

Specialized - Terraform

| | |
|----------------|-----------------------------|
| Code: | INNO-TERRA |
| Length: | 2 days |
| URL: | View Online |

Terraform is the leading tool for automating deployment and management of your infrastructure resources. Using Terraform as part of your continuous deployment pipeline enables repeatable results and eliminates human error when creating and managing infrastructure. As customers require new features, faster automation is the only way to keep up with demand. This class starts at the beginning and explains key concepts of infrastructure as code, Terraform principles, and industry best practices.

Skills Gained

Attendees will leave with all the skills necessary to plan, test, create and manage infrastructure across multiple environments. They will understand how to create re-usable, version-controlled configuration files for repeatable results. Attendees will learn best practices for securely storing variables and secrets.

Who Can Benefit

The audience for this class is Developers, Team Leads, DevOps, Architects, and any other Engineering personnel interested in an in-depth introduction to infrastructure management using Terraform. This class teaches best practices for creating, testing, and managing infrastructure.

Prerequisites

Attendees should have a basic understanding of Linux and command-line experience.

Course Details

What you learn:

Introduction

- Overview
- Architecture
- Why Terraform?
- Comparison of tools
- Core components
- Fundamental concepts

Programming Structure

- Providers
- Resources

- Variables
- Data sources
- Outputs

Resources

- Types and Arguments
- Behavior
- Dependencies
- Local only resources

Variables

- Inputs
- References
- Outputs and Locals
- Overrides

Terraform CLI

- Features
- Commands
- Managing infrastructure

State management

- Local or Remote?
- Remote storage options: S3, AzureRM, Google File Storage
- Integrating with Git
- Challenges with State file locking
- Importing existing resources

Provisioners

- Types of Provisioners
- Remote execute provisioners
- Local execute provisioners
- Storing provisioners in repository

Modules and Workspaces

- Understanding DRY (Don't Repeat Yourself) principle
- Variables and Modules
- Terraform module registry
- Terraform CLI workspaces

Security

- The right way to handle Access & Secret keys
- Terraform and Identity Access Management (IAM)

Wrap-up

- Review
- Q&A
- Next steps

Schedule (as of 3)

| Date | Location |
|------|----------|
|------|----------|

Download Whitepaper: Accelerate Your Modernization Efforts with a Cloud-Native Strategy
Get Your Free Copy Now