

**Introduction to Terraform**

**Course Number:** TRFM-104
**Duration:** 2 days

**Overview**

Accelebrate's Introduction to Terraform training course introduces attendees to the advantages of developing “Infrastructure as Code” with Terraform. Students learn how to manage the deployment of different infrastructure types via the appropriate Provider plugin. Participants also discover the ease with which infrastructure resources are defined in a declarative manner, allowing resources to be created, updated, or destroyed in a quick and efficient manner.

**Note:** This 2-day course can be taught in 4 half-day sessions online if preferred. In addition, this class can be adapted to other AWS services or other cloud providers.

**Prerequisites**

Students must be at ease working at the command-line, understand Amazon Web Services (AWS) such as EC2 or S3, and have experience using an SSH Client, such as OpenSSH on Linux, macOS or WSL, or PuTTY on Windows.

**Materials**

All Terraform training students receive comprehensive courseware.

**Software Needed on Each Student PC**

* Visual Studio Code (optional)
* Terraform Plugin for VS Code (optional)
* AWS CLI for your platform (Windows, Mac, or Linux) (optional)
* AWS access and secret key can be provided by you or by Accelebrate

**Objectives**

All students will learn how to:

* Use Terraform for standing up various AWS resources
* Apply plans to a Terraform Workflow
* Create basic and complex variables
* Write your own modules
* Find information about other AWS modules and data sources

**Outline**

* Introduction to Infrastructure as Code
	+ Infrastructure as Code, Config Management, Idempotence
	+ Terraform
	+ Installation
* Terraform Workflow
	+ The plan
	+ Applying and re-applying plans
	+ Destroying resources
	+ Various sub-commands
* HCL Configurations
	+ Providers
	+ Variables
	+ Resources
* Variable Types
	+ Variables, passing values to the configuration, Locals
	+ Basic and complex types
* Templates
	+ Control Structures
	+ Templates
* Data Sources
	+ Provider Data Sources
* State
	+ Local State
	+ Using “remote state” for working in teams
* Modules
	+ Modules structure
	+ Using modules
	+ Terraform Registry
	+ Writing your own modules
* Provisioners
	+ Local-exec
	+ File
	+ Remote-exec
* Importation of Foreign Resources
	+ Importation of resources created outside Terraform
		- terraform import
		- terraforming
* Auto-Scaling & Load-Balancing
	+ AWS EC2 ASG - Autoscaling Groups
	+ AWS EC2 ALB - Application Load Balancer
* Other AWS Resources
	+ Lambda
	+ VPC, EIP
	+ S3, EBS
	+ IAM, RDS
* Best Practices
	+ Terraform Best practices
	+ Tooling: Linters, scanners, testers
* HashiCorp Terraform Eco-System
	+ Terraform Cloud, Terraform Enterprise
	+ CDK
	+ Waypoint, Boundary
	+ Certification
* Conclusion