DevOpsSchool

Course curriculum · November 2026

DEVOPS · TRAINING PROGRAM

Java Devops

A practical, hands-on Java Devops curriculum: 541 topics. Learn live online in a small batch, as a private corporate batch, or self-paced with one year of LMS access.

LIVE ONLINE **Max 8-10 hours**	CORPORATE **Private team batch, on demand**	SELF-PACED VIDEO **1-year LMS access**
BATCH SIZE **Max 10 learners**	NEXT BATCHES **1st of every month · weekends: first Saturday**	CERTIFICATE **Verifiable certificate of completion**

INSIDE THIS CURRICULUM

Who should attend	Objectives, audience, prerequisites
Agenda	541 topics
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/devops/devops-training-course-Java.html
Talk to us: WhatsApp +91 99057 40781 · +1 (469) 756-6329 · contact@DevOpsSchool.com · reply within 1 hour

COURSE FIT

Who this program is for

Use this page to decide whether Java Devops matches your role and experience.

1

Objectives

Understand the core concepts and usage model, apply practical workflows through examples and guided tasks, and prepare for implementation, certification or team adoption.

2

Who should attend

Developers, DevOps engineers, administrators, SREs, security engineers, platform teams, release teams, technical leads and corporate teams working with this topic.

3

Before you start

Basic computer literacy and awareness of software, operations, cloud, DevOps or security workflows is helpful. Deep programming experience is not needed unless the topic requires it.

AGENDA AT A GLANCE · 541 TOPICS

Core curriculum

541 topics · Tools Covered

AGENDA · 541 TOPICS

Java Devops agenda

Live online batches run for max 8-10 hours; corporate batches are planned with your team.

CORE CURRICULUM — 541 topics

1. Tools Covered
2. Instructor-led, Live & Interactive Sessions
3. Level
4. Batches
5. Course Price at
6. Let’s Understand about Software Development Model
7. Overview of Waterfall Development Model
8. Challenges of Waterfall Development Model
9. Overview of Agile Development Model
10. Challenges of Agile Development Model
11. Requirement of New Software Development Model
12. Understanding an existing Pain and Waste in Current Software Development Model
13. What is DevOps?
14. Transition in Software development model
15. Waterfall -> Agile -> CI/CD -> DevOps -> DevSecOps
16. Understand DevOps values and principles
17. Culture and organizational considerations
18. Communication and collaboration practices
19. Improve your effectiveness and productivity
20. DevOps Automation practices and technology considerations
21. DevOps Adoption considerations in an enterprise environment
22. Challenges, risks and critical success factors
23. What is DevSecOps?
24. Let’s Understand DevSecOps Practices and Toolsets.
25. What is SRE?
26. Let’s Understand SRE Practices and Toolsets.
27. List of Tools to become Full Stack Developer/QA/SRE/DevOps/DevSecOps
28. Microservices Fundamentals
29. Microservices Patterns
30. Choreographing Services
31. Presentation components
32. Business Logic
33. Database access logic
34. Application Integration
35. Modelling Microservices
36. Integrating multiple Microservices

37. Keeping it simple
38. Avoiding Breaking Changes
39. Choosing the right protocols
40. Sync & Async
41. Dealing with legacy systems
42. Testing
43. What and When to test
44. Preparing for deployment
45. Monitoring Microservice Performance
46. Tools used for Microservices Demo using container
47. Centos,Ubuntu,VirtualBox,Vagrant, Bash Scripting - Concept
48. Ubuntu
49. Installing CentOS7 and Ubuntu
50. Accessing Servers with SSH
51. Working at the Command Line
52. Reading Files
53. Using the vi Text Editor
54. Piping and Redirection
55. Archiving Files
56. Accessing Command Line Help
57. Understanding File Permissions
58. Accessing the Root Account
59. Using Screen and Script
60. Overview of Hypervisor
61. Introduction of VirtualBox
62. Install VirtualBox and Creating CentOS7 and Ubuntu Vms
63. Vagrant
64. Understanding Vagrant
65. Basic Vagrant Workflow
66. Advance Vagrant Workflow
67. Working with Vagrant VMs
68. The Vagrantfile
69. Installing Nginx
70. Provisioning
71. Networking
72. Sharing and Versioning Web Site Files
73. Vagrant Share
74. Vagrant Status
75. Sharing and Versioning Nginx Config Files
76. Configuring Synced Folders
77. Cloud Computing using AWS Foundation Services

78. Introduction of AWS
79. Understanding AWS infrastructure
80. Understanding AWS Free Tier
81. IAM: Understanding IAM Concepts
82. IAM: A Walkthrough IAM
83. IAM: Demo & Lab
84. Computing:EC2: Understanding EC2 Concepts
85. Computing:EC2: A Walkthrough EC2
86. Computing:EC2: Demo & Lab
87. Storage:EBS: Understanding EBS Concepts
88. Storage:EBS: A Walkthrough EBS
89. Storage:EBS: Demo & Lab
90. Storage:S3: Understanding S3 Concepts
91. Storage:S3: A Walkthrough S3
92. Storage:S3: Demo & Lab
93. Storage:EFS: Understanding EFS Concepts
94. Storage:EFS: A Walkthrough EFS
95. Storage:EFS: Demo & Lab
96. Database:RDS: Understanding RDS MySql Concepts
97. Database:RDS: A Walkthrough RDS MySql
98. Database:RDS: Demo & Lab
99. ELB: Elastic Load Balancer Concepts
100. ELB: Elastic Load Balancer Implementation
101. ELB: Elastic Load Balancer: Demo & Lab
102. Networking:VPC: Understanding VPC Concepts
103. Networking:VPC: Understanding VPC components
104. Networking:VPC: Demo & Lab
105. Container using Docker & Docker Compose
106. What is Containerization?
107. Why Containerization?
108. How Docker is good fit for Containerization?
109. How Docker works?
110. Docker Architecture
111. Docker Installations & Configurations
112. Docker Components
113. Docker Engine
114. Docker Image
115. Docker Containers
116. Docker Registry
117. Docker Basic Workflow
118. Managing Docker Containers

119. Creating our First Image
120. Understading Docker Images
121. Creating Images using Dockerfile
122. Managing Docker Images
123. Using Docker Hub registry
124. Docker Networking
125. Docker Volumes
126. Deepdive into Docker Images
127. Deepdive into Dockerfile
128. Deepdive into Docker Containers
129. Deepdive into Docker Networks
130. Deepdive into Docker Volumes
131. Deepdive into Docker Volume
132. Deepdive into Docker CPU and RAM allocations
133. Deepdive into Docker Config
134. Docker Compose Overview
135. Install & Configure Compose
136. Understanding Docker Compose Workflow
137. Understanding Docker Compose Services
138. Writing Docker Compose Yaml file
139. Using Docker Compose Commands
140. Coding using Python & Django
141. Planning - Discuss some of the Small Project Requirement which include
142. Login/Registertration with Some Students records CRUD operations.
143. Design a Method --> Classes -> Interface using Core Python
144. Fundamental of Core Python with Hello-world Program with Method --> Classes
145. Coding in Flask using HTMl - CSS - JS - MySql
146. Fundamental of Flask Tutorial of Hello-World APP
147. UT - 2 Sample unit Testing using Pythontest
148. Package a Python App
149. AT - 2 Sample unit Testing using Selenium
150. Technology Demonstration
151. Software Planning and Designing using JAVA
152. Core Python
153. Flask
154. mySql
155. pytest
156. Selenium
157. HTMl
158. CSS
159. Js.

160. Coding using Java & Springboot
161. Planning - Discuss some of the Small Project Requirement which include Login/Registertration with Some Students records CRUD operations.
162. Design a Method --> Classes -> Interface using Core Java
163. Fundamental of Core Java with Hello-world Program with Method --> Classes -> Interface
164. Coding in SpringBoot using HTMl - CSS - JS - MySql
165. Fundamental of SpringBoot Tutorial of Hello-World APP
166. UT - 2 Sample unit Testing using Junit - Use Gradle or Maven
167. Package - War file - Use Gradle or Maven
168. AT - 2 Sample unit Testing using Selenium - Use Gradle or Maven
169. Core Java
170. SpringBoot
171. Junit
172. Planning and Designing - Jira + Confluence
173. Overview of Jira
174. Use cases of Jira
175. Architecture of Jira
176. Installation and Configuraration of Jira in Linux
177. Installation and Configuraration of Jira in Windows
178. Jira Terminologies
179. Understanding Types of Jira Projects
180. Working with Projects
181. Working with Jira Issues
182. Adding Project Components and Versions
183. Use Subtasks to Better Manage and Structure Your Issues
184. Link Issues to Other Resources
185. Working in an Agile project
186. Working with Issues Types by Adding/Editing/Deleting
187. Working with Custom Fields by Adding/Editing/Deleting
188. Working with Screens by Adding/Editing/Deleting
189. Searching and Filtering Issues
190. Working with Workflow basic
191. Introduction of Jira Plugins and Addons.
192. Jira Integration with Github
193. Exploring Confluence benefits and resources
194. Configuring Confluence
195. Navigating the dashboard, spaces, and pages
196. Creating users and groups
197. Creating pages from templates and blueprints
198. Importing, updating, and removing content
199. Giving content feedback

200. Watching pages, spaces, and blogs
201. Managing tasks and notifications
202. Backing up and restoring a site
203. Admin tasks
204. Add/Edit/Delete new users
205. Adding group and setting permissions
206. Managing user permissions
207. Managing addons or plugins
208. Customizing confluence site
209. Installing Confluence
210. Evaluation options for Confluence
211. Supported platforms
212. Installing Confluence on Windows
213. Activating Confluence trial license
214. Finalizing Confluence Installation
215. Source Code Versioning - Git using Github & GitOps
216. Introduction of Git
217. Installing Git
218. Configuring Git
219. Git Concepts and Architecture
220. How Git works?
221. The Git workflow
222. Working with Files in Git
223. Adding files
224. Editing files
225. Viewing changes with diff
226. Viewing only staged changes
227. Deleting files
228. Moving and renaming files
229. Making Changes to Files
230. Undoing Changes
231. Reset
232. Revert
233. Amending commits
234. Ignoring Files
235. Branching and Merging using Git
236. Working with Conflict Resolution
237. Comparing commits, branches and workspace
238. Working with Remote Git repo using Github
239. Push - Pull - Fetch using Github
240. Tagging with Git

241. Code Analysis & Securing Code (SAST) - SonarQube + SonarQube OWASP + Nginx
242. What is SonarQube?
243. Benefits of SonarQube?
244. Alternative of SonarQube
245. Understanding Various License of SonarQube
246. Architecture of SonarQube
247. How SonarQube works?
248. Components of SonarQube
249. SonarQube runtime requirements
250. Installing and configuring SonarQube in Linux
251. Basic Workflow in SonarQube using Command line
252. Working with Issues in SonarQube
253. Working with Rules in SonarQube
254. Working with Quality Profiles in SonarQube
255. Working with Quality Gates in SonarQube
256. Deep Dive into SonarQube Dashboard
257. Understanding Seven Axis of SonarQube Quality
258. Workflow in SonarQube with Maven Project
259. Workflow in SonarQube with Gradle Project
260. OWASP Top 10 with SonarQube
261. Build Management - Maven + Gradle
262. Maven
263. Introduction to Apache Maven
264. Advantage of Apache Maven over other build tools
265. Understanding the Maven Lifecycle and Phase
266. Understanding the Maven Goals
267. Understanding the Maven Plugins
268. Understanding the Maven Repository
269. Understanding and Maven Release and Version
270. Prerequisite and Installing Apache Maven
271. Understanding and using Maven Archetypes
272. Understanding Pom.xml and Setting.xml
273. Playing with multiples Maven Goals
274. Introducing Maven Dependencies
275. Introducing Maven Properties
276. Introducing Maven Modules
277. Introducing Maven Profile
278. Introducing Maven Plugins
279. How can Maven benefit my development process?
280. How do I setup Maven?
281. How do I make my first Maven project?

282. How do I compile my application sources?
283. How do I compile my test sources and run my unit tests?
284. How do I create a JAR and install it in my local repository?
285. How do I use plugins?
286. How do I add resources to my JAR?
287. How do I filter resource files?
288. How do I use external dependencies?
289. How do I deploy my jar in my remote repository?
290. How do I create documentation?
291. How do I build other types of projects?
292. How do I build more than one project at once?
293. Gradle
294. What is Gradle?
295. Why Gradle?
296. Installing and Configuring Gradle
297. Build Java Project with Gradle
298. Build C++ Project with Gradle
299. Build Python Project with Gradle
300. Dependency Management in Gradle
301. Project Structure in Gradle
302. Gradle Tasks
303. Gradle Profile and Cloud
304. Gradle Properties
305. Gradle Plugins
306. Package Management - Packer + Artifactory + JFrog XRay
307. Artifactory
308. Artifactory Overview
309. Understanding a role of Artifactory in DevOps
310. System Requirements
311. Installing Artifactory in Linux
312. Using Artifactory
313. Getting Started
314. General Information
315. Artifactory Terminology
316. Artifactory Repository Types
317. Artifactory Authentication
318. Deploying Artifacts using Maven
319. Download Artifacts using Maven
320. Browsing Artifactory
321. Viewing Packages
322. Searching for Artifacts

323. Manipulating Artifacts
324. Packer
325. Introduction to Selenium
326. Components of Selenium
327. Selenium IDE
328. Selenium Web driver
329. Selenium Grid
330. Installing and Configuring Selenium
331. Working with Selenium IDE
332. Working With Selenium Web driver with Java Test Case Setup and Working with Selenium Grid
333. Setup and Working with Selenium Grid
334. Jmeter
335. What Is Performance Testing?
336. Introducing JMeter
337. Configuring a Test Plan
338. Recording a Test
339. Assertions in Jmeter
340. Configuring a Dynamic Test
341. Collecting and Analyzing Test Results
342. Http cookie Managerto capture sessions
343. Jmeter controllers for Modules wise Metrics
344. Timers in Jmeter
345. Regular Expression in Jmeter
346. Data Driven testing in Jmeter
347. Handling Dynamic Responses
348. Jmeter NON GUI mode
349. Stress and Load Testing using JMeter
350. Scalability and Security Testing
351. Integration of JMeter with Selenium
352. Configuration & Deployment Management - Ansible
353. Overflow of Configuration Management
354. Introduction of Ansible
355. Ansible Architecture
356. Let’s get startted with Ansible
357. Ansible Authentication & Authorization
358. Let’s start with Ansible Adhoc commands
359. Let’s write Ansible Inventory
360. Let’s write Ansible Playbook
361. Working with Popular Modules in Ansible
362. Deep Dive into Ansible Playbooks
363. Working with Ansible Variables

364. Working with Ansible Template
365. Working with Ansible Handlers
366. Roles in Ansible
367. Ansible Galaxy
368. Infrastructure Coding - Terraform
369. Deploying Your First Terraform Configuration
370. Introduction
371. What's the Scenario?
372. Terraform Components
373. Updating Your Configuration with More Resources
374. Terraform State and Update
375. Data Type and Security Groups
376. Configuring Resources After Creation
377. Terraform Provisioners
378. Terraform Syntax
379. Adding a New Provider to Your Configuration
380. Terraform Providers
381. Terraform Functions
382. Intro and Variable
383. Resource Creation
384. Deployment and Terraform Console
385. Updated Deployment and Terraform Commands
386. Container Orchestration - Kubernetes & Helm Introduction
387. Understanding the Need of Kubernetes
388. Understanding Kubernetes Architecture
389. Understanding Kubernetes Concepts
390. Kubernetes and Microservices
391. Understanding Kubernetes Masters and its Component
392. kube-apiserver
393. etcd
394. kube-scheduler
395. kube-controller-manager
396. Understanding Kubernetes Nodes and its Component
397. kubelet
398. kube-proxy
399. Container Runtime
400. Understanding Kubernetes Addons
401. DNS
402. Web UI (Dashboard)
403. Container Resource Monitoring
404. Cluster-level Logging

405. Understand Kubernetes Terminology
406. Kubernetes Pod Overview
407. Kubernetes Replication Controller Overview
408. Kubernetes Deployment Overview
409. Kubernetes Service Overview
410. Understanding Kubernetes running environment options
411. Working with first Pods
412. Working with first Replication Controller
413. Working with first Deployment
414. Working with first Services
415. Introducing Helm
416. Basic working with Helm
417. Continuous Integration - Jenkins
418. Lets understand Continuous Integration
419. What is Continuous Integration
420. Benefits of Continuous Integration
421. What is Continuous Delivery
422. What is Continuous Deployment
423. Continuous Integration Tools
424. Integrations
425. Infrastructure
426. Host Map
427. Events
428. Dashboards
429. Develop an Instrumentation Strategy
430. Design and implement logging
431. Design and implement telemetry
432. IIntegrate logging and monitoring solutions
433. Develop a Site Reliability Engineering (SRE) strategy
434. Develop an actionable alerting strategy
435. Design a failure prediction strategy
436. Design and implement a health check
437. Design an authentication and authorization strategy
438. Design a sensitive information management strategy
439. Develop security and compliance
440. Design governance enforcement mechanisms
441. Manage source control
442. Develop a modern source control strategy
443. Plan and implement branching strategies for the source code
444. Configure repositories
445. Integrate source control with tools

446. Facilitate communication and collaboration
447. Communicate deployment and release information with business stakeholders
448. Generate DevOps process documentation
449. Automate communication with team members
450. Define and implement continuous integration
451. Design build automation
452. Design a package management strategy
453. Design an application infrastructure management strategy
454. Implement a build strategy
455. Maintain build strategy
456. Design a process for standardizing builds across organization
457. Define and implement a continuous delivery and release management strategy
458. Develop deployment scripts and templates
459. Implement an orchestration automation solution
460. Plan the deployment environment strategy
461. Log Monitoring Tool 1 - Splunk
462. What Is Splunk?
463. Overview
464. Machine Data
465. Splunk Architecture
466. Careers in Splunk
467. Introduction and Overview of NewRelic
468. What is Application Performance Management?
469. Understanding a need of APM
470. Understanding transaction traces
471. What is Application Performance?
472. APM Benefits
473. APM Selection Criteria
474. Why NewRelic is best for APM?
475. What is NewRelic APM?
476. How does NewRelic APM work?
477. NewRelic Architecture
478. NewRelic Terminology
479. Installing and Configuring NewRelic APM Agents for Application
480. Register a Newrelic Trial account
481. Installing a JAVA Agent to Monitor your Java Application
482. Installing a PHP Agent to Monitor your PHP Application
483. Installing New Relic Agent for .NET Framework Application
484. Installing a Docker based Agent to Monitor your Docker based Application
485. Understanding of NewRelic Configration settings of newrelic.yml
486. Understanding of NewRelic Agent Configration settings

487. Working with NewRelic Dashboard
488. Understanding a transactions
489. Understanding Apdex and Calculating and Setting Apdex Threshold
490. Understanding Circuit break
491. Understanding Throughput
492. Newrelic default graphs
493. Understanding and Configuring Service Maps
494. Understanding and Configuring JVM
495. Understanding Error Analytics
496. Understanding Violations
497. Understanding and Configuring Deployments
498. Understanding and Configuring Thread Profiler
499. Depp Dive into Transaction Traces
500. Profiling with New Relic
501. Creating and managing Alerts
502. Working with Incidents
503. Sending NewRelic Alerts to Slack
504. Assessing the quality of application deployments
505. Monitoring using Newrelic
506. View your applications index
507. APM Overview page
508. New Relic APM data in Infrastructure
509. Transactions page
510. Databases and slow queries
511. Viewing slow query details
512. External services page
513. Agent-specific UI
514. Viewing the transaction map
515. Istio Introduction - Concept
516. Environment Setup (Kubernetes cluster and Istio(1.7.X/1.8.0)) -
517. Istio Architecture
518. Traffic Management
519. Security -
520. Telemetry, Observability
521. Ingress Gateway: HTTP
522. Ingress Gateway: HTTPS/TLS
523. Ingress Gateway: mTLS, ciphersuites, TLS version - Concept
524. VirtualService
525. DestinationRule
526. Fault Injection
527. Request Timeout

528. Traffic Mirroring
529. Gateway Routing
530. Load Balancing
531. Egress HTTP
532. Egress TLS, mTLS
533. Service Entry
534. AuthorizationPolicy
535. Request Authentication
536. Peer Authentication
537. Envoy Filters
538. External Authorization
539. Rate Limiting
540. Egress Mongo Setup
541. Debugging with istioctl

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	This program plus 20+ more courses and 50+ tools, at your own pace.	**INR 833/month** billed yearly at INR 9,996
Corporate	Planned with your team	A private batch for your team, on demand, with examples mapped to your platforms.	**Quote on request**

ASSESSMENT AND CERTIFICATE

Your progress is checked through hands-on tasks and assignments. 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 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 courses: www.devopsschool.com/courses/

ENROL OR RESERVE A SEAT
Java Devops

WHATSAPP · INDIA
+91 99057 40781

WHATSAPP · USA
+1 (469) 756-6329

EMAIL
contact@DevOpsSchool.com

More DevOps training: www.devopsschool.com/courses/devops/