DevOpsSchool

Course curriculum · November 2026

LEVEL 04 OF 04 · ADVANCED

Json Advanced Training

Process large JSON fast and safely, evolve schemas without breaking clients and govern JSON contracts.

LIVE ONLINE **Max 8-10 hours**	CORPORATE (DAY-WISE) **5 days · Days 1-5**	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**

Foundation	Essential	Intermediate	Advanced
1 day · Included	2 days · Included	3 days · Included	5 days · This curriculum

INSIDE THIS CURRICULUM

What you will achieve	Outcomes and level focus
Who should attend	Audience and prerequisites
Agenda	48 topics · Foundation to Advanced
Hands-on labs	4 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/json/json-advance-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 Json course path, you will be able to:

1

Write correct JSON

Create, format and debug JSON documents that any parser accepts.

2

Automate JSON workflows

Query and transform JSON from APIs and cloud tools with jq, JSONPath and code.

3

Govern JSON contracts

Define, validate, secure and evolve JSON schemas used by APIs and data pipelines.

WHAT THE ADVANCED LEVEL ADDS

Performance, security and JSON at scale

Hands-on: Build a validated streaming JSON pipeline. Stream a multi-gigabyte JSON Lines export through a validating parser, reject bad records against a schema and publish clean records in a compact format.

WHO SHOULD ATTEND

- Experienced Json practitioners responsible for quality, security and scale.
- Senior engineers, architects and leads who design and troubleshoot Json solutions.
- Teams making Json an organisation-wide standard.

BEFORE YOU START

- The Intermediate level, or equivalent hands-on experience.
- You can already: Define JSON contracts with JSON Schema, validate them in CI and patch, sign and store JSON safely.
- 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 ADVANCED

Json Advanced Training agenda

48 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: JSON syntax, data types and validation

Included from Foundation

1. What JSON is: a text data-interchange format defined by RFC 8259 and ECMA-404, and where it is used
2. JSON compared: JSON vs XML vs YAML for readability, typing, comments and tooling
3. Syntax rules: objects, arrays, name/value pairs, double-quoted keys and no trailing commas
4. Data types: string, number, boolean, null, object and array, and what JSON cannot represent (dates, undefined)
5. Strings and escaping: quotes, backslashes, newlines, \u escapes and UTF-8 encoding
6. Numbers: integers, decimals, exponents and precision limits beyond 2^53 in JavaScript
7. Nesting: modelling real records with nested objects and arrays of objects
8. Formatting: pretty-printing and minifying with python3 -m json.tool and jq .
9. Common syntax errors: single quotes, comments, trailing commas and unescaped characters, linting in the editor
10. JSON files and media type: .json files, application/json and character encoding rules
11. JSON in the browser: inspecting API responses in the developer tools Network tab

DAY 2 — Essential: Parsing, generating and querying JSON

Included from Essential

12. JavaScript: JSON.parse() and JSON.stringify() with reviver, replacer and indentation arguments
13. Python: json.load(), json.loads(), json.dump() and json.dumps() with indent, sort_keys and ensure_ascii
14. Other languages: Jackson or Gson in Java, encoding/json in Go and System.Text.Json in .NET
15. Serialising objects: mapping classes to JSON, ISO 8601 dates and handling null or missing fields
16. jq fundamentals: filters, .[], select(), map(), keys, to_entries and building new objects
17. JSON on the command line: kubectl get -o json, aws --output json with --query (JMESPath) and curl piped to jq
18. JSONPath: RFC 9535 syntax, recursive descent, filter expressions and kubectl -o jsonpath
19. JSON in REST APIs: request and response bodies, Content-Type and Accept headers, error objects with status codes
20. Fetching JSON: fetch() with response.json(), Python requests .json() and handling invalid responses
21. JSON Lines (NDJSON): one record per line for logs and streaming, processed with jq -c
22. Converting formats: JSON to YAML or CSV with yq, jq @csv and Python
23. Configuration files: package.json, tsconfig.json, editor settings and JSON with comments (JSONC)

DAY 3 — Intermediate: JSON Schema, patching and data contracts

Included from Intermediate

24. JSON Schema basics: $schema, type, properties, required, enum and additionalProperties in draft 2020-12
25. Schema composition: $ref, $defs, allOf, anyOf, oneOf, if/then/else and format keywords
26. Validating in code and CI: Ajv in JavaScript, jsonschema in Python and failing builds on invalid data
27. OpenAPI 3.1 and JSON Schema: describing request and response bodies and generating API clients
28. JSON Pointer (RFC 6901) and JSON Patch (RFC 6902): add, remove, replace, move, copy and test operations
29. Merge strategies: JSON Merge Patch (RFC 7396) and Kubernetes strategic merge patch with kubectl patch
30. API design: consistent naming, pagination, envelopes and errors as RFC 9457 problem details

31. JSON Web Tokens: header, payload and signature, registered claims and verifying JWTs (RFC 7519)
32. JSON in PostgreSQL: json vs jsonb, the ->, ->> and @> operators and GIN indexes
33. Document databases: MongoDB documents, BSON types and querying nested fields
34. Structured logging: JSON log lines with consistent fields for Elasticsearch, Loki or CloudWatch
35. Troubleshooting: parse errors, byte order marks, encoding problems, duplicate keys and number precision bugs

DAYS 4-5 **Advanced: Performance, security and JSON at scale** **New at this level**

36. Large documents: streaming parsers such as ijson, Jackson streaming and stream-json instead of loading everything
37. Performance: fast parsers like simdjson and orjson, payload size and gzip or Brotli compression
38. Binary alternatives: BSON, CBOR, MessagePack and Protocol Buffers compared with JSON
39. Security: prototype pollution via __proto__, injection risks, safe parsing and never using eval()
40. Resource limits: maximum depth, size and number limits to prevent denial of service in parsers
41. Signing and encryption: JWS and JWE, key management and choosing algorithms
42. Schema evolution: API versioning, backward-compatible changes and deprecating fields
43. Linked data: JSON-LD and schema.org structured data for search engines
44. Data pipelines: JSON in Kafka topics, schema registries and conversion to Parquet for analytics
45. Policy as code: validating Kubernetes and Terraform JSON with OPA Rego or CUE
46. Code generation: TypeScript and other types from JSON Schema with tools such as quicktype
47. Governance: shared schema catalogues, naming standards and review of API contract changes
48. Capstone: design a versioned JSON API with schemas, validation, JWT auth and a streaming export

PRACTICE

Hands-on labs

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

FOUNDATION

Model and validate a course catalogue

Write a nested JSON document describing courses, levels and trainers, fix deliberate syntax errors and pretty-print the result with jq.

ESSENTIAL

Query a live API with jq

Call a public REST API with curl, extract and reshape fields with jq and JSONPath, then convert the result to CSV with a Python script.

INTERMEDIATE

Enforce a JSON Schema contract in CI

Write a JSON Schema for an API payload, validate fixtures with Ajv and jsonschema in a CI job, and apply JSON Patch updates to stored documents.

ADVANCED

Build a validated streaming JSON pipeline

Stream a multi-gigabyte JSON Lines export through a validating parser, reject bad records against a schema and publish clean records in a compact format.

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 Json levels, plus 20+ more courses and 50+ tools, at your own pace.	**INR 833/month** billed yearly at INR 9,996
Corporate (day-wise)	5 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 JSON 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/json/

ENROL OR RESERVE A SEAT
Json Advanced Training

WHATSAPP · INDIA
+91 99057 40781

WHATSAPP · USA
+1 (469) 756-6329

EMAIL
contact@DevOpsSchool.com