

## Day - 1

- Introduction to APM and its benefits
- Overview of key APM components, including agents, collectors, and dashboards
- Installation and configuration of an APM agent for a sample application
- Understanding and configuring transaction tracking in APM
- Introduction to Elastic APM and its benefits
- Installing and configuring Elastic APM agent for your application
- Understanding and configuring transaction types and naming conventions
- APM terminology
- Agent: A piece of software installed on the application server that collects performance data and sends it to the APM system for analysis.
- Transaction: A unit of work performed by the application, such as a user request or a database query.
- Trace: A detailed record of the sequence of events that occur during a transaction, including any external calls to other services or databases.
- Distributed tracing: The ability to track transactions across multiple services or systems, providing a complete end-to-end view of the transaction.
- Metrics: Numeric values that provide insight into the performance of the application, such as response time or error rate.
- Alerting: The ability to generate notifications when performance metrics fall outside of predefined thresholds.
- Dashboard: A visual representation of performance data, typically including graphs, charts, and tables.
- Service map: A graphical representation of the relationships between services and their dependencies.
- Profiling: The process of analyzing the performance of individual functions or methods within the application code.
- Log correlation: The ability to link APM data with log data to provide additional context and insight into application performance issues.

## Day - 2

- Exploring the Elastic APM dashboard and visualizations
- Analyzing transaction performance and identifying bottlenecks
- Introduction to distributed tracing and its benefits
- Configuring and analyzing distributed tracing in Elastic APM
- Correlating transaction traces with logs and metrics in Elasticsearch
- Using Elastic APM with popular frameworks like Spring, Node.js, and Django

- Best practices for optimizing your application performance using Elastic APM
- Analyzing APM data and identifying performance bottlenecks
- Introduction to distributed tracing and its benefits
- Configuring and analyzing distributed tracing in APM
- Best practices for optimizing application performance using APM

## Day - 3

- **Understanding and configuring error tracking and logging in Elastic APM**
- **Using custom metrics and spans to track specific application behavior**
- **Advanced features like service maps, anomaly detection, and alerting**

- **Integrating Elastic APM with other Elastic Stack tools like Elasticsearch, Kibana, and Logstash**
- **Q&A and troubleshooting common issues with Elastic APM**