

OPSCRAFT ACADEMY

PROFESSIONAL DEVOPS TRAINING PROGRAM

Linux · AWS · Git · CI/CD · Docker · Kubernetes · Terraform · Ansible · Monitoring

BUILD · SHIP · REPEAT

Practical training • Hands-on labs • Real-world projects • Interview preparation

Duration: 3–4 Months

Program Overview

A structured, hands-on DevOps program designed to build practical skills across cloud infrastructure, automation, CI/CD, containers, Kubernetes, Infrastructure as Code and monitoring.

FOUNDATION + CLOUD

Linux · Shell Scripting · Git & GitHub · AWS

Strong fundamentals and cloud infrastructure

CI/CD

Maven · Tomcat · SonarQube · Nexus · Jenkins

Automated build, test and deployment workflows

CONTAINERS + ORCHESTRATION

Docker · ECR/ECS · Kubernetes · Helm · EKS

Containerized applications and production-style orchestration

AUTOMATION + INFRASTRUCTURE

Terraform · Ansible

Infrastructure provisioning and configuration management

OBSERVABILITY + PROJECTS

Prometheus · Grafana · Capstone Projects

Monitoring, alerting and end-to-end implementation

01 • PHASE 1 • FOUNDATION

Linux & System Administration

Build a strong Linux foundation for DevOps operations.

Tools: Linux · Ubuntu

• Linux fundamentals and architecture • Users, groups and permissions • Package management and networking commands • Practical Linux administration labs	• File system, navigation and file management • Processes, services and system administration • Disk usage, compression and system troubleshooting
--	--

Recommended duration	5–6 Days
----------------------	----------

02 • PHASE 1 • FOUNDATION

Shell Scripting

Automate routine administration and operational tasks with shell scripts.

Tools: Bash · Shell

• Shell fundamentals and script structure • Conditions and loops • File, user and service automation • Hands-on automation exercises	• Variables, input/output and operators • Functions and reusable scripts • Backup and monitoring scripts
Recommended duration	4–5 Days

03 • PHASE 1 • FOUNDATION

Git & GitHub

Manage source code effectively and collaborate using modern Git workflows.

Tools: Git · GitHub

• Version control fundamentals • Repositories, commits and remote operations • Pull requests and collaboration • GitHub repositories and access management	• Git workflow and core commands • Branches, merging and conflict resolution • Stash, tags, rebase and squash
--	---

Recommended duration	4–5 Days
----------------------	----------

04 • PHASE 1 • CLOUD

AWS & Nginx

Deploy and operate applications on core AWS infrastructure.

Tools: AWS · EC2 · S3 · VPC · IAM · CloudFront · Nginx

• Cloud and AWS fundamentals • Load Balancing and high-availability concepts • IAM, VPC and network security • RDS, DynamoDB and AWS CLI • AWS billing and cost awareness	• EC2, AMI, EBS and Auto Scaling • S3 and CloudFront • Lambda, SNS and CloudWatch • Nginx reverse proxy and web hosting
Recommended duration	25–30 Days

05 • PHASE 2 • CI/CD

Build & Code Quality

Understand application build, dependency and code-quality workflows.

Tools: Maven · Nexus · SonarQube · JaCoCo

• Build lifecycle and build tools • Dependencies, plugins and repositories • Nexus repository concepts • JUnit / TestNG and JaCoCo coverage	• Maven project structure and POM • Maven lifecycle and common commands • SonarQube code-quality analysis
---	---

Recommended duration	4–5 Days
----------------------	----------

06 • PHASE 2 • CI/CD

Jenkins & CI/CD

Build automated CI/CD pipelines from source control to deployment.

Tools: Jenkins · GitHub · Maven

• CI/CD fundamentals • Jobs, pipelines and Jenkinsfile • Credentials and role-based access • Deployment to application servers	• Jenkins setup and administration • Build triggers and webhooks • Pipeline integration with Git, Maven, Nexus and SonarQube • Agents and distributed build architecture
Recommended duration	12–15 Days

07 • PHASE 3 • CONTAINERS

Docker & Containerization

Containerize applications and manage images, networks and persistent data.

Tools: Docker · Docker Hub · Docker Compose

• Container fundamentals and architecture • Docker commands and image lifecycle • Volumes and networking • Docker Compose and YAML	• Images, containers and Dockerfiles • Docker Hub and image publishing • Multi-stage builds • Container deployment patterns
--	---

Recommended duration	12–15 Days
----------------------	------------

08 • PHASE 3 • ORCHESTRATION

Kubernetes, Helm & EKS

Deploy, scale and manage containerized applications on Kubernetes.

Tools: Kubernetes · Helm · EKS · kubectl

• Kubernetes architecture and core objects • Services and networking • Namespaces and resource management • Ingress and application exposure • Helm charts and releases • Jenkins and Kubernetes integration	• Pods, Deployments and ReplicaSets • ConfigMaps and Secrets • Persistent volumes and claims • kubectl and troubleshooting • AWS EKS fundamentals
Recommended duration	12–15 Days

09 • PHASE 4 • AUTOMATION

Terraform & Ansible

Provision infrastructure and automate server configuration.

Tools: Terraform · Ansible

• Infrastructure as Code concepts • Providers, variables and outputs • AWS infrastructure provisioning • Ansible inventory and ad-hoc commands • Automation and server configuration	• Terraform workflow and state • Modules and reusable infrastructure • Configuration management concepts • Playbooks, variables and modules
Recommended duration	8–10 Days

10 • PHASE 4 • OBSERVABILITY

Prometheus & Grafana

Implement monitoring, dashboards and alerting for infrastructure and Kubernetes.

Tools: Prometheus · Grafana · PromQL · Alertmanager

• Monitoring and observability fundamentals • Metrics, exporters and PromQL • Alerting with Alertmanager • Alerts and notifications	• Prometheus architecture and configuration • Service discovery • Grafana data sources and dashboards • Kubernetes monitoring
Recommended duration	5–6 Days

Modern DevOps Toolset

Additional tools introduced as a dedicated module so the core syllabus remains clean and easy to follow.

GitHub Actions

Cloud-native CI/CD

Build and automate CI/CD workflows directly from GitHub repositories.

Argo CD

GitOps

Deploy and continuously reconcile Kubernetes applications using Git as the source of truth.

Trivy

Security Scanning

Scan container images and infrastructure configurations for vulnerabilities and misconfigurations.

OpenTofu

Infrastructure as Code

Explore an open-source IaC workflow compatible with Terraform-style configurations.

AWS CloudFormation

AWS Infrastructure

Understand AWS-native infrastructure provisioning and stack management.

Note: The modern toolset is an additional exposure module. Primary hands-on projects remain centered on the core DevOps stack.

Hands-On Projects

The program concludes with practical projects that combine multiple tools and reflect common DevOps workflows.

AWS Web Application Deployment

Deploy and expose a web application using AWS infrastructure, Nginx, load balancing and scaling.

AWS · Nginx · ELB · Auto Scaling

End-to-End CI/CD Pipeline

Create a CI/CD pipeline integrating source control, build, code quality, artifact management and application deployment.

Git · Maven · Jenkins · SonarQube · Nexus · Tomcat

Kubernetes Application Deployment

Deploy an application to Kubernetes, package it with Helm and expose it through cloud infrastructure.

Kubernetes · EKS · Helm · DNS

Infrastructure Automation

Provision cloud infrastructure with IaC and automate server configuration.

Terraform · Ansible · AWS

Monitoring & Observability

Build dashboards, collect metrics and configure alerting for infrastructure and Kubernetes workloads.

Prometheus · Grafana · PromQL · Alerting

Interview Preparation

Regular mock interviews, practical troubleshooting discussions, project walkthroughs and DevOps-focused interview preparation are integrated throughout the program.

Ready to start?

OpsCraft Academy

Phone / WhatsApp: +91 86184 72802

Email: opscraftacademy@gmail.com

Instagram: [@ops.craft.academy](https://www.instagram.com/ops.craft.academy)