

Note:

This is an old document prepared as needed for a client engagement (identified as CLIENT in the document) by Software View in September, 2007. Some topics need updates since they have evolved over time. Furthermore, some of the trivial sections of the original proposal have been removed.

Proposal For

Comprehensive Java Focused On-Boarding Training

September, 2007

Includes Java SE 6, J2EE, Java EE 5, OOP, SQL + RDBMS, Coding Standards, Best Practices and Patterns, Test Infected Programming with JUnit, Eclipse, Ant, Hibernate, Spring, The Internet, WWW, Networking Fundamentals, HTML, JavaScript, CSSv2, Struts, Tiles, JavaServer Faces, AJAX and XML/Web Services

by
Software View
<http://www.swview.org>

Contact Person (Software View):
Kamal Wickramanayake
Software View
No 2, 2nd Lane
Pangiriwatta Road, Gangodawila
Nugegoda, Sri Lanka

Email: kamal@swview.org
Tel: 0602 182933

Table of Contents

Section 1: Comprehensive Java Focused On-Boarding Training.....	2
1.1:Description.....	2
1.2:Modules And Objectives.....	2
1.3:Target group.....	7
1.4:Prerequisites.....	7
1.5:Duration.....	7
1.6:Additional notes.....	7
Section 2:Curriculum Of The Training.....	8
Section 3:Sample Project.....	34
Section 4:Evaluation Of Participants.....	34
Section 5:About Training Material.....	35
Section 6:Training Delivery Modes.....	36
Section 7:Code Of Conduct For Participants.....	36
Section 8:Facilities To Be Provided By CLIENT.....	37
Section 9:Pricing.....	38
Rationale.....	38
Section 10:Quality Guarantee.....	38
Appendix A: An Overall View Of The Comprehensive Java Focused On-Boarding Training.....	39

Section 1: Comprehensive Java Focused On-Boarding Training

1.1: Description

The rising complexities of modern enterprise computing systems and growing customer demand for high quality and well integrable software solutions necessitate the development companies to set up internal controls to guarantee and strengthen their ability to deliver the expectations with tight constraints. Hence, creating skill enhancement opportunities and conditioning the developers with expert knowledge beyond the creation of mere learning sessions provide a meaningful platform for a development company to excel and reduce direct and indirect costs.

This comprehensive Java training is a well crafted learning opportunity to meet a specific need: to equip the freshers into Java Standard Edition and Java Enterprise Edition with pragmatic development experience so that their competency in the technologies and performance will significantly be enhanced so that they will be able to work in live projects.

The training is comprehensive in its coverage of core Java technologies (Java Standard Edition and Java Enterprise Edition) focusing on multiple versions of them. Furthermore, this training brings a number of additional frameworks, development tools, related technologies and best practices together to guarantee that the participants will be conversant, productive and professional with the “real world” software development.

Software View has extended the curriculum proposed by CLIENT based on our extensive training delivery background. The curriculum was further enhanced by CLIENT to match with specific needs in several rounds of discussions. As suggested by CLIENT, the training has also been fine tuned to fulfill the knowledge requirements for the latest SCJP certification from Sun Microsystems.

Based on our experience, we are confident that CLIENT will be able to use the participants not just as shadow developers, but as developers with significant productivity.

1.2: Modules And Objectives

The high level objective of this training is

“to equip the freshers into Java Standard Edition and Java Enterprise Edition with pragmatic development experience so that their competency in the technologies and performance will significantly be enhanced so that they will be able to work in live projects and not just as freshers .”

The training includes 18 modules each spanning into one or more sections and subsection as detailed in section 4 of this document. Following are the descriptions and the specific objectives of each module.

Module 1: Object Orientation

Java is a pure object orientation programming (OOP) language and hence the developers are required to have a firm understanding of what OOP is and how to apply them correctly. This module provides this vitally required foundation of object orientation.

At the end of this module, participants will be able to

1. Define the meanings and purposes of Object Oriented Programming (OOP) fundamentals with sound clarity
2. Apply OOP principles correctly
3. Apply supplementary Object Orientation (OO) best practices in system design

Module 2: Introduction To Java

Anyone into serious Java development should know the Java evolution process at least to keep up with future trends. This evolution process itself is governed by the Java Community Process. Should anyone is interested, he or she can contribute to the process.

At the end of this module, participants will be able to

1. Describe how Java evolved into this stage and future trends
2. Describe the operation of Java Community Process (JCP)

Module 3: Java Standard Edition

Java Standard Edition covers the core language and developers with no exception should first be conversant with the language to use higher level application environments like Java Enterprise Edition. This module deals with the core language.

At the end of this module, participants will be able to

1. Use Java Standard Edition in enterprise grade application development
2. Sit for the latest SCJP certification (Java Standard Edition 5) from Sun Microsystems

Module 4: Java Standard Edition (JFC/Swing)

JFC/Swing is the part of Java Standard Edition facilitating graphical user interface (GUI) creation. This module is a sufficient introduction to the Swing library, but does not go into fine grained details of UI customizations. Yet, participants will be able to work with default UI components with ease.

At the end of this module, participants will be able to

1. Develop moderately complex graphical user interfaces with JFC/Swing

Module 5: Coding Standards

Adherence to coding standards is a must in software development unless the development is done for no real purpose. Coding standards facilitates the developers to comprehend source code easily and hence plays a major role in collaborative development environments, issue correction and other maintenance activities.

At the end of this module, participants will be able to

1. Organize Java source files properly
2. Use standard Java formatting conventions in source files

3. Utilize standard Java naming conventions

Module 6: Development Tools

Productivity in development has a high positive correlation with the ability to use development tools properly and utilize their features. Eclipse is a world renowned tooling platform. Ant has been a major build automation tool. For Java developers, these two freely available tools are indispensable.

At the end of this module, participants will be able to

1. Use Eclipse IDE with ease and compassion
2. Use Ant as an automated enterprise application build tool

Module 7: Test Infected Programming

Test Infected Programming (also known as Test First Programming) is a style of development and a development best practice. It encourages development of automated tests while the application code is being developed. The results are: possible development time reductions, enhanced quality code, reduced bugs, automated tests that are repeatable with no extra cost and ease of software maintenance.

At the end of this module, participants will be able to

1. Describe the benefits of test infected programming

Practice test infected programming with JUnit testing framework

Module 8: Structured Query Language + Database

Relational databases have become the norm of persistence mechanism in enterprise applications. This module covers the theories behind them, Structured Query Language which is used to access and modify data and also techniques to improve the performance of such databases and applications.

At the end of this module, participants will be able to

1. Use relational databases for persistence
2. Create informed table schemas
3. Use Structured Query Language (SQL) and stored procedures
4. Enhance the performance of a given relational database and an application by basic improvement of queries, basic schema changes, indexing etc.

Module 9: Hibernate

Hibernate has grown to be one of the most extensively used Object-Relational mapping tool in Java applications. This module first introduces the concepts behind Object-Relational mapping and then the way Hibernate is used.

At the end of this module, participants will be able to

1. Describe different object-relational mapping techniques

2. Use Hibernate for persistence

Module 10: Spring Framework

Spring is another framework that has gained significant popularity in Java applications. Spring framework implements a development best practice/pattern known as “dependency injection”. Spring encourages and improves software modularity and hence reuse and software reconfigurability with minimal overhead.

At the end of this module, participants will be able to

1. Use Spring framework for enhanced application modularity
2. Use transaction abstraction layer of Spring framework

Module 11: The Internet/World Wide Web

At the root of modern enterprise applications is the network provisioning and collaboration enactment contributed by both the Internet and the World Wide Web. Hence, enterprise application developers should know the way they operate and evolution to cope with increasingly network dependent software systems.

At the end of this module, participants will be able to

1. Describe the evolution of the Internet, World Wide Web
2. Use networking mechanisms of the Internet as applied for enterprise application development with Java
3. Create web pages with HTML, JavaScript and Cascading Style Sheets (CSS)

Module 12: Java Enterprise Edition

Java Enterprise Edition encapsulates a large number of Java and non-Java technologies together and proposes a robust software architecture for enterprise applications. One major portion of it describes how server side components can generate a dynamic web-based user interface (Servlets/JavaServer Pages). The other major portion describes implementing business logic in server side components (Enterprise Beans).

At the end of this module, participants will be able to

1. Develop moderately complex secure web applications with using JavaBeans, JavaServer Pages, and Servlets
2. Use, configure and manage JBoss and Tomcat application servers
3. Develop secure and transaction aware business tier components with Enterprise Java Beans

Module 13: Java Enterprise Edition Patterns

Complexity of enterprise applications are best managed by using patterns that are popular among development communities. This module introduces a collection of patterns that are applicable to Java enterprise applications.

At the end of this module, participants will be able to

1. Describe the importance of patterns in enterprise application development
2. Apply patterns and best practices in multi-tier application development

Module 14: Java Enterprise Edition 5

Java Enterprise Edition version 2, commonly known as J2EE was a huge success. Java Enterprise Edition 5 is a major revision of J2EE. It dramatically cuts down development time and some of the pain points of J2EE.

At the end of this module, participants will be able to

1. Define the differences between J2EE and Java Enterprise Edition 5
2. Use Java Enterprise Edition 5 in the web tier and business tier
3. Use Java Persistence API (JPA) for persistence

Module 15: Struts And Tiles

Struts is a framework for Java web applications that implements a best practice/pattern known as Model-View-Controller architecture. Tiles is both a templating mechanism and an approach to improve code (JavaServer Pages) reuse in the web tier. Together, they provide a foundation for robust, highly scalable and configurable web tier user interface implementations.

At the end of this module, participants will be able to

1. Develop Model-View-Controller pattern based web applications with Struts
2. Develop template based web applications with Tiles

Module 16: JavaServer Faces

JavaServer Faces (JSF) is yet another MVC based UI development framework which is presently being embraced by Java application developers. JSF is the successor of Struts framework.

At the end of this module, participants will be able to

1. Describe JSF component model and navigation model
2. Use JSF for web tier user interface development

Module 17: XML/Web Services

XML and Web Services represent two highly utilized technologies in modern software systems. They improve the interoperability of heterogeneous computer systems by proposing a standard set of documents, exchange protocols and related services. XML based Web Services is an enabler of Service Oriented Architecture which describes the present paradigm of computer systems utilization in enterprises.

At the end of this module, participants will be able to

1. Describe how XML is used in Java Enterprise Edition
2. Develop Web Services
3. Develop clients (Web Service consumers)

4. Describe the role of Web Services in Service Oriented Architecture (SOA)

Module 18: Asynchronous JavaScript (AJAX)

Going against conventional web based user interface development, AJAX has increasingly gained momentum as an intuitive technique to implement rich user interfaces. This module covers the concepts behind AJAX and how AJAX is used to improve the user experience.

At the end of this module, participants will be able to

1. Describe how AJAX operates and the concepts behind
2. Use AJAX in web applications

1.3: Target group

Freshers (with some programming exposure)

1.4: Prerequisites

Some level of computer programming ability

1.5: Duration

30 Days (from 9.00 am - 5.00 pm)

1.6: Additional notes

1. This training fulfills the knowledge requirements for the latest SCJP certification - Sun Certified Programmer for the Java 2 Platform, Standard Edition 5.0 (CX-310-055). Participants will be able to sit for the exam immediately after the completion of this training.
1. The training offers hands-on lab experience packed with expert knowledge.
2. A sample project will be used to strengthen the learning process. This project will be articulated prior to the commencement of the training with input/approval from CLIENT.
3. For all modules, there will be exercises. Details of the exercises will be added prior to the commencement of the training.
4. Delivery of the training will be learner centric. Multiple delivery techniques will be used - not just lectures.
5. Participants will go through two MCQ based quizzes to assess their learnings. The first quiz will be based on Java Standard Edition and the related topics. The second will focus on Java Enterprise Edition and the related.
6. Participants are expected to reinforce their learnings by spending some more time beyond usual training hours. From time to time, home work will also be given.

Section 2: Curriculum Of The Training

Following is the full curriculum of the training presented according to the sections covered on each day. See the appendix for an overall view of the training.

Apart from the lunch break, a morning and evening tea breaks will also be given each of which may be of 10 – 15 minute in length.

Day: 1	Module Coverage: Coding Standards, Introduction To Java, Java Standard Edition, Object Orientation
09:00 - 11:00 (2 hours)	Getting Started With Object Oriented Programming (OOP) <ul style="list-style-type: none">• History of programming language paradigms• Purpose of OOP fundamentals• Meaning of abstraction, encapsulation, inheritance, polymorphism
11:00 - 12:00 (1 hour)	Getting Started With Java <ul style="list-style-type: none">• History of Java• The Java platform• Java evolution into various areas• Java Community Process (JCP)• Future of Java• Writing the first Java program
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 14:00 (1 hour)	Working With Java Source Files <ul style="list-style-type: none">• Comments• Package statement• Import statements• Formatting conventions• Naming conventions
14:00 - 15:00 (1 hour)	Object Oriented Programming With Java <ul style="list-style-type: none">• Defining a class• Method declaration• Field declaration• Creating objects and using them
15:00 - 17:00 (2 hours)	Variables/Data types

- Primitive and reference data types
- Static, instance and local variables
- Arrays
- Effect of modifiers

Day: 2	Module Coverage: Java Standard Edition
09:00 - 12:00 (3 hours)	Methods <ul style="list-style-type: none"> • Method declaration • Pass by value and pass by reference • Methods with variable length argument list • Access methods • Constructors • Effect of modifiers
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 14:00 (1 hour)	Flow Control <ul style="list-style-type: none"> • if/switch statements • Loops (for, for-each, while, do-while, labels, break, continue)
14:00 - 16:00 (2 hours)	Operators <ul style="list-style-type: none"> • Assignment operators • Arithmetic operators • Relational operators • instanceof operator • Logical operators • Conditional operator • Determining equality of two objects or two primitives
16:00 - 17:00 (1 of 3 hours)	Abstract Classes And Interfaces <ul style="list-style-type: none"> • Abstract classes • Parent/Child classes • Method overriding and overloading • Covariant return types • Interfaces and implementing interfaces <p>Continued... (See below)</p>

Day: 3	Module Coverage: Development Tools, Java Standard Edition, Object Orientation
09:00 - 11:00 (2 of 3 hours)	Continuation From Above (Abstract Classes And Interfaces)
11:00 - 12:00 (1 of 2 hours)	Object Oriented Programming In Action (With Java) <ul style="list-style-type: none"> • Implementing Encapsulation • Implementing Abstraction • Implementing Inheritance (Class, Interface) • Implementing Polymorphism <p>Continued... (See below)</p>
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 14:00 (1 of 2 hours)	Continuation From Above (Object Oriented Programming In Action (With Java))
14:00 - 15:00 (1 hour)	Eclipse Integrated Development Environment (IDE) - I <ul style="list-style-type: none"> • Background of Eclipse Platform • Eclipse capabilities beyond an IDE • Concepts (Workbench, workspace, perspectives, editors, views) • Organization of a Java project • Compiling a project • Running a project
15:00 - 17:00 (2 of 4 hours)	Error Handling with Exceptions <ul style="list-style-type: none"> • Overview (What & Why) • Catching/Handling Exceptions (try-catch-finally blocks) • Throwing Exceptions • Creating Own Exception Classes • Runtime Exceptions, checked exceptions, errors • Commonly encountered exceptions (ArrayIndexOutOfBoundsException, ClassCastException...) <p>Continued... (See below)</p>

Day: 4	Module Coverage: Java Standard Edition
09:00 - 11:00 (2 of 4 hours)	Continuation From Above (Error Handling with Exceptions)
11:00 - 12:00 (1 of 2 hours)	Some Commonly Used Classes <ul style="list-style-type: none">• Primitive wrapper classes• Autoboxing, unboxing• String/StringBuilder/StringBuffer classes Continued... (See below)
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 14:00 (1 of 2 hours)	Continuation From Above (Some Commonly Used Classes)
14:00 - 16:00 (2 hours)	Assertions <ul style="list-style-type: none">• What are assertions?• How to use assertions?• Proper and inappropriate use of assertions
16:00 - 17:00 (1 hour)	Enumerations <ul style="list-style-type: none">• What are enumerations?• Using enumerations

Day: 5	Module Coverage: Java Standard Edition
09:00 - 11:00 (2 hours)	Annotations <ul style="list-style-type: none"> • Purpose of annotations • Using existing annotations • Defining new annotations • Annotation processing
11:00 - 12:00 (1 of 3 hours)	Nested Classes <ul style="list-style-type: none"> • What is nesting of classes? • Different types of class nesting • Goods/bads of nested classes <p>Continued... (See below)</p>
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 15:00 (2 of 3 hours)	Continuation From Above (Nested Classes)
15:00 - 16:00 (1 hour)	Other Training Activity <ul style="list-style-type: none"> • Reserved block for other activity (e.g. Participants - CLIENT Senior interaction)
16:00 - 17:00 (1 of 2 hours)	About Command Line, CLASSPATH And JAR Files <ul style="list-style-type: none"> • Using command line • CLASSPATH • Using/Creating JAR files <p>Continued... (See below)</p>

Day: 6	Module Coverage: Development Tools, Java Standard Edition
09:00 - 10:00 (1 of 2 hours)	Continuation From Above (About Command Line, CLASSPATH And JAR Files)
10:00 - 12:00 (2 of 3 hours)	<p>Garbage Collection/Memory Management</p> <ul style="list-style-type: none"> • Overview of Garbage Collector • Different Garbage Collection Algorithms and Their Effect (Reference Counting, Mark and Sweep, Heap Compaction, Generational Garbage Collection, Train Collectors) • finalize() Method of java.lang.Object class • Forcing Finalization with System.gc() <p>Continued... (See below)</p>
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 14:00 (1 of 3 hours)	Continuation From Above (Garbage Collection/Memory Management)
14:00 - 16:00 (2 hours)	<p>Fine Tuning JVM</p> <ul style="list-style-type: none"> • Analyzing Garbage Collector operations • Selecting a suitable Garbage collector • Application Launcher Standard/Non-Standard Options • Controlling heap size
16:00 - 17:00 (1 of 2 hours)	<p>Eclipse IDE - II</p> <ul style="list-style-type: none"> • Using different builders • Application launchers • Refactoring features • Debugging features <p>Continued... (See below)</p>

Day: 7	Module Coverage: Development Tools, Object Orientation
09:00 - 10:00 (1 of 2 hours)	Continuation From Above (Eclipse IDE - II)
10:00 - 12:00 (2 hours)	Building With Ant <ul style="list-style-type: none">• Build automation with Ant• Anatomy of a build file• Ant tasks• Using Ant inside Eclipse
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 17:00 (4 hours)	Programming Best Practices With A Focus On OOP <ul style="list-style-type: none">• Tight encapsulation, loose coupling, and high cohesion in classes• Working with different forms of relationships between objects• What is an interface?• Design by contract• Open-Close Principle

Day: 8	Module Coverage: Java Standard Edition, Test Infected Programming
09:00 - 12:00 (3 of 4 hours)	<p>Test Infected Programming with JUnit</p> <ul style="list-style-type: none"> • Overview of Unit Testing • Different Unit Testing Mechanisms and Role of JUnit • Writing TestCases and TestSuites • Testing for Exceptions • Best Practices of Using JUnit • Overview of JUnit Extensions • Using JUnit inside Eclipse <p>Continued... (See below)</p>
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 14:00 (1 of 4 hours)	Continuation From Above (Test Infected Programming with JUnit)
14:00 - 17:00 (3 of 5 hours)	<p>I/O Manipulation</p> <ul style="list-style-type: none"> • Overview of I/O Streams (Readers/Writers/Input Streams/Output Streams) • Using File Streams • Using Piped Streams • Wrapping Streams • Buffered Streams • Filtered Streams • Object serialization/de-serialization (ObjectInputStream, ObjectOutputStream, Serializable) • Brief Introduction to nio Package <p>Continued... (See below)</p>

Day: 9	Module Coverage: Java Standard Edition
09:00 - 11:00 (2 of 5 hours)	Continuation From Above (I/O Manipulation)
11:00 - 12:00 (1 of 4 hours)	Formatting And Parsing Of Strings And Streams <ul style="list-style-type: none">• Regular expressions• java.util.regex package (Pattern, Matcher)• Formatter and Scanner classes and PrintWriter.format/printf methods Continued... (See below)
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 16:00 (3 of 4 hours)	Continuation From Above (Formatting And Parsing Of Strings And Streams)
16:00 - 17:00 (1 of 4 hours)	Internationalization/Localization <ul style="list-style-type: none">• Developing code with internationalization in mind• What is a locale?• Purpose and use of Locale class• Using java.text package Continued... (See below)

Day: 10	Module Coverage: Java Standard Edition
09:00 - 12:00 (3 of 4 hours)	Continuation From Above (Internationalization/Localization)
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 14:00 (1 hour)	Other Training Activity <ul style="list-style-type: none">• Reserved block for other activity (e.g. Participants - CLIENT Senior interaction)
14:00 - 17:00 (3 of 6 hours)	Collections Framework <ul style="list-style-type: none">• Overview of Collections Framework• Collection/Set/List/Map Interfaces• Implementations available in Java SE API• Generics• Limitations of the non-generic Collections API• Difference between using methods with type parameters and generic methods• Comparable/Comparator interfaces• hashCode() and equals() methods• Difference between == operator and equals() method• Using java.util package facilities to manipulate collections Continued... (See below)

Day: 11	Module Coverage: Java Standard Edition
09:00 - 12:00 (3 of 6 hours)	Continuation From Above (Collections Framework)
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 17:00 (4 hours)	<p>Multi-Threading</p> <ul style="list-style-type: none"> • Overview (Application/Process/Thread/Multi-Threading/Multi-Processing/Multi-Processor) • Thread Class • Runnable Interface • Thread Priority (Pre-Emptive and Time Slicing, Thread.yield(), Thread.sleep()) • Different Thread Synchronization Mechanisms (Synchronization blocks, Poling, wait and notify) • Thread Grouping • Inter-Thread Communication (Shared data, Pipes)

Day: 12	Module Coverage: Structured Query Language + Database
09:00 - 12:00 (3 of 10 hours)	<p>Relational Databases And Structured Query Language (SQL)</p> <ul style="list-style-type: none"> • Relational databases and Entity-Relation model • Normalization/De-normalization • SQL as a query language • Creating schema • CRUD operations (Insert, Select, Update, Delete) • Joins • Subqueries • Stored procedures • Basic performance enhancement techniques <p>Continued... (See below)</p>
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 17:00 (4 of 10 hours)	<p>Continuation From Above (Relational Databases And Structured Query Language (SQL))</p> <p>Continued... (See below)</p>

Day: 13	Module Coverage: Java Standard Edition, Structured Query Language + Database
09:00 - 12:00 (3 of 10 hours)	Continuation From Above (Relational Databases And Structured Query Language (SQL))
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 17:00 (4 hours)	Java Database Connectivity (JDBC) API <ul style="list-style-type: none"> • JDBC and Drivers • Making a Connection • Display a Result Set • Statements and Result Sets • Iterating through ResultSet • Creating a Prepared Statement • Handling Transactions • Batch Updates • Locking

Day: 14	Module Coverage: Hibernate, Java Standard Edition (JFC/Swing)
09:00 - 12:00 (3 of 4 hours)	Object Relational Management With Hibernate <ul style="list-style-type: none"> • Different OR mapping techniques • Persistence with Hibernate <p>Continued... (See below)</p>
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 14:00 (1 of 4 hours)	Continuation From Above (Object Relational Management With Hibernate)
14:00 - 15:00 (1 hour)	Other Training Activity <ul style="list-style-type: none"> • Reserved block for other activity (e.g. Participants - CLIENT Senior interaction)
15:00 - 17:00 (2 of 4 hours)	Getting Started With Graphical User Interface (GUI) Development <ul style="list-style-type: none"> • From Abstract Window Toolkit (AWT) to Swing • Anatomy of a simple Swing application • Concepts (Containers, Components, LayoutManagers, LookAndFeel) • Swing thread model <p>Continued... (See below)</p>

Day: 15	Module Coverage: Java Standard Edition (JFC/Swing)
09:00 - 11:00 (2 of 4 hours)	Continuation From Above (Getting Started With Graphical User Interface (GUI) Development)
11:00 - 12:00 (1 of 4 hours)	Using Different Swing Components <ul style="list-style-type: none">• Model-View-Controller implementation of Swing components• Label, TextField, Button, List• Table, Tree• Menu Continued... (See below)
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 16:00 (3 of 4 hours)	Continuation From Above (Using Different Swing Components)
16:00 - 17:00 (1 of 3 hours)	Working With Layout Managers <ul style="list-style-type: none">• BorderLayout• FlowLayout• GridLayout• BoxLayout• Absolute positioning without a layout manager Continued... (See below)

Day: 16	Module Coverage: Java Standard Edition (JFC/Swing), Spring Framework
09:00 - 11:00 (2 of 3 hours)	Continuation From Above (Working With Layout Managers)
11:00 - 12:00 (1 of 3 hours)	Swing Event Model <ul style="list-style-type: none">• Delegation event model• Using different event handlers• Correct multi-threading approaches Continued... (See below)
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 15:00 (2 of 3 hours)	Continuation From Above (Swing Event Model)
15:00 - 17:00 (2 of 4 hours)	Spring Framework <ul style="list-style-type: none">• Dependency injection pattern (or IOC)• Transaction management abstraction layer• What else is there?• Wiring an application with Spring Continued... (See below)

Day: 17	Module Coverage: Spring Framework, The Internet/World Wide Web
09:00 - 11:00 (2 of 4 hours)	Continuation From Above (Spring Framework)
11:00 - 12:00 (1 of 4 hours)	<p>Introduction To The Internet</p> <ul style="list-style-type: none"> • Evolution of the Internet • Packet switching networking • Protocol stack • Differences between TCP (Transmission Control Protocol) and UDP (User Datagram Protocol) • Internet services (Web, Email, File Transferring, Remote Login) • Common application level protocols (HTTP, SMTP, POP, FTP) • The role of Internet Engineering Task Force (IETF) • Internet standards and Request For Comments (RFCs) <p>Continued... (See below)</p>
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 16:00 (3 of 4 hours)	Continuation From Above (Introduction To The Internet)
16:00 - 17:00 (1 of 4 hours)	<p>Domain Name System (DNS)</p> <ul style="list-style-type: none"> • Who assigns IP addresses? • The need to use host names • Domain registration • DNS hosting • Forward/reverse address resolution • Canonical host names • DNS and email transfer mechanism <p>Continued... (See below)</p>

Day: 18	Module Coverage: The Internet/World Wide Web
09:00 - 12:00 (3 of 4 hours)	Continuation From Above (Domain Name System (DNS))
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 14:00 (1 hour)	Useful Tools <ul style="list-style-type: none">• ping• traceroute• nslookup• telnet• ftp
14:00 - 17:00 (3 hours)	World Wide Web (WWW) <ul style="list-style-type: none">• Past, present and future of WWW• Impact of Google, MySpace, YouTube, SecondLife• Search engines• Roles of HTTP and HTML• Analyzing HTTP traffic using a network protocol analyzer• World Wide Web Consortium (W3C)

Day: 19	Module Coverage: Java Standard Edition, The Internet/World Wide Web
09:00 - 12:00 (3 hours)	Networking With Java <ul style="list-style-type: none">• Overview of networking with Java• Using the Socket class• Using the ServerSocket class• Multi-threading considerations
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 16:00 (3 hours)	Introduction To HTML, JavaScript And Cascading Stylesheets <ul style="list-style-type: none">• Evolution of HTML• Implementing simple HTML pages• JavaScript as a client side programming language• JavaScript document model• Using JavaScript in web pages• Introduction to style sheets• Using Cascading Style Sheets V2
16:00 - 17:00 (1 hour)	Other Training Activity <ul style="list-style-type: none">• Reserved block for other activity (e.g. Participants - CLIENT Senior interaction)

Day: 20	Module Coverage: Java Enterprise Edition
09:00 - 10:00 (1 hour)	Introduction To Java Enterprise Edition (Java EE) <ul style="list-style-type: none">• Overview of distributed computing (Centralized, Client-Server, n-Tier)• Java EE architecture• Java EE technologies• Past, present and the future of Java EE
10:00 - 12:00 (2 hours)	Getting Ready For Web Application Development <ul style="list-style-type: none">• Setting up Integrated Development Environment (Eclipse + Web Tools Platform)• Setting up application server (Tomcat)• Setting up database (MySQL)• Getting familiar with the tools
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 17:00 (4 hours)	JSP/Servlet Basics <ul style="list-style-type: none">• Java facilities for web application development• Implementing a Servlet• Life cycle of a Servlet• Implementing a JavaServer Page (JSP)• Life cycle of a JavaServer Page• Servlet API• Chaining and Filtering• JSP Syntax

Day: 21	Module Coverage: Java Enterprise Edition
09:00 - 12:00 (3 of 4 hours)	<p>Writing Basic Web Applications</p> <ul style="list-style-type: none"> • HTTP requests and responses • HTTP status codes • Cookies • URL parameters • Form handling • Error handling • Event logging <p>Continued... (See below)</p>
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 14:00 (1 of 4 hours)	Continuation From Above (Writing Basic Web Applications)
14:00 - 16:00 (2 hours)	<p>Session Tracking</p> <ul style="list-style-type: none"> • Scoped objects • Session tracking with URL rewriting • Session tracking with persistent cookies • Session tracking with user authentication credentials • The Session Tracking API
16:00 - 17:00 (1 of 3 hours)	<p>Security In Web Tier</p> <ul style="list-style-type: none"> • Programmatic and declarative security • Authentication (HTTP Basic, Form Based) • Authorization • HTTPS <p>Continued... (See below)</p>

Day: 22	Module Coverage: Java Enterprise Edition
09:00 - 11:00 (2 of 3 hours)	Continuation From Above (Security In Web Tier)
11:00 - 12:00 (1 of 4 hours)	Writing Complex Web Applications <ul style="list-style-type: none">• Using JavaBeans• Custom Tag Libraries (Jakarta Taglibs)• Java Standard Tag Library• Internationalization and localization Continued... (See below)
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 16:00 (3 of 4 hours)	Continuation From Above (Writing Complex Web Applications)
16:00 - 17:00 (1 of 2 hours)	Best Practices In Web Application Development <ul style="list-style-type: none">• Model 1 and Model 2 architectures• How Model 2 (MVC) web application implementations differ from MVC based widget creation?• Patterns in the web tier• Request processing in a multi-tier application Continued... (See below)

Day: 23	Module Coverage: Java Enterprise Edition, Struts And Tiles
09:00 - 10:00 (1 of 2 hours)	Continuation From Above (Best Practices In Web Application Development)
10:00 - 12:00 (2 of 6 hours)	<p>Struts and Tiles</p> <ul style="list-style-type: none"> • Struts implementation of MVC pattern • Overview of templating with Tiles <p>Continued... (See below)</p>
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 17:00 (-3 of 6 hours)	Continuation From Above (Struts and Tiles)

Day: 24	Module Coverage: JavaServer Faces
09:00 - 12:00 (3 of 6 hours)	<p>JavaServer Faces (JSF)</p> <ul style="list-style-type: none"> • JSF UI component model and navigation model • Life cycle of a JSF page • JSF and JSP • JSF based development <p>Continued... (See below)</p>
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 16:00 (3 of 6 hours)	Continuation From Above (JavaServer Faces (JSF))
16:00 - 17:00 (1 hour)	<p>Other Training Activity</p> <ul style="list-style-type: none"> • Reserved block for other activity (e.g. Participants - CLIENT Senior interaction)

Day: 25	Module Coverage: Asynchronous JavaScript (AJAX), Java Enterprise Edition, XML/Web Services
09:00 - 11:00 (2 hours)	Asynchronous JavaScript (AJAX) <ul style="list-style-type: none"> • What is AJAX? • Implementing a simple AJAX based page • What are the AJAX frameworks around?
11:00 - 12:00 (1 of 4 hours)	XML Technologies And Web Services <ul style="list-style-type: none"> • Introduction to XML technologies in Java EE • JAX-RPC • Web service description language • Writing, deploying, testing a Web service • Web services as an enabler of Service Oriented Architecture (SOA) <p>Continued... (See below)</p>
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 16:00 (3 of 4 hours)	Continuation From Above (XML Technologies And Web Services)
16:00 - 17:00 (1 hour)	Introduction To Enterprise Java Beans Technology <ul style="list-style-type: none"> • Overview of distributed computing with Enterprise Beans • Advantages of EJB Technology • Types of Enterprise Beans • Life cycle of Enterprise Beans • Client types

Day: 26	Module Coverage: Java Enterprise Edition, Java Enterprise Edition Patterns
09:00 - 10:00 (1 hour)	Setting Up The Development Environment <ul style="list-style-type: none"> • Installation of JBoss application server • Getting familiar with JBoss • Configuration settings in Eclipse IDE
10:00 - 12:00 (2 hours)	Session Beans <ul style="list-style-type: none"> • Writing stateless/stateful session beans • Packaging • Deployment • Writing a client
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 16:00 (3 hours)	Java Naming And Directory Interface (JNDI) <ul style="list-style-type: none"> • What is JNDI? • Difference between a naming service and a directory • JNDI architecture • Accessing resources with JNDI • JNDI naming conventions
16:00 - 17:00 (1 of 4 hours)	Java Enterprise Edition Patterns <ul style="list-style-type: none"> • Session Facade • Service Locator • Data Access Object • Transfer Object • Business Delegate • Web service broker <p>Continued... (See below)</p>

Day: 27	Module Coverage: Java Enterprise Edition, Java Enterprise Edition Patterns
09:00 - 12:00 (3 of 4 hours)	Continuation From Above (Java Enterprise Edition Patterns)
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 17:00 (4 hours)	Persistence In The EJB Tier (EJB 2.x) <ul style="list-style-type: none"> • Persistence with Entity Beans • Bean managed and container managed persistence (BMP/CMP) • Writing CMP Entity Beans • Container managed relationships (CMR) • Enterprise JavaBeans Query Language

Day: 28	Module Coverage: Java Enterprise Edition
09:00 - 12:00 (3 of 4 hours)	Message-Driven Beans (MDB) <ul style="list-style-type: none"> • Role of middleware in enterprise application integration • Purpose of JMS and MDBs • Point-to-Point and Publisher-Subscriber Messaging • Coding an MDB (Unified Domains) • JMS client coding (Unified Domains) <p>Continued... (See below)</p>
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 14:00 (1 of 4 hours)	Continuation From Above (Message-Driven Beans (MDB))
14:00 - 17:00 (3 hours)	Transaction Management <ul style="list-style-type: none"> • ACID properties of a transaction • Isolation levels • Bean Managed Transactions (BMT) • Container Managed Transactions (CMT) • Distributed transaction management with 2-phase committing

Day: 29	Module Coverage: Java Enterprise Edition, Java Enterprise Edition 5
09:00 - 12:00 (3 hours)	Security in EJB Tier <ul style="list-style-type: none">• Terminology• Programmatic and declarative security• Responsibilities of Bean Developer/Application Assembler/Deployer• Scope of methods
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 14:00 (1 hour)	Getting Started With Java Enterprise Edition 5 <ul style="list-style-type: none">• Differences between J2EE and Java EE 5• Setting up the development environment (NetBeans IDE)
14:00 - 16:00 (2 hours)	Enterprise Bean Development <ul style="list-style-type: none">• Session Bean Development• Message Driven Bean Development
16:00 - 17:00 (1 of 6 hours)	Java Persistence API (JPA) <ul style="list-style-type: none">• Persistence in web-tier• Persistence in business tier• Defining entity classes• Managing entities and their relationships• Java Persistence Query Language <p>Continued... (See below)</p>

Day: 30	Module Coverage: Java Enterprise Edition 5
09:00 - 12:00 (3 of 6 hours)	Continuation From Above (Java Persistence API (JPA)) Continued... (See below)
12:00 - 13:00 (1 hour)	Lunch Break
13:00 - 15:00 (2 of 6 hours)	Continuation From Above (Java Persistence API (JPA))
15:00 - 16:00 (1 hour)	XML-based Web Services <ul style="list-style-type: none">• Java API for XML-based Web Services (JAX-WS)
16:00 - 17:00 (1 hour)	Other Training Activity <ul style="list-style-type: none">• Reserved block for other activity (e.g. Participants - CLIENT Senior interaction)

Section 3: Sample Project

Apart from code fragments participants will be writing during the training, they will also build a sample application from the scratch to solidify their learnings.

The sample application will facilitate the participants to use all the major tools, techniques, frameworks and best practices taught during the training.

Participants will be asked to work on the sample project not during the usual (9.00am – 5.00pm) training hours.

Work on the sample project will be taken into consideration for the evaluation as indicated below.

Section 4: Evaluation Of Participants

Participants will be evaluated in two ways:

1. Two comprehensive MCQ based quizzes will be administered to assess their learnings. The first quiz will be based on Java Standard Edition and the related topics. The second will focus on Java Enterprise Edition and the related.
2. Success in the sample project

Furthermore, the trainer will provide a detailed report on each participant. But such comments should not be considered full scale evaluations due to the limited duration the trainer will interact with each participant.

Section 5: About Training Material

Software View will provide training material on a CDROM. Specifications, API documentation and links to on-line resources will also be included in this CDROM. Exceptions are:

1. Usage of slides (e.g. PowerPoint) will almost be zero during the training and hence the CDROM will not contain such material. But if ever used, they will be included in the CDROM.
2. The comprehensive quizzes will also be not included in the CDROM and not given to CLIENT in any form as they are assets of Software View.

A document in the CDROM will include links to the resource material in the following form where each topic of the training will be provided with specific references:

- Testing for Exceptions
- Other Important Classes in the JUnit Framework
- Best Practices of Using JUnit
- Overview of JUnit Extensions
- JUnit with Eclipse
- Point participants to a simple Maven tutorial and ask them to look at reports (code coverage, unit tests)

Resources:

1. [JUnit Test Infected: Programmers Love Writing Tests](#)
A small tutorial by the authors of JUnit.
2. [JUnit: A Cook's Tour](#)
Describes the internal architecture of JUnit.

Garbage Collection/Memory Management

- Overview of Garbage Collector
- Different Garbage Collection Algorithms and Their Effect (Reference Counting, Mark and Sweep, Heap)
- finalize() Method of java.lang.Object class
- Forcing Finalization with System.gc()

Resources:

1. [Java theory and practice: A brief history of garbage collection](#)
Different garbage collection algorithms are described here.
2. [Java theory and practice: Garbage collection in the 1.4.1 JVM](#)
Generational garbage collection in JVM

Section 6: Training Delivery Modes

The trainer will utilize multiple delivery modes. They will include:

1. Usual lectures delivered by the trainer
2. Participant presentations
3. Case studies
4. Fact finding sessions
5. Revision sessions

Section 7: Code Of Conduct For Participants

All participants are expected to behave with due respect to others in a training session. Participants' behavior in a training session should not in any circumstances trouble the other participants or disrupt achieving the training objectives. Before the training starts, participants will be informed of what is accepted and what is not. This does not mean that participants cannot oppose, argue with others. Participants are more than welcome to do so, but with care not to hurt others physically or mentally.

The trainer will inform a participant if he or she was noticed with unacceptable behavior. In extreme situations, such a participant will be asked to leave the training.

Depending on the circumstance, the trainer may inform CLIENT management about such situations.

Section 8: Facilities To Be Provided By CLIENT

1. The classroom
2. Computers for the participants
 - A Microsoft operating system or Linux
 - 1GB RAM recommended
 - Above 2.0 GHz recommended
 - Should be in a network environment preferably with Internet connectivity
3. Multimedia projector
 - A screen may or may not be required based on the training room
4. White board/ ~5 marker pens (several colors)
5. A bundle of A4 papers
6. A flip chart (A stand is not required, but paper)

Section 9: Pricing

Software View offers a preferred price for CLIENT and a preferred scheme for this x week long training.

REMOVED

Rationale

1. We value this training at least at Rs. xxx per participant. Above pricing scheme will still be favorable for CLIENT even if the training is valued at a lower price.
2. We have significantly reduced the price to cater for the facilities provided by CLIENT and a further reduction has been done as a favor for CLIENT hoping that we might deliver more trainings for CLIENT in future.
3. Training delivered by a reputed trainer who has conducted about 150 trainings for fresh and expert software developers in off-shore software service providing companies. The trainer is highly conversant with pragmatic aspects as well.
4. Software View feels that no one else in Sri Lanka is presently capable of delivering a broad training like the proposed.

Section 10: Quality Guarantee

Software View is dead serious about meeting client expectations. Hence, we offer a guarantee that no one else would ever think of:

Due to any fault of Software View, if the expected training objectives as identified in this document are not achieved or the quality of the training is realized to be poor than expected, CLIENT may reduce a portion of the agreed payment to be made to Software View. The amount to be reduced will unilaterally be determined by CLIENT. Software View will not question or trouble anyone in CLIENT about such reductions.

If such a reduction is preferred, CLIENT will write a cheque valued after the reduction and simply include a note describing the reason for us to understand the issue and improve our services:

1. The note will not carry generic statements like “It was not value the money”, but be descriptive enough to nail down the exact issue or issues.
2. At least two executives representing CLIENT (e.g. CTO, GM, MD,...) will sign such a note with the company stamp.

Appendix A: An Overall View Of The Comprehensive Java Focused On-Boarding Training

No	Module/Sections	Duration/(hrs)
1	Object Orientation <ul style="list-style-type: none"> • Getting Started With Object Oriented Programming (OOP) • Object Oriented Programming In Action (With Java) • Programming Best Practices With A Focus On OOP 	8
2	Introduction To Java <ul style="list-style-type: none"> • Getting Started With Java 	1
3	Coding Standards <ul style="list-style-type: none"> • Working With Java Source Files 	1
4	Java Standard Edition <ul style="list-style-type: none"> • Object Oriented Programming With Java • Variables/Data types • Methods • Flow Control • Operators • Abstract Classes And Interfaces • Error Handling with Exceptions • Some Commonly Used Classes • Assertions • Enumerations • Annotations • Nested Classes • About Command Line, CLASSPATH And JAR Files • Garbage Collection/Memory Management • Fine Tuning JVM • I/O Manipulation • Formatting And Parsing Of Strings And Streams • Internationalization/Localization • Collections Framework • Multi-Threading • Java Database Connectivity (JDBC) API • Networking With Java 	63
5	Development Tools	5

	<ul style="list-style-type: none"> • Eclipse Integrated Development Environment (IDE) - I • Eclipse IDE - II • Building With Ant 	
6	Test Infected Programming <ul style="list-style-type: none"> • Test Infected Programming with JUnit 	4
7	Structured Query Language + Database <ul style="list-style-type: none"> • Relational Databases And Structured Query Language (SQL) 	10
8	Hibernate <ul style="list-style-type: none"> • Object Relational Management With Hibernate 	4
9	Java Standard Edition (JFC/Swing) <ul style="list-style-type: none"> • Getting Started With Graphical User Interface (GUI) Development • Using Different Swing Components • Working With Layout Managers • Swing Event Model 	14
10	Spring Framework <ul style="list-style-type: none"> • Spring Framework 	4
11	The Internet/World Wide Web <ul style="list-style-type: none"> • Introduction To The Internet • Domain Name System (DNS) • Useful Tools • World Wide Web (WWW) • Introduction To HTML And JavaScript 	15
12	Java Enterprise Edition <ul style="list-style-type: none"> • Introduction To Java Enterprise Edition (Java EE) • Getting Ready For Web Application Development • JSP/Servlet Basics • Writing Basic Web Applications • Session Tracking • Security In Web Tier • Writing Complex Web Applications 	43

	<ul style="list-style-type: none"> • Best Practices In Web Application Development • Introduction To Enterprise Java Beans Technology • Setting Up The Development Environment • Session Beans • Java Naming And Directory Interface (JNDI) • Persistence In The EJB Tier (EJB 2.x) • Message-Driven Beans (MDB) • Transaction Management • Security in EJB Tier 	
13	Struts And Tiles <ul style="list-style-type: none"> • Struts and Tiles 	6
14	JavaServer Faces <ul style="list-style-type: none"> • JavaServer Faces (JSF) 	6
15	Asynchronous JavaScript (AJAX) <ul style="list-style-type: none"> • Asynchronous JavaScript (AJAX) 	2
16	XML/Web Services <ul style="list-style-type: none"> • XML Technologies And Web Services 	4
17	Java Enterprise Edition Patterns <ul style="list-style-type: none"> • Java Enterprise Edition Patterns 	4
18	Java Enterprise Edition 5 <ul style="list-style-type: none"> • Getting Started With Java Enterprise Edition 5 • Enterprise Bean Development • Java Persistence API (JPA) • XML-based Web Services 	10
Other Training Related Activities		6
Total Duration		210 (30 days)