Skillsoft
global knowledge™

IMPLEMENTING A CI/CD PIPELINE

Course Code: 5890

This class is a hands-on experience in how to create your CI/CD pipeline from start to finish.

Building, testing, and deploying code can be complex and time-consuming. This three-day, instructor-led, hands-on, technical class provides the solution of how to create an automated CI/CD pipeline that builds, tests, and deploys code every time a developer pushes a code change, ensuring only high-quality and approved code gets deployed.

Developers don't use isolated software tools, so why learn tools in isolation? This course combines the most widely used enterprise technologies into single, integrated, end-to-end pipelines. We'll teach you git from the ground up so you can securely push code to a remote repository such as Github or Bitbucket. Then we'll teach you how to configure servers using Ansible from code stored in git. Next, we'll integrate git and Ansible with Jenkins CI/CD to build, test, and deploy your code into Staging and Production environments, creating an automated end-to-end DevOps pipeline.

If desired, private clients can substitute Bitbucket for Github and/or Chef for Ansible. It is also possible to expand the class duration to include Terraform and/or Docker.

What You'll Learn

In this Implementing a CI/CD Pipeline Course, you will:

- Maintain code using Git and GitHub
- Create, test, and deploy Ansible playbooks through a Jenkins CI/CD pipeline
- Complete integration tests
- Write CI/CD as code using Jenkins
- Compile, test, package, and permanently store Java applications
- Store artifacts

Who Needs to Attend

- DevOps Engineers
- System Operations
- Code Developers
- Software Developers
- Quality Assurance Professionals
- Software Testers
- Product Owners

- Infrastructure Engineers
- Development Team Members
- Operations Team Members
- Project Managers
- Technical Product Managers

Prerequisites

- Understand the purpose of infrastructure servers
- Familiar with scripting of any kind
- Experience working at the command line of Unix, Linux, or PowerShell
- Experience with a text editor of any kind

IMPLEMENTING A CI/CD PIPELINE

Course Code: 5890

CLASSROOM LIVE	$3,295 CAD	3 Day

Classroom Live Outline

Technology Overview

- Terraform – Infrastructure control (optional with 5-day delivery)
- Git - Source Control Management
- Chef - Configuration Management
- Inspec - Cookbook Testing
- Jenkins - Continuous Integration
- Docker – Containerization (optional for 4-day delivery)

Terraform – Infrastructure Automation (optional with 5-day delivery)

- Terraform use and purpose
- Getting started with Terraform
- General concepts
- Installation
- Configuration Files
- Provisioning and Building Infrastructure
- Changing Infrastructure
- Destroying Infrastructure
- Resource Dependencies
- Input Variables
- Output Variables
- Modules – Pre-defined Configurations
- Remote state management
- Terraform Enterprise
- Triggering a Plan from Version Control
- Workspaces
- Terraform Enterprise Runs
- Migration from Open Source to Enterprise
- AWS
    - Managing infrastructure in AWS
    - Creating IAM Policies
    - Authenticating to AWS

    - Launching a serverless application - Lambda
- Azure
    - Managing infrastructure in Azure
    - Authentication Methods for Azure
- Git – Source Control Management
    - Purpose and overview of Git
    - Use cases for Git
    - Git workflow
    - Github, Bitbucket and other Git providers
    - Installation and configuration
    - Finding help on Git
    - Creating Local Git Repositories
    - Basic Git Commands
    - Comparing commits
    - Using a Remote Repository
    - Branching and Merging
    - Using SSH keys with Git private repositories
    - Chef – Configuration Management
    - Chef Architecture and call flow
    - Chef Use cases
    - Idempotence
    - Resources
    - Recipes
    - Cookbooks
    - Integration Testing and Test Kitchen
    - Chef Server – a central management repository
    - Ohai, Attributes and the Node Object
    - Chef Roles
- Jenkins – Continuous Integration / Continuous Deployment
    - Jenkins Overview, Use Cases and History
    - Initializing a Jenkins server
    - Projects & Jobs
    - Freestyle versus Pipeline
    - Writing a Declarative Pipeline Project - CI/CD as Code
    - Distributing builds using Master and Agent Nodes
    - Views and Folders
    - Managing Credentials
    - Integrating with Git - Source Control Management
    - Triggers: Webhooks and Polling
    - Notifications: Slack and SMTP Email
    - Testing Chef Cookbooks with Test Kitchen in Jenkins
    - Multibranch Pipelines - Reading Entire Repositories
    - Using Maven to Compile, Test & Package Java Applications
    - Storing artifacts in JFrog Artifactory
    - Deploying Chef cookbooks with Jenkins

- Docker – Containerization And Micro-Service (optional with 4-day delivery)
  - Container Use cases: what problems Docker solves
  - Docker Concepts and Components
  - Most-used Docker Commands
  - Storing Docker images
  - Docker Compose
  - Launching microservices using Docker
  - Building a Docker Image using Jenkins
  - Deploying a Docker container using Chef and Jenkins

Finally, we'll create the end-to-end Pipeline using Git, Chef & Jenkins:

Three-day option:

- Write Chef cookbooks with Inspec tests
- Push the cookbooks to a branch within a Git repo
- This push will automatically trigger Jenkins to download the cookbooks
- Jenkins will then initiate testing the Chef cookbooks using Test Kitchen
- If any tests fail, Jenkins notifies the developer to fix and repeat the build
- Once all tests pass, Jenkins can require human approval. Once approved, Jenkins will upload the cookbooks to the Chef Server
- Jenkins will then trigger Chef-managed webservers (in AWS) to download the cookbooks from the Chef Server and deploy the cookbook code
- Jenkins will then send notifications over both Slack and email regarding any failures or the successful deployment of the cookbooks

Four-day option adds:

- Upon successful testing in Test Kitchen, create and save a Docker image
- Notify teams of successful deployment of Docker containers

Five-day option adds:

- Write Inspec tests for Terraform configurations
- Test our Terraform Configuration in Test Kitchen
- Jenkins will use Terraform to launch Docker Hosts running Docker containers, as well as use Chef to configure those containers to run micro-services

Skillsoft
global knowledge™

IMPLEMENTING A CI/CD PIPELINE

Course Code: 5890

VIRTUAL CLASSROOM LIVE	$3,295 CAD	3 Day

Virtual Classroom Live Outline

Part 1: Technology Overview

1. Git - Source Control Management
2. Chef - Configuration Management
3. Jenkins - Continuous Integration/Continuous Deployment

Part 2: Git - Source Control Management

1. Purpose overview and use cases
2. Git workflow
3. Configuring git on your local machine
4. Getting help with Git
5. Local vs. Global vs. System configurations
6. Basic Git Commands
7. Creating local git repositories
8. Branching and merging
9. Using remote repositories
10. Pushing code to Github using public and private SSH keys

Part 3: Ansible - Configuration Management

1. Ansible purpose and use cases
2. Architecture and call flow
3. Ansible installation, configuration, and validation
4. Control nodes and managed nodes
5. Ansible managed hosts
6. Host inventory; hosts and groups
7. Repeatable code: Playbooks
8. Introduction to YAML
9. Modularizing code: Roles
10. Ansible variables
11. Dynamic configuration with facts
12. Finding errors: Ansible unit testing
13. Ensuring code quality: Ansible integration testing

Part 4: Jenkins – Continuous Integration / Continuous Deployment

1. CI/CD overview, use cases and history
2. Plugin architecture
3. Initializing a Jenkins server
4. Projects and jobs
5. Freestyle jobs
6. CI/CD as Code: Pipeline projects
7. Declarative vs. scripted pipelines
8. Jenkins Environment variables and parameters
9. Distributed architecture: Master and agent nodes
10. Views and Folders
11. Managing credentials and secrets
12. Integrating with git Source Control Management
13. Triggers: Webhooks and Polling
14. Notifications: Instant messaging and SMTP Email
15. Approval inputs
16. Testing Ansible playbooks in Jenkins
17. Multibranch Pipelines: Reading entire repositories
18. Conditional Logic
19. Deploying Chef cookbooks with Jenkins: An automated end-to-end deployment pipeline

Mar 18 - 20, 2026 | 10:00 AM - 6:00 PM EDT

Jun 10 - 12, 2026 | 10:00 AM - 6:00 PM EDT

Sep 15 - 17, 2026 | 10:00 AM - 6:00 PM EDT

IMPLEMENTING A CI/CD PIPELINE

Course Code: 5890

PRIVATE GROUP TRAINING	3 Day

Visit us at www.globalknowledge.com or call us at 1-866-716-6688.

Date created: 2/9/2026 6:33:14 AM