

CI CD with Ansible and Terraform

Duration: 8 Days (8 hours/day)

Prerequisites:

- 1) Basic knowledge of Linux OS
- 2) Basic knowledge of Cloud Azure.

Course Objective: In this course participants will learn how to automate Linux Server with Ansible, Deploy Infrastructure on Azure, Versioning of Files using Git and Jenkins to automate Infrastructure Lifecycle.

Cloud Platform: Microsoft Azure, **Terraform Version:** Latest, **Ansible Version:** Latest Open Source

Lab Requirement: Koenig DC (CentOS 9) & Participant Azure Account

Git and GitHub

Module 1 – Git

Introduction to Version Control System History of GIT

Git Basics

States in Git Installing Git

Lab: Installation of GIT on system

Configuration of Git

Working with Repositories

Lab: Create GIT repository

Basic Git Commands

Lab: Create commits and switch in between commits

Working with Remotes

Lab: Create repository on GITHUB and then push local repository on GITHUB repository

Tagging

Lab: Give tag to commits

Git Branching

Lab: Create branch and then switch and merge branches

Automation with Ansible

Module 1 – Introduction to Ansible

Evolution of Infrastructure

Overview of Infrastructure as a Code What is Configuration Management Ansible Overview

Case Study

Module 2 – Ansible Architecture and Installation

Ansible Architecture and It's Working Ansible in DevOps
Installation and Configuration Working with Command Line Tools

Lab: Installation and configuration of Ansible on machine

Module 3 – Ansible Modules

Overview of Modules Types of Modules Core Modules

Extras Modules Return Values

Ad-Hoc Commands Case Study

Lab: Run Ansible modules with Ansible Ad-Hoc Commands

Module 4 – The Playbook Grammar

Introduction to YAML Playbook YAML Definition Playbook Terms

Playbook Tasks

Writing Ansible Playbooks

Lab: Create playbook for running multiple task on managed nodes

Module 5 – Variables, Conditions, Loops, Handlers and Jinja2 Templates

Variables

Lab: Run playbook by using variables

Loops

Lab: Run playbook by using loop

Notify and Handlers

Lab: Run playbook by using notify and handlers

Jinja2 Templates

Lab: Create jinja2 templates file and use in playbook

Module 6 – AWX Tower

Installing AWX Tower Features of Ansible Tower Managing Jobs

Lab: Installation and configuration of AWX Tower on system

Manage and Track Inventory Remote Command Execution

Lab: Run playbooks on AWX Tower

Case Study

Terraform with Azure

Module 1 - Getting Started & Setting Up Labs

Introduction to Infrastructure as Code and Terraform

Lab: Installation of Terraform on Windows

Comparison between Terraform and Ansible

Introduction to Azure CLI

Understanding Terraform Providers

Authenticate Azure with Terraform

Lab: Setting Up Terraform on Windows and Azure Authentication

Basic Terraform commands: init, plan, apply

Lab: Defining Provider & Using Basic Terraform commands

Module 2 – Building Cloud Infrastructure with Terraform

Lab: Creating Resource Groups in Azure

Lab: Provisioning Virtual Networks, Subnets, Public Ips, and Network Interfaces

Lab: Deploying Windows and Linux VMs

Lab: Configuring Azure Storage, Security Groups, and Load Balancers

Understanding Terraform State file

Understanding Working of State file – Desired State & Current State

Terraform Provider Versioning

Lab: Methods to define Terraform Provider Versions

Module 3 – Read, Generate, Modify Configurations

Understanding Attributes and Output Values in Terraform

Lab: Handling Terraform attributes and output values

Lab: Referencing attributes across resources

Understanding Terraform Variables and Data Types – (String, Number, Boolean, List, Map)

Lab: Methods to Define Variables & Variable Arguments

Lab: Fetching Data from List & Map in Variables

Understanding Meta-Arguments – (for_each, count, depends_on)

Lab: Using Meta-Arguments

Understanding conditional expression and locals

Lab: Using Conditional expression and Locals

Understanding Expressions – for & Splat expression

Lab: Using for and Splat expression

Understanding Data Sources & Dynamic Blocks

Lab: Using Data Sources

Lab: Using Dynamic Blocks

Lab: Exploring debugging techniques in Terraform

Terraform Commands – validate, fmt

Lab: Using terraform validate and terraform fmt

Lab: Replacing Resource in terraform manually – taint and replace

Lab: Using Terraform Graph utility

Lab: Saving Terraform Plan to a file and apply from plan file

Module 4 - Terraform Provisioners

Understanding provisioners in Terraform

Understanding Connection Block

Types of provisioners

Lab: Implementing remote-exec and local-exec provisioners

Module 5 - Terraform Modules & Workspaces

Applying the DRY (Don't Repeat Yourself) principle

Understanding Usage of Terraform Modules

Standard Structure of Terraform Modules

Lab: Creating and Using local Modules

Lab: Utilizing Modules from Terraform Registry

Understanding and implementing Terraform workspaces

Lab: Working with Terraform Workspaces

CI/CD Tool - Jenkins

Module 1 – Introduction to Jenkins

Introduction to Jenkins

Jenkins Installation

Introduction to Jenkins UI

Create First Job

Lab: Installation of Jenkins

Module 2 – Add SSH Node Credentials in Jenkins

Install SSH Plugin in Jenkins Add Credentials of Node

Lab: Install SSH Plugin in Jenkins and add SSH server credentials for authentication

Integrate Node SSH Server with Jenkins

Lab: Integration of SSH server with Jenkins to run projects

Module 3 – Build Job on Remote Machine

Deploy Web Server Automatically through Jenkins

Lab: Deploy Web Server by using Jenkins and access Web Server from browser

Module 4 – Jenkins Security

Enable/Disable Login in Jenkins Allow Users to Sign up

Lab: Enable/Disable Login for users

Create Users Manually in the Jenkins DB Create and Manage Roles for Jenkins Users

Lab: Create users and roles and give privileges to users as per roles

Module 5 – Jenkins Email Integration

Install a Mail Plugin Integration Jenkins and G-mail

Add notifications to your jobs

Lab: Configure and enable Gmail notification for Jenkins Projects

Module 6 – Jenkins Ansible Integration

Store Playbooks, Inventory and Configuration Files on GitHub Automatically Pull Code from GitHub then Run on Ansible Server

Lab: Integration of Jenkins with Ansible and pull playbooks from GITHUB and run on Ansible server

Module 7 – CICD with Jenkins and Terraform

Store Terraform Files, Playbooks, Inventory and Configuration Files on GitHub

Automatically Deploy VMs With the Help of Terraform and Once Machine is Deployed then Run Ansible to Configure Those VMs

Lab: Intergrating Jenkins with Ansible and Terraform