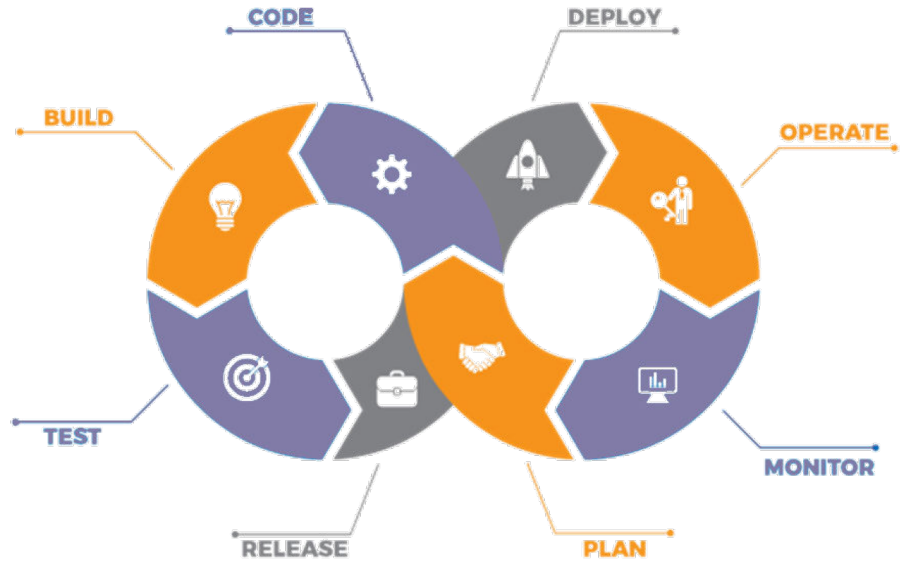# Introduction to Terraform

Agenda

01   What is Infrastructure as a Code?

02   Infrastructure as a Code vs Configuration Management

03   Introduction to Terraform

04   Installing Terraform on AWS EC2

05   Basic Terraform Operations

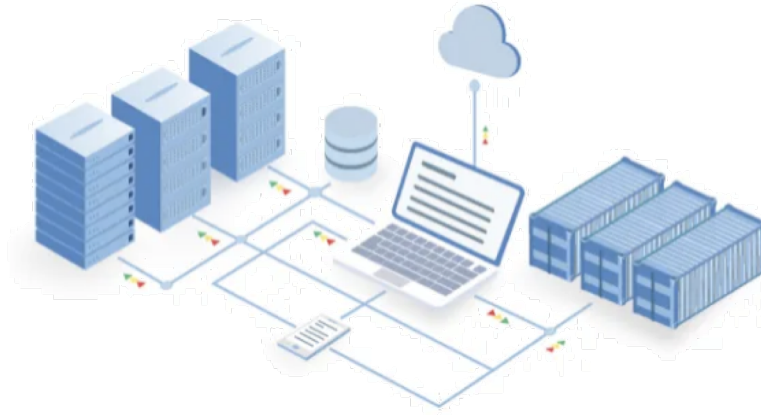**06**   Terraform Code Basics

07   Deploying an end to end Architecture using Terraform

What is Infrastructure as a Code?

What is Infrastructure as Code (IaC)?

Infrastructure as Code (IaC) is the management of **infrastructure** (networks, virtual machines, load balancers, and connection topology) in a descriptive model, using the same versioning as **DevOps** team uses for source **code**. **Infrastructure as Code** evolved to solve the problem of environment drift in the release pipeline.

Infrastructure as Code vs Configuration Management

IaC vs Configuration Management

Infrastructure as a Code

1. Can create / destroy hardware architectures
2. Can install software while bootstrapping servers
3. Should not be used as a replacement to CM tools

Configuration Management

1. Works only with softwares
2. Cannot work on hardware level, but can install any software
3. Cannot be used as a replacement to IaC tools

Introduction to Terraform

What is Infrastructure as Code (IaC)?

Terraform is an open-source infrastructure as code software tool created by HashiCorp. It enables users to define and provision a datacenter infrastructure using a high-level configuration language known as Hashicorp Configuration Language, or optionally JSON.

HashiCorp
Terraform

Installing Terraform for AWS

Installing Terraform for AWS

1. Install Terraform on local/EC2 Machine

2. Create a user, and use this user to authenticate terraform to AWS

3. Initialize Terraform Directory

Basic Terraform Operations

Basic Terraform Operations

There are four kinds of operations in Terraform:

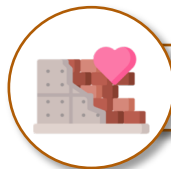**Init**

Plan

Apply

Destroy

HashiCorp Terraform

Terraform Code Basics

Terraform Code Basics

Following is a snipped of terraform file. Terraform files are saved with the extension of tf (*.tf). It has two main components, **block type** and **Key Value Pairs**

Block Type

```
1  provider "aws" {
2
3    region = "ap-south-1"
4    access_key = "AKIA2ESXROHTZZNFXDKI"
5    secret_key = "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"
6
7  }
8
```

Key Value Pairs

HashiCorp
Terraform

Deploying end to end Infrastructure using Terraform