Jenkins Training Course Curriculum

Prerequisites

- Basic understanding of CI/CD concepts
- Familiarity with version control systems (Git)
- Basic knowledge of Linux/Unix commands
- Understanding of software development lifecycle

Lab Setup Requirements

- Jenkins LTS version
- Java Development Kit (JDK)
- Git
- Docker (optional)
- Maven/Gradle (for build examples)
- Text editor (VS Code, IntelliJ, or any preferred editor)

Course Duration: 3 Days

Day 1: Jenkins Fundamentals and Architecture

- Introduction to CI/CD
  - Continuous Integration concepts
  - Continuous Delivery/Deployment concepts
  - Benefits and best practices
- Jenkins Architecture
  - Controller/Agent architecture
  - Master-slave configuration
  - Distributed builds
  - Security considerations
- Jenkins Installation and Setup
  - Installation methods
  - Initial configuration
  - Plugin management
  - User management and security
- Basic Jenkins Jobs
  - Freestyle projects
  - Pipeline projects
  - Job configuration
  - Build triggers
  - Build parameters
- Lab Exercise: Setting up Jenkins and creating basic jobs

Day 2: Jenkins Pipelines and Groovy

- Introduction to Jenkins Pipelines
  - Pipeline concepts
  - Pipeline syntax
  - Pipeline structure
- Groovy Fundamentals for Jenkins
  - Basic Groovy syntax
  - Variables and data types
  - Control structures
  - Functions and closures
- Declarative Pipeline
  - Pipeline structure
  - Stages and steps
  - Environment variables
  - Post-build actions
- Scripted Pipeline
  - Advanced Groovy scripting
  - Custom functions
  - Error handling
  - Shared libraries
- Lab Exercise: Creating and managing pipelines

Day 3: Advanced Pipeline Features and Deployment

- Pipeline Best Practices
  - Code organization
  - Reusable components
  - Error handling
  - Performance optimization
- Advanced Pipeline Features
  - Parallel execution
  - Matrix builds
  - Pipeline templates
  - Pipeline visualization
- Deployment Automation
  - Deployment strategies
  - Environment management
  - Deployment pipelines
  - Rollback procedures
- Integration with Tools
  - Version control integration
  - Build tools integration
  - Container orchestration
  - Cloud platforms
- Final Project
  - End-to-end pipeline implementation
  - Deployment automation
  - Best practices implementation

- Q&A session