

## Effective DevOps with Ansible, AWS and Docker

### Course Summary

#### Description

This hands-on course is packed with practical, real-world advice on building and evolving modern application infrastructures, while fully automating application deployment. With a focus on Ansible and Amazon Web Services, the key ideas translate to other tools and providers as well.

Starting with the core useful abstractions of Ansible - playbooks and plays, templating, and roles - before exploring higher-level devops patterns of node orchestration, configuration, deployment and control. The course concludes with containerization, using Docker.

Note this training is thoroughly hands-on; students are expected to fully participate, as they master the fundamentals of implementing devops with Ansible.

#### Objectives

Students will gain experience with the most useful features of Ansible; understand the key facets of configuration management; learn the principles of cloud server orchestration; and gain valuable, rarely-taught wisdom and insight into devops best practices.

#### Topics

- Introduction
- Core modules and how to use them
- Variables in Ansible
- More Advanced Playbooks
- Handlers
- Deploying applications
- Roles in Ansible
- Basic Database handling
- More Advanced Inventories
- Orchestration of multiple hosts
- Extending Ansible
- Managing credentials
- Cloud Orchestration with Ansible and AWS
- Tagging AWS resources
- Configuring AWS instances
- More advanced AWS orchestration
- The Ansible ecosystem: Ansible Galaxy
- Containers with Docker (Optional 4th Day)
- Container Orchestration (Optional 4th Day)
- Using Containers on AWS (Optional 4th Day)

#### Audience

This course is designed for developers, managers, and project leaders.

#### Prerequisites

There are no prerequisites for this course.

#### Duration

Four days

## Effective DevOps with Ansible, AWS and Docker

### Course Outline

- I. Introduction**
  - A. Overview of YAML syntax (the format of playbooks)
  - B. Understanding configuration management and server orchestration
  - C. Building the Ansible Inventory
  - D. Ansible commands
  - E. Advanced SSH security for Ansible
  - F. Playbooks for organization of tasks, policies, and designs
  - G. Overview of YAML syntax (the format of playbooks)
- II. Core modules and how to use them**
  - A. Installing system packages
  - B. Working with files
  - C. Starting and stopping services and daemons
- III. Variables in Ansible**
  - A. Templating
  - B. Leveraging the standard vars layout
  - C. Simple scalar values
  - D. Mapping types and list types
- IV. More Advanced Playbooks**
  - A. Looping semantics
  - B. Diving deeper into modules and tasks
  - C. Conditionals (if, when)
  - D. Inclusions
- V. Handlers**
  - A. Thinking in terms of idempotency
  - B. Triggering groups of tasks
  - C. "notify" actions and single triggers
  - D. Use cases for handlers
- VI. Deploying applications**
  - A. Design and architecture of a REST API server
  - B. Integrating Git, and the ansible git module
  - C. Deploying an API server from Github
- VII. Roles in Ansible**
  - A. Intro: roles as a powerful core abstraction
- B. Dividing responsibilities and functionality into roles**
- C. Implementing simple roles**
- VIII. Basic Database handling**
  - A. Installation and configuration of SQL databases
  - B. Role separation of DB and webserver concerns
- IX. More Advanced Inventories**
  - A. Groupings
  - B. Inventory-specific data
  - C. Dynamic inventories
- X. Orchestration of multiple hosts**
  - A. Inventory organization
  - B. Network configuration
- XI. Extending Ansible**
  - A. Creating custom modules
  - B. Basic structure of a module
  - C. Interfaces and output formats
- XII. Managing credentials**
  - A. Credential models for Ansible
  - B. Using ansible-vault
  - C. Best practices for securely storing credentials
- XIII. Cloud Orchestration with Ansible and AWS**
  - A. Working with AMIs
  - B. ec2 module: create/terminate/etc. instances
  - C. Working with security groups
- XIV. Tagging AWS resources**
  - A. Tag setting
  - B. Operations on tag sets
  - C. Tags and dynamic inventories
- XV. Configuring AWS instances**
  - A. Assigning elastic IPs
  - B. Instance bootstrap
  - C. Ansible-pull and pull-mode playbooks

## Effective DevOps with Ansible, AWS and Docker

### Course Outline (cont'd)

#### **XVI. More advanced AWS orchestration**

- A. Load balancing with Ansible
- B. Autoscaling groups
- C. Adjusting scaling policies
- D. Tagging AWS resources
- E. Attaching EBS volumes to instances

#### **XVII. The Ansible ecosystem: Ansible Galaxy**

#### **XVIII. Containers with Docker (Optional 4th Day)**

- A. Understanding containers - benefits and challenges
- B. Creating and running simple docker images
- C. Provisioning images with Dockerfiles
- D. Ansible and Docker

#### **XIX. Container Orchestration (Optional 4th Day)**

- A. Using Docker registries
- B. Image management
- C. Clustering with Kubernetes

#### **XX. Using Containers on AWS (Optional 4th Day)**

- A. ECS - Amazon's EC2 Container Service
- B. ECR - AWS container registry