

## Achieving better performance, scalability and maintenance using Java 8

### Workshop Details:

<b>Duration :</b>	3 Days
<b>Objectives and Take away :</b>	<p>Deep dive in new features (Lambda expressions, streams, Date API) and comparison with equivalent old features(of java 7), exploiting Java 8 for better performance and scalability through parallel processing (i.e. multi-CPU systems) etc], rigorous hands on.</p> <p><b>Take away:</b> Usage of features of Java 8 for harnessing power of multi-core system. Enhancing performance of application using these features.</p>
<b>Participants' Entry Profile :</b>	The participants must have prior experience with Java 6/7.
<b>Training Methodology :</b>	<p>The workshop will follow Synergetics methodology of</p> <ul style="list-style-type: none"> <li>Concept Visualization</li> <li>Active Experimentation</li> <li>Application Development</li> </ul> <p>The workshop will be 100% Hands-On with each participant having access to system during the session</p>

### Setup Requirements

<b>Hardware and Software Requirements:</b>	<p>Participant's as well as Trainer's Machine are required to have :</p> <p><b>Hardware</b></p> <ul style="list-style-type: none"> <li>Intel Core i3 2.2 GHz</li> <li>CD Rom Drive</li> <li>LCD Color Monitor</li> <li>2 GB RAM</li> <li>LAN Connectivity</li> </ul> <p><b>Software</b></p> <ul style="list-style-type: none"> <li>Windows XP or 7</li> <li>JDK 1.8 Latest</li> <li>Netbeans 8 IDE OR Eclipse JSE Luna with Java 8 support.</li> </ul> <ul style="list-style-type: none"> <li>• The installable must match in bits architecture of the machine.</li> <li>• A shared disk-space among participants and trainer for trainer to share training stuff with participants.</li> <li>• Internet connection to fetch maven dependencies .</li> </ul>
<b>Training Lab Requirements:</b>	<p>Whiteboard 6 feet by 4 feet (minimum)</p> <p>Whiteboard markers – Red, Blue, Green, Black</p> <p>Video Projector (1024 X 768 resolutions)</p>

### Pre-requisite for training

A skill-set equivalent to Synergetics-Programming in Java with JSE-7(L1\_01\_JSE-7)

- Working with basic Programming construct.
- Application Development using OOPS.
- Making robust application by Handling Errors and Exception
- In-built classes in Java API
- Java Collections Framework & choosing right collection in a scenario
- Multithreading and Concurrency API
- JAVA IO Support and Serialization.
- Java Reflection, Annotations, Enumeration
- Networking & Logging in an JAVA Application
- Connecting, Querying and Transaction management using the Java Database Connectivity

### Course Content

<b>Day 1</b>	<ul style="list-style-type: none"> <li>➤ <b>Module 1: Lambda Expressions</b> <ul style="list-style-type: none"> <li>▪ Why Functional Style code</li> <li>▪ Syntax of Lambda</li> <li>▪ Functional interfaces</li> <li>▪ Built-In functional interfaces</li> <li>▪ Runnable, ActionListener</li> <li>▪ Lambda Expressions for functional interfaces</li> <li>▪ Default methods</li> <li>▪ Creating user-defined functional interface</li> </ul> </li> <li>➤ <b>Module 2: Collections enhancements</b> <ul style="list-style-type: none"> <li>▪ Iterating Lists using "Consumer"</li> <li>▪ Iterating List using "Lambda"</li> <li>▪ Transforming list using "Lambda"</li> </ul> </li> <li>➤ <b>Module 3: Stream API</b> <ul style="list-style-type: none"> <li>▪ What are new "streams"?</li> <li>▪ Streams and Collections</li> <li>▪ Transforming list using streams</li> <li>▪ Finding Element using streams</li> <li>▪ Picking an element</li> </ul> </li> </ul>
<b>Day 2</b>	<ul style="list-style-type: none"> <li>➤ <b>Module 4: Concurrency Enhancements</b> <ul style="list-style-type: none"> <li>▪ Concurrency in Stream API</li> <li>▪ Serial &amp; Parellel Streams</li> <li>▪ Bulk Data operations</li> <li>▪ Concurrency updates</li> <li>▪ Scalable update variables</li> <li>▪ fork and join improvements</li> <li>▪ Stamped Lock</li> </ul> </li> <li>➤ <b>Module 5: New Date/Time API</b> <ul style="list-style-type: none"> <li>▪ LocalDate &amp; LocalDateTime</li> <li>▪ TemporalAdjuster to perform date operations</li> </ul> </li> </ul>

**Day 3**

- **Module 6: Java 8 Profiles**
- **Module 7: JDBC Enhancements**
  - REF\_CURSOR
  - DriverAction interface
  - Additional SQL TYPES
  - RowSet changes
  
- **Module 8: Scripting enhancements**
  - Hashhorn JavaScript engine
  - invoking script from java code
  - JDBC Enhancements & JJS Tools