DevOpsSchool

Course curriculum · November 2026

LEVEL 03 OF 04 · INTERMEDIATE

# Observability Intermediate Training

Define SLOs with burn-rate alerts, tune OpenTelemetry pipelines and sampling, control cardinality and manage dashboards and alerts as code.

| | | |
|---|---|---|
| LIVE ONLINE<br>**Max 8-10 hours** | CORPORATE (DAY-WISE)<br>**3 days · Days 1-3** | SELF-PACED VIDEO<br>**1-year LMS access** |
| BATCH SIZE<br>**Max 10 learners** | NEXT BATCHES<br>**1st of every month · weekends: first Saturday** | CERTIFICATE<br>**Verifiable certificate of completion** |

| Foundation | Essential | Intermediate | Advanced |
|---|---|---|---|
| 1 day · Included | 2 days · Included | 3 days · This curriculum | 5 days · Next step |

## INSIDE THIS CURRICULUM


| | |
|---|---|
| What you will achieve | Outcomes and level focus |
| Who should attend | Audience and prerequisites |
| Agenda | 34 topics · Foundation to Intermediate |
| Hands-on labs | 3 labs |
| Formats and fees | Live, 1:1, self-paced, corporate |
| Trainer and reviews | Rajesh Kumar |
| Enrolment | Batches, payment, contacts |


## WHY DEVOPSSCHOOL

- Rated 4.6 on Google
- 1,500+ verifiable certificates issued in the public registry
- Trainer Rajesh Kumar: 20 years in industry
- Small batches: max 10 learners, every session recorded

**Course page:** www.devopsschool.com/courses/observability/observability-intermediate-training.html
**Talk to us:** WhatsApp +91 99057 40781 · +1 (469) 756-6329 · contact@DevOpsSchool.com · reply within 1 hour

OUTCOMES

# What you will achieve

By the end of the Observability course path, you will be able to:

1

### Understand observability signals

Explain logs, metrics, traces and alerts and when to use each to answer questions.

2

### Build observability pipelines

Instrument services with OpenTelemetry and run Prometheus, Grafana, Loki and Tempo stacks.

3

### Run observability at scale

Use SLOs, cost controls and standards to operate observability across an organisation.

WHAT THE INTERMEDIATE LEVEL ADDS

### SLOs, OpenTelemetry pipelines, scale and automation

**Hands-on:** SLO-driven alerting as code. Define SLOs for a service, generate burn-rate alerts, add tail sampling in the Collector and manage the dashboards and alerts through a Git pull request.

WHO SHOULD ATTEND

- Regular Observability users who want to automate, integrate and manage it at a deeper level.
- Specialists who support Observability for colleagues or customers.
- Leads preparing Observability for wider use across projects.

BEFORE YOU START

- The Essential level, or equivalent hands-on experience.
- You can already: Collect, store and query logs, metrics and traces, build useful dashboards and route alerts with sensible thresholds and escalation.
- Basic computer skills and a laptop with a stable internet connection for the live sessions and labs.
- This curriculum includes every earlier level, so it is complete on its own.

AGENDA · FOUNDATION TO INTERMEDIATE

# Observability Intermediate Training agenda

34 topics. Each level builds on the one before it, so this curriculum lists every earlier level in full. Days are for corporate (day-wise) delivery; live online batches cover the same agenda in max 8-10 hours.

## DAY 1 — Foundation: Observability concepts, signals and tool landscape

Included from Foundation

1. Introduction to observability: what it is, why it is important and how it differs from traditional monitoring
2. Key concepts and terminology: telemetry, signals, instrumentation, cardinality, golden signals, RED and USE methods
3. Logging overview and types of logs: application, system, access and audit logs, structured versus unstructured
4. Metrics overview and types of metrics: counters, gauges, histograms and summaries
5. Tracing overview and types of traces: spans, trace context propagation and distributed traces
6. Alerting overview and types of alerts: symptom-based versus cause-based, threshold and anomaly alerts
7. Telemetry pipeline architecture: agents, collectors, storage backends and visualisation layers
8. OpenTelemetry overview: API, SDKs, Collector, OTLP protocol and semantic conventions
9. Popular observability tools and frameworks: Prometheus, Grafana, Loki, Tempo, Jaeger, Elastic, OpenSearch and SaaS platforms
10. First stack: Prometheus, node_exporter and Grafana with Docker Compose and a first dashboard

## DAY 2 — Essential: Collection, storage, querying, dashboards and alert routing

Included from Essential

11. Log collection and aggregation: Fluent Bit, Vector or Grafana Alloy shipping logs to Loki or Elasticsearch
12. Log analysis and visualization: LogQL queries in Grafana and Kibana searches for troubleshooting
13. Metric collection and storage: Prometheus scraping, exporters, service discovery, retention and remote write
14. Metric analysis and visualization: PromQL rate, sum by and histogram_quantile in Grafana panels
15. Trace collection and storage: OpenTelemetry auto-instrumentation, Collector pipelines, Jaeger or Grafana Tempo
16. Trace analysis and visualization: waterfall views, critical path, service maps and trace-to-log links
17. Alerting rules and thresholds: Prometheus alerting rules, for durations and Grafana alerting
18. Alert notification and escalation: Alertmanager routes, grouping, inhibition, silences and on-call escalation
19. Instrumenting an application: custom metrics with client libraries and structured logs carrying trace IDs
20. Kubernetes observability basics: kube-prometheus-stack, kube-state-metrics and cAdvisor
21. Dashboard design: golden signals layouts, variables and avoiding dashboard sprawl
22. Signal correlation: exemplars linking metrics to traces and logs to traces

## DAY 3 — Intermediate: SLOs, OpenTelemetry pipelines, scale and automation

**New at this level**

23. Service level objectives: SLIs, SLO targets, error budgets and multi-window burn-rate alerts with Sloth or Pyrra
24. OpenTelemetry Collector in depth: receivers, processors such as batch, memory_limiter and attributes, exporters and routing
25. Sampling strategies: head-based versus tail-based sampling and the cost versus fidelity trade-off
26. Cardinality management: label hygiene, relabelling, drop rules and finding expensive series
27. Long-term metrics storage: Thanos, Grafana Mimir or VictoriaMetrics for retention and global queries
28. Continuous profiling: Grafana Pyroscope and eBPF profilers as a fourth signal

29. eBPF-based observability: zero-code instrumentation with Grafana Beyla and network visibility with Cilium Hubble
30. Synthetic and real user monitoring: blackbox_exporter probes, synthetic checks and browser RUM
31. Observability as code: dashboards and alert rules in Git with the Grafana Terraform provider and CI validation
32. Telemetry security: PII redaction, TLS between agents and backends and multi-tenant access
33. Incident investigation workflow: alert to dashboard to trace to log, runbooks and postmortems
34. Troubleshooting the pipeline: dropped spans, scrape failures, backpressure and collector memory issues

PRACTICE

## Hands-on labs

You build these yourself during the course, guided by the trainer.

FOUNDATION

**Stand up a first observability stack**

Run Prometheus, node_exporter and Grafana with Docker Compose, build a host dashboard and identify which questions logs and traces would answer next.

ESSENTIAL

**Full-signal view of a microservice**

Instrument a small microservice with OpenTelemetry, send metrics, logs and traces to Prometheus, Loki and Tempo, and route a latency alert to a chat channel.

INTERMEDIATE

**SLO-driven alerting as code**

Define SLOs for a service, generate burn-rate alerts, add tail sampling in the Collector and manage the dashboards and alerts through a Git pull request.

HOW YOU CAN LEARN

## Formats and fees

The same agenda in four formats. Fees are in Indian rupees as of November 2026; the course page shows them in your currency.

| FORMAT | DURATION | WHAT YOU GET | FEE |
|---|---|---|---|
| **Live & interactive (online)** | Max 8-10 hours | Instructor-led public batch of max 10 learners with demos, labs, Q&A and assignment review. Weekday batches start on the 1st of every month, weekend batches on the first Saturday of every month. | **INR 34,999** |
| **1:1 live training (online)** | Max 8-10 hours, one-to-one | The same agenda, taught one-to-one by the trainer. Starts immediately, without waiting for the next batch; sessions planned around your working hours, at your pace. | **INR 99,999** |
| **Self-paced video** | 1-year LMS access | All four Observability levels, plus 20+ more courses and 50+ tools, at your own pace. | **INR 833/month** billed yearly at INR 9,996 |
| **Corporate (day-wise)** | 3 days for this level | A private batch for your team, on demand. Foundation 1 day, Essential 2, Intermediate 3, Advanced 5 (cumulative). | **Quote on request** |

ASSESSMENT AND CERTIFICATE

Your progress is checked through hands-on labs and assignments at every level. On completion you receive a DevOpsSchool certificate of completion with a unique credential ID that anyone can verify at devopsschool.com/certificates.

INCLUDED WITH EVERY LIVE BATCH

- Every live session is recorded and stays in your LMS for a year.
- Can't attend your batch? Move to next month's batch.
- Free demo class before you pay.
- Free 10-15 minute call with an advisor or the trainer to check the course fits you.

YOUR TRAINER

## Learn from Rajesh Kumar

Principal Architect, trainer and mentor in DevOps, SRE, DevSecOps and platform engineering, with 20 years across 8+ software MNCs, including Staff DevOps/SRE Engineer for a leading Japanese mobile-payment platform and Principal DevOps Architect and Manager in Europe.

Profile: www.rajeshkumar.xyz · Videos: youtube.com/@TheDevOpsSchool

| | |
|---|---|
| **20** years in industry | **200+** corporate trainings |
| **10,000+** engineers mentored | **70+** organisations coached |

REVIEWS

## What learners say

Rated 4.6 on Google · more reviews at devopsschool.com/testimonials and in our video testimonials.

> “Rajesh is very good trainer. Rajesh was able to resolve our queries and question effectively. We really liked the hands-on examples covered during this training program.”
>
> Indrayani, India

> “The training was very useful and interactive. Rajesh helped develop the confidence of all.”
>
> Abhinav Gupta, Pune

START YOUR OBSERVABILITY TRAINING

## Join the next batch

Weekday batches start on the 1st of every month and weekend batches on the first Saturday of every month, with max 10 learners each. 1:1 training can start immediately; corporate batches run on demand. We reply within 1 hour.

Pay by UPI, credit and debit cards, net banking or PayPal. Before you pay, read our refund policy. All levels of this course: www.devopsschool.com/courses/observability/

**ENROL OR RESERVE A SEAT**
**Observability Intermediate Training**

**WHATSAPP · INDIA**
**+91 99057 40781**

**WHATSAPP · USA**
**+1 (469) 756-6329**

**EMAIL**
**contact@DevOpsSchool.com**

**Next step after this level: Observability Advanced Training** - Design observability for large, multi-cluster estates with controlled cost, AI-assisted operations and an organisation-wide practice and roadmap.