



## Terraform - Infrastructure as Code

- 3 Days
- Lecture and Hands-on Labs
- Includes all objectives found on HashiCorp's Terraform Associate Certification

### Course Overview

As enterprises seek to deploy and maintain increasingly complex cloud infrastructure, there is a necessity to use “Infrastructure as Code” (IaC) tools, like Terraform. An open-source, state management tool developed by HashiCorp, Terraform allows developers to use a common coding interface to work through their various clouds safely and efficiently. Attendees will leave being able to write and understand Terraform code (HCL), have a clear understanding of Terraform's various components and supporting tools, as well as when to reach for Terraform over another IaC tool, such as Ansible.

### Who Should Attend

- DevOps Engineers
- Software Developers
- Technical Managers and Leads
- System and Cloud Administrators
- Network Engineers and Developers

### What You'll Learn



- Writing Terraform HCL code
- Deploying into common clouds such as AWS, Azure, Google Cloud, Docker, Oracle, Kubernetes, and VMWare
- Where Terraform fits in the Enterprise CI/CD model
- Differences between Terraform and Ansible
- Best practices
- Prepare for HashiCorp's Terraform Associate Certification
- AI LLM prompt engineering for Terraform snippets and jumpstarting solutions

### Outline

#### AI LLM Toolkit

-  Lecture + Lab: Large Language Model toolkit for AI Solution Assistance

#### Introduction to Terraform

-  Lecture: Terraform Course Map
-  Lecture: Introduction to Terraform

#### Software Control Management

-  Lecture + Lab: SCM Option #1 - GitHub
-  Lecture + Lab: SCM Option #2 - GitLab

## Overview of Terraform

- 📺 Lecture + Lab: Terraform Install

## Terraform Modules

- 🗣️ Lecture: Terraform HCL Syntax
- 📺 Lecture + Lab: Up and Running with Terraform
- 📺 Lecture + Lab: Terraform Variables
- 📺 Lecture + Lab: Output Values
- 🗣️ Lecture: Avoid the :latest Tag
- 🏆 Challenge: Terraform and Docker

## Beyond Basics

- 📺 Lecture + Lab: Terraform CLI Workspaces
- 📺 Lecture + Lab: Terraform Expressions and Errors
- 📺 Lecture + Lab: Resources - replace vs taint
- 📺 Lecture + Lab: Dynamic Operations with Functions
- 📺 Lecture + Lab: Creating a Terraform Module
- 📺 Lecture + Lab: Moving State - terraform state mv
- 📺 Lecture + Lab: Dynamic Provisioning with tfvars Files
- 📺 Lecture + Lab: Data Sources and HTTP Provider
- 🗣️ Lecture: Import pre-existing infrastructure
- 📺 Lecture + Lab: CHALLENGE - AWS import

## Loops

- 🗣️ Lecture: for\_each
- 📺 Lecture + Lab: Looping Constructs - for\_each

## Provisioning

- 📺 Lecture + Lab: local-exec Provisioner
- 📺 Lecture + Lab: Creating Delays
- 📺 Lecture + Lab: Terraform - templatefile Function

## Terraform Cloud

- 📺 Lecture + Lab: Terraform Cloud and Terraform Enterprise
- 📺 Lecture + Lab: Triggering Cloud Builds via Git Commits

## Dynamic Blocks

- 📺 Lecture + Lab: Dynamic Blocks


## AWS

- 📺 Lecture + Lab: Terraform and AWS
- 📺 Lecture + Lab: Output Values and AWS
- 📺 Lecture + Lab: AWS and looping with count vs for\_each
- 📺 Lecture + Lab: Correcting Resource Drift and AWS
- 🏆 Challenge: Terraform and AWS


## Azure

- 📺 Lecture + Lab: Terraform and Azure

## Google Cloud Platform

-  Lecture + Lab: Terraform and Google Cloud Platform



## Oracle

-  Lecture + Lab: Terraform and Oracle Cloud Infrastructure





## Terraform and Enterprise

-  Lecture + Lab: Deploy a Go RESTful API microservice with Terraform
-  Lecture: Terraform vs. Ansible
-  Lecture + Lab: Terraform and Ansible


## VMWare

-  Lecture: Terraform and VMWare
-  Lecture + Lab: Terraform and VMWare

## Helpful DevOps Tools (OPTIONAL)

-  Lecture + Lab: Open Policy Agents and Terraform
-  Lecture + Lab: GitHub Actions - GitLeaks
-  Lecture + Lab: GitHub Actions - Terraform
-  Lecture: Terragrunt

## Terraform Review

-  Lecture: HashiCorp Terraform Study Guide

## Prerequisites

Although not required, students with some experience programming, or pre-existing knowledge of cloud architecture, will most appreciate the technical nature of this hands-on course.

## Next Courses

- Jenkins Automation Server Essentials (2 days)
- Ansible Essentials (5 days)
- Go Essentials (5 days)
- Git and GitHub (2 days)
- Git and GitLab (2 days)

## Certification

- Terraform Essentials - Certification Project