



---

## DevOps Online Training Course Curriculum

### About Course:

DevOps engineers are all-rounders; DevOps bridge the gap between the development, test and production teams. This course will teach you how to become a full-fledged DevOps engineer. In this course we will start with building with 3-tier infrastructure and deploy an application and scale it horizontally and vertically, provision using configuration management tools, Monitor the deployed application for zero-downtime. This certification course curriculum covers DevOps tools like Docker, Jenkins, Kubernetes, agile methodologies, scrum, Git, Maven, Jenkins, Continuous Integration and Testing.

### Who should Learn DevOps for Developers?

- Software Professionals

**Course duration:** 30 hrs

---

## DevOps 360-degree view

### Why DevOps?

- Business Perspective
- IT Perspective
- Developer Perspective
- Tester Perspective
- Operations Perspective

### What is DevOps?

- Definition
- Stakeholders of DevOps

### What is Agile and Scrum?

- Agile Development Process
- Role of Dev in Agile

### Problem That DevOps Solves

### Making a DevOps Transition

## How to do DevOps Automation?

- Introduction
- Tools
- DevOps Technology Categories
- Collaboration
- Planning
- Issue Tracking
- Monitoring
- Configuration Management
- Source Control
- Dev Environments
- Continuous Integration
- Continuous Testing
- Continuous Deployment

## DevOps Tools

### Source Code Management

- What is Version Control System?

### Git

- Installation
- Configuration
- Basic Commands
- Branches
- Repositories
- Tracking

### Build

- Automated Build Process

### Maven

- Introduction
- Maven Structure
- Maven Dependencies
- Maven Repositories
- Maven Plugins
- Integrated Maven Build

## Continuous integration

### Jenkins

- What is Jenkins?

- Best Practices

### **Installation and configuration**

- Prerequisites
- Download & installation
- Configuration tour

### **Managing Jenkins**

- Securing Jenkins
- Manage Credentials
- Manage Plugins

### **Creating Application Builds**

- Anatomy of the build
- Cloning sample project
- Manual compilation with Maven
- Manually Testing, Packaging and Running the Appn
- Creating a Jenkins Job and configuring a Git Repon
- Compiling in Jenkins
- Browsing the workspace in Jenkins
- App Packaging in Jenkins
- Importing Job config.xml

### **Plugins**

- Introduction
- Plugin Architecture
- Extension Pointsn
- Getting Plugins
- Plugin Wiki
- Useful Plugins Overview
- Source Code Plugins
- Trigger Plugins
- Build Tool Plugins

### **Continuous Testing and Continuous Integration and Testing**

- Adding steps to Freestyle Project
- Creating a Pipeline job to execute Maven
- Archiving in a Pipeline
- Checking out git repository in pipeline
- The Master Agent Model
- Allocating a node and workspace in Pipeline
- High level progress with Pipeline stages
- Triggering Automated Builds
- Notifications when a build fails
- Duplicating a job
- Executing unit tests

## **Finding and Managing Plugins**

- The need for plugins
- Integrated Code Coverage
- Assessing a plugin
- Installing the HTML Publisher plugin
- Publishing HTML Reports
- Testing Plugins and Plugin Types

## **Building Continuous Delivery Pipeline**

- Continuous Delivery
- Backup and Restore
- Starting point and Pipeline stashing
- A Second Node Allocation
- Executing a Deployment pipeline
- Check in pipeline script to Git

## **Containerization**

### **Docker**

- Docker Overview
- Docker Overview
- Understanding Docker
- Difference between Physical Servers, Virtual Machines and Docker
- Docker Installation
- Docker CLI Overview
- Docker and container

### **Building Container Images**

- Docker file
- Docker file instructions
- Multi stage Docker build

### **Storing and Distributing Images**

- Docker Hub
- Docker Store
- Docker Registry

### **Managing Containers**

- Docker container Commands
- Docker Network and Volumes

### **Docker Networking**

### **Docker Volumes (Storage)**

## **Docker Compose**

- Installation
- Docker Compose Yaml file
- Docker Compose Commands
- Docker App

## **Docker Swarm and Services**

- Introduction
- Roles within a Docker Swarm
- Creating and managing a Swarm
- Managing a cluster
- Docker Swarm services & stacks

## **Kubernetes**

- Overview
- Introduction to Microservices
- Clustering and Orchestration
- Kubernetes Architecture
- Kubernetes Core Concepts
- Pods
- Namespaces
- API primitives
- Kubernetes runtime
- Health checks
- Application Scheduling
- Kubernetes Networking
- Service Discovery
- DNS
- Multitenancy
- Kubernetes Namespaces
- Kubernetes Storage Overview
- Persistent Storage & Stateful sets
- Monitoring, Logging & Troubleshooting
- Creating Kubernetes Clusters
- Cluster Authentication, Authorization & Container Security
- Running Stateful Applications with Kubernetes

## **OpenShift**

- Overview
- Understanding OpenShift's business value
- OpenShift flavors
- Installing OpenShift
- OpenShift Concepts
- OpenShift Networking

- CI/CD Pipelines in OpenShift

BISP