



## Terraform Fundamentals

Devops

- **Nível:** Intermédio
  - **Duração:** 14h
- 

### Sobre o curso

**Aprenda a utilizar o Terraform para automação de infraestruturas multi-cloud, dominando a infraestrutura como código (IaC).**

O Terraform é uma das ferramentas mais populares para o provisionamento de infraestruturas no mercado, suportando múltiplos fornecedores de cloud (AWS, Azure, GCP, Oracle, etc.). Este curso permite aos formandos utilizarem o Terraform para definir infraestrutura como código (IaC) e automatizar o deployment de infraestruturas.

---

### Destinatários

- Administradores de Sistemas
  - Engenheiros de Sistemas
  - Engenheiros de DevOps
- 

### Objetivos

- Compreender os conceitos e componentes fundamentais do Terraform.
  - Desenvolver competências para criar, alterar e destruir recursos utilizando o Terraform.
  - Utilizar e gerir o Terraform Remote State e configurar armazenamento com a AWS.
  - Implementar boas práticas de gestão e configuração utilizando Git com Terraform.
- 

### Pré-requisitos

- Inglês técnico.
  - Conhecimento básico de cloud computing.
  - Conhecimentos fundamentais de redes TCP/IP.
  - Familiaridade com a consola Linux.
-

# Programa

- Introdução ao Terraform
- Conceitos Básicos do Terraform
- Configuração Avançada do Terraform
- Gestão de Configuração com Controlo de Versões
- Integrações de Provisionamento

## Introdução ao Terraform

- O que é Infraestrutura como Código (IaC)
- Conceitos chave: Recursos, Fornecedores, Estado, Linguagem de Configuração
- Criar um Laboratório de configuração e instalar o Terraform
- Iniciar um projeto Terraform

## Conceitos Básicos do Terraform

- Ficheiros de configuração do Terraform: main.tf, variables.tf, outputs.tf
- Declarar recursos e gerir o estado
- Ciclo de vida do Terraform: Planear, Aplicar, Destruir
- Utilização de variáveis e fontes de dados
- Gestão de Estado e Colaboração
- Fornecedores mais utilizados: AWS, Azure, GCP, Docker, etc.

## Configuração Avançada do Terraform

- Criação condicional de recursos com count e for\_each
- Blocos dinâmicos e expressões
- Gestão de dados sensíveis com variáveis de entrada sensíveis

## Gestão de Configuração com Controlo de Versões

- Utilização de controlo de versões (Git) com o Terraform
- Fluxos de trabalho do Git para projetos Terraform
- Melhores práticas do Terraform e revisão de código

## Integrações de Provisionamento

- Deploy de scripts de provisionamento
- Integração com CloudInit
- Integração com Ansible