

**DevOps with Jenkins, Terraform, and HashiCorp Stack**

**Course Number:** JEN-110WA
**Duration:** 3 days

**Overview**

This DevOps with Jenkins, Terraform, and HashiCorp Stack training class teaches attendees how to build DevOps CI/CD workflows for Cloud and Virtual environments. This class dives deeply into DevOps with Jenkins, HashiCorp Stack, and Terraform for AWS resource deployment.  Participants learn how to use Amazon Web Services’ features, including Virtual Private Cloud, Elastic Load Balancing, and Application Virtualization.

**Prerequisites**

No prior experience is presumed.

**Materials**

All students receive comprehensive courseware.

**Software Needed on Each Student PC**

Attendees will not need to install any software on their computer for this class. The class will be conducted in a remote environment that Accelebrate will provide; students will only need a local computer with a web browser and a stable Internet connection. Any recent version of Microsoft Edge, Mozilla Firefox, or Google Chrome will be fine.

**Objectives**

* Confidently use the Jenkins, Terraform, and HashiCorp stack
* Understand the various key components of the stack
* Understand Git and the concepts and basic operations
* Branch, merge, and perform operations on Git repositories
* Improve the reliability of the build and release processes
* Understand the various pieces of Jenkins for CI/CD
* Configure Maven and Jenkins to deploy the generated artifacts to Artifactory
* Use Terraform with AWS
* Use Atlas with Terraform
* Implement secret security with Vault
* Configure Packer for Virtualization
* Implement Vagrant with Terraform and AWS

**Outline**

* Introduction to Continuous Integration, Continuous Delivery, and Jenkins-CI
	+ Foundation of Agile AppDev
	+ XP Flow
	+ Extreme Programming
	+ Agile Development
	+ What is Continuous Integration
	+ What is Continuous Integration (cont'd)
	+ Typical Setup for Continuous Integration
	+ Setup Notes for Continuous Integration
	+ CI with Artifact Management
	+ What is Continuous Delivery?
	+ Why Continuous Delivery?
	+ DevOps and Continuous Delivery
	+ Continuous Delivery Challenges
	+ Continuous Delivery vs Continuous Deployment
	+ Jenkins Continuous Integration
	+ Jenkins Features
	+ Running Jenkins
* Getting Started With Maven
	+ Terminology and Basic Concepts
	+ Artifacts
	+ Lifecycle
	+ Default Lifecycle
	+ Plugins
	+ Running Maven - the Story So Far
	+ Running Maven from an IDE
	+ Common Goals
	+ pom.xml
	+ Artifact Coordinates
	+ Standard Layout for Sources
* A Web Application in Maven
	+ A More Complex Project
	+ Putting it Together With Maven
	+ Packaging the Target Artifact
	+ The Source Tree
	+ Dependencies
	+ Transitive Dependencies
	+ Dependency Scope
	+ Working With Servers
	+ Declaring and Configuring Plugins
	+ Running the Plugin
	+ Binding a Plugin Goal to the Lifecycle
	+ Archetypes
* Job Types in Jenkins
	+ Different types of Jenkins Items
	+ Configuring Source Code Management(SCM)
	+ Working with Subversion
	+ Working with Git
	+ Storing Credentials
	+ Service Accounts
	+ Build Triggers
	+ Schedule Build Jobs
	+ Polling the SCM
	+ Polling vs. Triggers
	+ Maven Build Steps
* Continuous Delivery and the Jenkins Pipeline
	+ Continuous Delivery
	+ DevOps and Continuous Delivery
	+ Continuous Delivery Challenges
	+ Continuous Delivery with Jenkins
	+ The Pipeline Plugin
	+ Defining a Pipeline
	+ A Pipeline Example
	+ Parallel Execution
	+ Creating a Pipeline
	+ Invoking the Pipeline
	+ Interacting with the Pipeline
	+ Pipeline vs. Traditional Jobs
* Jenkins Plugins
	+ Jenkins Plugins
	+ Jenkins for Teams
	+ Installing Jenkins Plugins
* Best Practices for Jenkins
	+ Secure Jenkins
	+ Users
	+ Backups
	+ Reproducible Builds
	+ Testing and Reports
	+ Large Systems
	+ Distributed Jenkins
* Overview of the Amazon Web Services (AWS)
	+ Amazon Web Services
	+ The History of AWS
	+ The Initial Iteration of Moving amazon.com to AWS
	+ The AWS (Simplified) Service Stack
	+ Accessing AWS
	+ Direct Connect
	+ Shared Responsibility Model
	+ Trusted Advisor
	+ The AWS Distributed Architecture
	+ AWS Services
	+ Managed vs. Unmanaged Amazon Services
	+ Compute and Networking Services
	+ Elastic Compute Cloud (EC2)
	+ AWS Lambda
	+ Auto Scaling
	+ Elastic Load Balancing (ELB)
	+ Virtual Private Cloud (VPC)
	+ Route53 Domain Name System
	+ Elastic Beanstalk
	+ Security and Identity Services
	+ Identity and Access Management (IAM)
	+ AWS Directory Service
	+ AWS Certificate Manager
	+ AWS Key Management Service (KMS)
	+ Storage And Content Delivery
	+ Elastic Block Storage (EBS)
	+ Simple Storage Service (S3)
	+ Glacier
	+ CloudFront Content Delivery Service
	+ Database Services
	+ Relational Database Service (RDS)
	+ DynamoDB
	+ Amazon ElastiCache
	+ Redshift
	+ Messaging Services
	+ Simple Queue Service (SQS)
	+ Simple Notifications Service (SNS)
	+ Simple Email Service (SES)
	+ AWS Monitoring with CloudWatch
	+ Other Services Example
* Introduction to Git
	+ What is Git?
	+ Git's Design Goals
	+ Branching and Merging
	+ Centralized Version Control
	+ Distributed Version Control
	+ Git Basics
	+ Getting Git
	+ Git on the Server
	+ Git Repository Managers
	+ Git on Somebody Else's Server
* Basic Git Operations
	+ Using Git
	+ Definitions
	+ Commit
	+ How to Think About Commits
	+ Viewing History
	+ Configuring Git
	+ Configuration Scope
	+ User Identification
	+ GPG Signing
	+ Gnu Privacy Guard
	+ GPG Basics
	+ GPG and Git
	+ .gitignore
	+ Other Useful Configurations
* Branching, Merging, and Remotes
	+ Branching
	+ Branches in Git
	+ Merge
	+ Fast Forward Merge
	+ --no-ff
	+ More Than One Repository
	+ Working with Remotes
	+ Fetch and Pull
	+ Push
	+ Pull Requests
	+ Tagging a Commit
	+ Lightweight Tags
	+ Annotated Tags
	+ Sharing Tags
	+ Checking Out a Tag
* Git Work Flows
	+ Work Flows
	+ Local Work Flow
	+ Feature Branches
	+ Centralized Workflow
	+ Integration Manager Workflow
	+ Other Work Flows Are Possible
* Introduction to GitFlow
	+ What is GitFlow
	+ Benefits
	+ How GitFlow works?
	+ GitFlow Extension
	+ Initializing GitFlow
	+ Features
	+ Release
	+ Hotfixes
* Examining History
	+ Looking at History
	+ Log
	+ Blame
	+ Bisect
* Introduction to Terraform
	+ What is Terraform
	+ Why Infrastructure as Code
	+ Who is Hashicorp
	+ Other Infrastructure Automation Tools
	+ Terraform
	+ How Does Terraform Work?
	+ Sample Configuration File
	+ Terraform vs Configuration Management
	+ Drawbacks of Configuration Management
	+ Benefits of Declarative Infrastructure
	+ Drawbacks of Declarative Infrastructure
	+ Terraform processing
	+ Master mode
	+ Drawbacks of Master mode
	+ Masterless mode
	+ Agents
	+ Drawbacks of Agents
	+ Agentless
* Terraform Basics
	+ The Terraform Command
	+ Running Terraform
	+ The Current State
	+ Directory Structure
	+ Load Order
	+ Configuration Format
	+ HCL
	+ String Interpolation
	+ Configuration Files
	+ Variables
	+ Outputs
	+ Data Sources
	+ Resources
	+ Data Sources and Resources
	+ Providers
	+ Built-in Providers
	+ Configuring Providers
	+ Modules
	+ Defining a Module
	+ Configuring the AWS Provider
	+ Basic Usage - Define an EC2 Instance
	+ Manipulating DNS Entries
* VPC ELB and ALB with Terraform
	+ Amazon Virtual Private Cloud
	+ VPC Advantages
	+ Regions and Availability Zones
	+ VPC Concepts
	+ Security Groups
	+ Network Address Translation (NAT)
	+ Public IP Addresses
	+ Load Balancing
	+ VPC with Terraform
	+ Some Notes on Load Balancers
* Using ECS and ECR with Terraform
	+ What is Docker
	+ Where Can I Run Docker?
	+ Docker and Containerization on Linux
	+ Linux Kernel Features: cgroups and namespaces
	+ The Docker-Linux Kernel Interfaces
	+ Docker Containers vs. Traditional Virtualization
	+ Docker as Platform-as-a-Service
	+ Docker Integration
	+ Docker Services
	+ Docker Application Container Public Repository
	+ Docker Command-line
	+ Starting, Inspecting, and Stopping Docker Containers
	+ Docker Benefits
	+ Docker on AWS - ECR and ECS
	+ Using Amazon ECR
	+ Authenticating to ECR
	+ Building a Docker Image
	+ Push to ECR
	+ Elastic Container Service
	+ ECS Cluster
	+ ECS Container Instances
	+ Container Instance with Terraform
	+ Task Definition
	+ Task Definition with Terraform
	+ Service Definition
* Introduction to Vagrant
	+ What is Vagrant?
	+ Why Use Vagrant?
	+ The Vagrantfile
	+ Sample Vagrantfile
	+ Vagrant Box
	+ Running Vagrant
	+ Networking
	+ Multi-Machine Usage
* Introduction to Packer
	+ What is Packer?
	+ Why Use Packer?
	+ Use Cases
	+ Supported Platforms
* Utilizing Packer
	+ Installing Packer
	+ Building an Image
	+ The Build Process
	+ Templates
	+ Template Example
	+ Builders
	+ Amazon EC2 Builder
	+ VirtualBox Builder
	+ Provisioners
	+ Configuration Management Provisioners
	+ Post-Processors
	+ Parallel Builds
	+ Remote Builds
* Conclusion