OpenShift Container Deployment

Audience OpenShift Container Deployment Course

The OpenShift Container Deployment course is intended for developers, DevOps engineers, and system administrators who want to deploy, manage, and scale applications using OpenShift.

Prerequisites OpenShift Container Deployment Course

Experience with Linux, basic knowledge of containers and Kubernetes is recommended. Familiarity with DevOps concepts and CI/CD workflows is helpful for optimal understanding.

Training Execution OpenShift Container Deployment

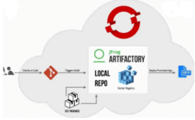
The training consists of interactive presentations combined with live demos and practical hands-on labs under the guidance of an experienced trainer.

Certificate OpenShift Container Deployment

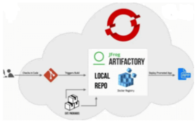
Participants receive a certificate of completion for the OpenShift Container Deployment course after successfully finishing the training.

Duration: 2 days **Price: € 1499**

Open Schedule



Openshift Container Deployment



Content Course OpenShift Container Deployment

In the course OpenShift Container Deployment, participants learn how to deploy, manage, and scale applications in an OpenShift environment. The training covers container technology, OpenShift architecture, CI/CD pipelines, and autoscaling techniques.

OpenShift Fundamentals

The course starts with an overview of OpenShift, differences with Kubernetes, and how to use the web console and oc CLI tool. Topics include projects, RBAC, templates, operators, and services.

Containers and Images

This module covers container technology, building images with Docker and Podman, using image registries, Source-to-Image (S2I) workflows, and security best practices for containers.

OpenShift Architecture

Participants learn about OpenShift's internal architecture, including master and worker nodes, control plane components, networking, storage solutions, multi-tenancy, monitoring, and logging.

Deploying Applications

This module teaches deploying applications with DeploymentConfigs, implementing rolling updates, blue-green deployments, resource management, health checks, ConfigMaps, and Secrets.

Configuring CI/CD Pipelines

Participants learn how to set up CI/CD pipelines using OpenShift Pipelines and Tekton, integrate with Git repositories, manage triggers, implement pipeline security, and follow CI/CD best practices.

Managing and Scaling Applications

The focus is on scaling applications manually and automatically, using autoscalers, ensuring application resilience, setting resource limits, monitoring via Prometheus, and logging with Elasticsearch and Kibana.

Modules Course OpenShift Container Deployment

Module 1: OpenShift Fundamentals	Module 2: Containers and Images	Module 3: OpenShift Architecture
What is OpenShift? Kubernetes vs OpenShift OpenShift Web Console oc CLI Tool Projects and Namespaces Role-Based Access Control OpenShift Templates Operators Overview Developer vs Admin Perspective Intro to OpenShift Services	Containers vs Virtual Machines Container Lifecycle Building Container Images Docker vs Podman Image Registries Pushing and Pulling Images Image Streams BuildConfigs in OpenShift Source-to-Image (S2I) Security Best Practices	Master and Worker Nodes Control Plane Components etcd Data Store Networking in OpenShift SDN vs OVN Ingress and Routes Storage in OpenShift Persistent Volumes Monitoring and Logging Multi-Tenancy and Isolation
Module 4: Deploying Applications	**Module 5: Configuring CI/CD Pipelines**	**Module 6: Scaling Applications**
Deployment Strategies Creating DeploymentConfigs Rolling Updates Blue-Green Deployment Canary Deployment Managing Resources Environment Variables Secrets and ConfigMaps Health Checks Automatic Rollbacks	OpenShift Pipelines Overview Tekton Pipelines Basics Pipeline Resources and Tasks Building CI/CD Pipelines Integrating with Git Repositories Triggers and Webhooks Managing Pipeline Runs Pipeline as Code (YAML) Security and Permissions Best Practices for CI/CD	Scaling Applications Manually Horizontal Pod Autoscaler Cluster Autoscaling Self-healing Applications Managing Application Updates Resource Quotas and Limits Observability with Prometheus Application Metrics and Alerts Using OpenShift Console Logging with ELK Stack