Elastic Logstash Kibana Full Stake (ELK Stack) Training

About DevOpsSchool

DevOpsSchool is a unit of "Cotocus PVT ltd" and a leading platform which helps IT organizations and professionals to learn all the emerging technologies and trend which helps them to learn and embrace all the skills, intelligence, innovation and transformation which requires to achieve the end result, quickly and efficiently. We provide over 40 specialized programs on DevOps, Cloud, Containers, Security, AI, ML and on Big data that are focused on industry requirement and each curriculum is developed and delivered by leading experts in each domain and aligned with the industry standards.

ABOUT COURSE

The ELK stack comprises of three words Elasticsearch, Logstash, and Kibana. Elasticsearch is a NoSQL database which is based on the Lucene search engine. Logstash is a log pipeline tool that accepts inputs from various sources, executes different transformations, and exports the data to different targets. Kibana is a visualization layer that works on top of Elasticsearch. These three different open source products are generally used in log analysis in IT environments.

DevOpsSchool offers exclusive ELK stack Training Program to various level of IT professionals. We have xperienced ELK stack Experts/Instructors to conduct classes online and offline to help the candidates to grab the skill set and it’s capability which can be utilized by them.

Co-coordinator – Akanksha Kumari

Call/WhatsApp: - +91 1800 889 7977

Mail Address: -

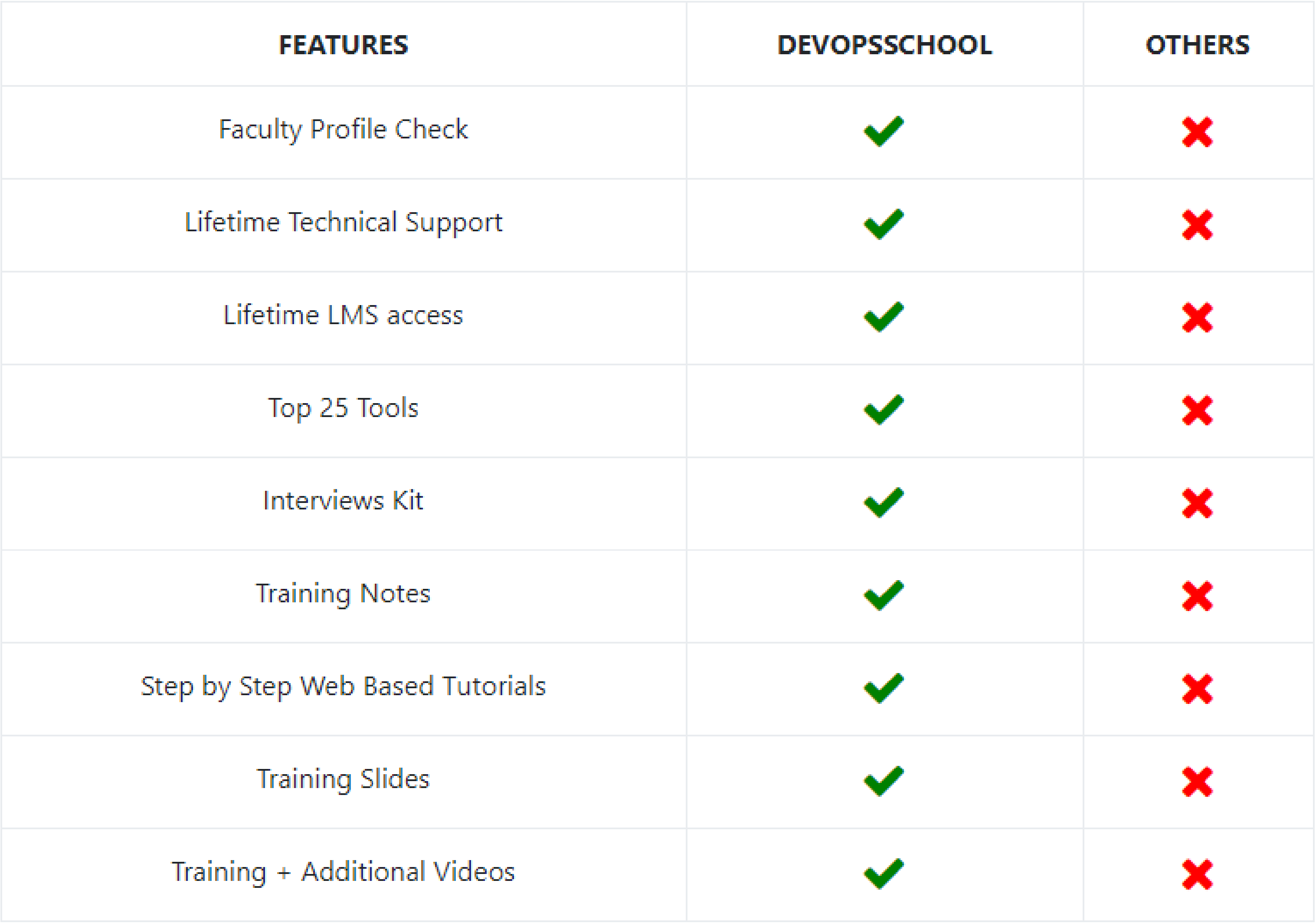
contact@DevOpsSchool.com

Secondary contact – Patrick

Call/WhatsApp: - +91 7004 215 841

Mail Address: -contact@DevOpsSchool.com

Duration	20 Hours
Mode	Online (Instructor-led, live & Interactive)
Projects (Real time scenario based)	1



FEATURES	DEVOPSSCHOOL	OTHERS
Faculty Profile Check	✓	✗
Lifetime Technical Support	✓	✗
Lifetime LMS access	✓	✗
Top 25 Tools	✓	✗
Interviews Kit	✓	✗
Training Notes	✓	✗
Step by Step Web Based Tutorials	✓	✗
Training Slides	✓	✗
Training + Additional Videos	✓	✗

AGENDA OF THE ELASTIC LOGSTASH KIBANA FULL STAKE (ELK STACK)

Getting Started

- Introduction to this course
- Introduction to Elasticsearch
- Overview of the Elastic Stack (ELK+)
- Elastic Stack

Architecture of Elasticsearch

- Introduction to this section
- Nodes & Clusters
- Nodes & Clusters
- Indices & Documents
- A word on types
- Another word on types
- Sharding
- Sharding
- 4 questions
- Replication
- Replication
- 6 questions
- Keeping replicas synchronized
- Searching for data
- Distributing documents across shards

Installing Elasticsearch & Kibana

- Running Elasticsearch & Kibana in Elastic Cloud
- Installing Elasticsearch on Mac/Linux
- Using the MSI installer on Windows
- Installing Elasticsearch on Windows
- Configuring Elasticsearch
- Installing Kibana on Mac/Linux
- Installing Kibana on Windows
- Configuring Kibana
- Kibana now requires data to be available
- Introduction to Kibana and dev tools

Managing Documents

- Creating an index
- Adding documents
- Retrieving documents by ID
- Replacing documents
- Updating documents
- Scripted updates
- Upserts
- Deleting documents
- Deleting indices
- Batch processing
- Importing test data with cURL
- Exploring the cluster

Mapping

- Introduction to mapping
- Dynamic mapping
- Meta fields
- Field data types
- Adding mappings to existing indices
- Changing existing mappings
- Mapping parameters
- Adding multi-fields mappings
- Defining custom date formats
- Picking up new fields without dynamic mapping

Analysis & Analyzers

- Introduction to the analysis process
- A closer look at analyzers
- Using the Analyze API
- Understanding the inverted index
- Analyzers
- Overview of character filters
- Overview of tokenizers
- Overview of token filters
- Overview of built-in analyzers
- Configuring built-in analyzers and token filters
- Creating custom analyzers
- Using analyzers in mappings
- Adding analyzers to existing indices
- A word on stop words

Introduction to Searching

- Search methods
- Searching with the request URI
- Introducing the Query DSL
- Understanding query results
- Understanding relevance scores
- Debugging unexpected search results
- Query contexts
- Full text queries vs term level queries
- Basics of searching s

Term Level Queries

- Introduction to term level queries
- Searching for a term
- Searching for multiple terms
- Retrieving documents based on IDs
- Matching documents with range values
- Working with relative dates (date math)
- Matching documents with non-null values
- Matching based on prefixes
- Searching with wildcards
- Searching with regular expressions
- Term Level Queries

Full Text Queries

- Introduction to full text queries
- Flexible matching with the match query
- Matching phrases
- Searching multiple fields
- Full Text Queries

Adding Boolean Logic to Queries

- Introduction to compound queries
- Querying with boolean logic
- Debugging bool queries with named queries
- How the “match” query works

Joining Queries

- Introduction to this section
- Querying nested objects
- Nested inner hits
- Mapping document relationships
- Adding documents
- Querying by parent ID
- Querying child documents by parent
- Querying parent by child documents
- Multi-level relations
- Parent/child inner hits
- Terms lookup mechanism
- Join limitations
- Join field performance considerations

Controlling Query Results

- Specifying the result format
- Source filtering
- Specifying the result size
- Specifying an offset
- Pagination
- Sorting results
- Sorting by multi-value fields
- Filters

Aggregations

- Introduction to aggregations
- Metric aggregations
- Introduction to bucket aggregations
- Document counts are approximate
- Nested aggregations
- Filtering out documents
- Defining bucket rules with filters
- Range aggregations
- Histograms
- Global aggregation
- Missing field values
- Aggregating nested objects

Improving Search Results

- Introduction to this section
- Proximity searches
- Affecting relevance scoring with proximity
- Fuzzy match query (handling typos)
- Fuzzy query
- Adding synonyms
- Adding synonyms from file
- Highlighting matches in fields
- Stemming

Building a Web Application Search Engine

- A quick note
- Introducing Application & Client Libraries
- Adding a simple query
- Paginating search results
- Adding fuzziness
- Aggregations & Filters
- Adding product details page

Thank you!

Connect with us for more info

Call/WhatsApp: - +91 968 682 9970

Mail: - contact@DevOpsSchool.com

www.DevOpsSchool.com