

DevOps Course Curriculum

Module 1: Introduction to DevOps

- . DevOps culture and principles
- . The DevOps lifecycle
- . Benefits of DevOps in software development
- . Tools overview: Jenkins, Docker, Kubernetes, etc.

Module 2: Git Basics

- . Git fundamentals
- . Initializing a repository (`git init`)
- . Staging changes (`git add`)
- . Committing changes (`git commit`)
- . Basic workflow: modify-stage-commit
- . Tools: Git, GitHub/GitLab

Module 3: Advanced Git and Collaboration

- . Branching and merging
- . Creating and switching branches (`git branch`, `git checkout`)
- . Merging branches (`git merge`)
- . Collaborative development with GitHub/GitLab
- . Pushing and pulling changes (`git push`, `git pull`)
- . Pull requests and code reviews

Module 4: Continuous Integration (CI) Concepts

- . CI principles and benefits
- . Setting up a basic CI pipeline
- . Defining pipeline stages (build, test, deploy)
- . Configuring build triggers
- . Tools: Jenkins (introduction), GitLab CI, CircleCI

Module 5: Jenkins for CI

- . Jenkins architecture
- . Creating and running basic jobs
- . Configuring build steps
- . CI best practices
- . Tools: Jenkins, Blue Ocean plugin

Module 6: Infrastructure as Code (IaC) Introduction

- . Concepts and benefits of IaC
- . Overview of IaC tools
- . Declarative vs. imperative approaches
- . Tools: Overview of Ansible, Terraform, CloudFormation

Module 7: Ansible Basics

- . Ansible architecture and concepts
- . Writing simple playbooks
- . Running ad-hoc commands
- . Tools: Ansible, Ansible Tower

Module 8: Introduction to Containers

- . Container concepts and benefits
- . Comparison with virtual machines
- . Overview of container runtimes
- . Tools: Docker, containerd

Module 9: Docker Fundamentals

- . Docker architecture
- . Basic Docker commands
- . Container lifecycle (create, start, stop, remove)
- . Tools: Docker, Docker Hub

Module 10: Dockerfiles and Docker Compose

- Writing a Dockerfile
- Building and running custom images
- Introduction to Docker Compose
- Managing multi-container applications

Module 11: Kubernetes Architecture

- Kubernetes components
- Master and worker nodes
- Control plane components
- Tools: Kubernetes, Minikube, kubectl

Module 12: Kubernetes Resources

- Pods and deployments
- Services and ingress
- ConfigMaps and Secrets

Module 13: Continuous Delivery (CD) Concepts

- CD principles and benefits
- Deployment pipelines
- Environment promotion strategies
- Tools: Jenkins, GitLab CI, Spinnaker

Module 14: Deployment Strategies

- Blue-Green deployments
- Canary releases
- Rolling updates
- Tools: Kubernetes, Istio

Module 15: Automated Testing in CD

- Types of automated tests (unit, integration, acceptance)
- Test execution in CI/CD pipelines
- Test reporting and quality gates
- Tools: JUnit, Selenium, Cucumber, Jest

Module 16: Cloud Computing Fundamentals

- Cloud service models (IaaS, PaaS, SaaS)
- Public, private, and hybrid clouds
- Benefits of cloud in DevOps
- Tools: AWS, Azure, Google Cloud Platform

Module 17: Cloud Services for DevOps

- Compute services (EC2, Azure VM, Google Compute Engine)
- Storage services (S3, Azure Blob Storage, Google Cloud Storage)
- Database services (RDS, Azure SQL Database, Google Cloud SQL)
- Tools: AWS CLI, Azure CLI, Google Cloud SDK

Module 18: Monitoring in DevOps

- Importance of monitoring
- Key metrics and KPIs
- Overview of monitoring tools
- Tools: Prometheus, Grafana, Nagios

Module 19: Prometheus and Grafana

- Prometheus architecture and data model
- Setting up basic Prometheus monitoring
- Creating dashboards with Grafana

Module 20: Log Management

- Centralized logging concepts

- . Introduction to ELK Stack
- . Elasticsearch for log storage and search
- . Logstash for log processing
- . Kibana for log visualization

Module 21: DevSecOps

- . Security in CI/CD pipelines
- . Automated security scanning
- . Basic infrastructure and application security concepts
- . Tools: SonarQube, OWASP ZAP

Module 22: Site Reliability Engineering (SRE)

- . SRE principles and practices
- . Service Level Objectives (SLOs) and Error Budgets
- . Reliability and scalability focus
- . Tools: Prometheus (for SLIs/SLOs), Pingdom

Module 23: Incident Management and Chaos Engineering

- . Incident detection and classification
- . Escalation procedures
- . Conducting effective postmortems
- . Introduction to Chaos Engineering
- . Tools: PagerDuty, Jira, Chaos Monkey

Module 24: GitOps and Advanced Kubernetes

- . GitOps principles and workflows
- . Git as the single source of truth
- . Introduction to Helm for package management
- . Brief overview of Kubernetes Operators and CRDs
- . Tools: ArgoCD, Helm

Module 25: Performance Optimization and Serverless Computing

- Application and infrastructure performance optimization techniques
- Introduction to serverless computing
- Cloud cost optimization strategies
- Tools: New Relic, AWS Lambda, Azure Functions, Google Cloud Functions