

Lab: Implementing file, remote-exec and local-exec provisioners



Module 5 - Terraform Modules & Workspaces

Applying the DRY (Don't Repeat Yourself) principle
Understanding Usage of Terraform Modules
Standard Structure of Terraform Modules
Lab: Creating and Using local Modules
Lab: Utilizing Modules from Terraform Registry
Understanding and implementing Terraform workspaces
Lab: Working with Terraform Workspaces

Module 6 - Remote State Management

Integrating Terraform with Git for team collaboration
Understanding Basic Working of Git
Lab: Handling Git commands (initialize, commit, push, tagging, branching)
Challenges and security considerations in Terraform state
Lab: Remote state management with Terraform, including importing existing resources
Understand Terraform State Backend Configuration

Module 7 – Terraform Cloud and Enterprise Overview

Introduction to Terraform Cloud Creating infrastructure with Terraform Cloud Overview of Sentinel Security in Terraform Lab: Deploying Infrastructure with Terraform Cloud and Sentinel Security