

## JAVA TRAINING CURRICULUM

### The object paradigm

- Object-oriented (OO) programming
- Encapsulation, inheritance and polymorphism
- OO analysis and design: "Is a" and "Has a" relationships
- Designing an OO application step by step
- Diagramming object structure with Unified Modeling Language (UML)

### Introduction to Java

- High level Java architecture
- Description of why it is platform independent, interpreted etc.
- Introduction to the JVM, JRE, JDK
- Introduction to the Java landscape
- What is J2SE, J2EE, J2ME, JMX, Java Web Service...
- Describe the concept of a specification and implementation that lays the ground to describe the various Java technologies and give examples of some of the specifications and the most popular implementations.
- Describe at a high level the components of each of these Java technologies and where they fit into the overall platform architecture.
- Describe the "client side" vs "server side" technologies, web vs non-web technologies.
- Introduce the various Java APIs - Servlets, JSP, JSTL, JSF,
- EJBs, JAX-RPC, SOAP, JWS (Web services), JMS, JDBC, JTA, JTS, JCE, JSSE, JAXB etc.
- At a high level describe Weblogic, Websphere, JBoss, tomcat etc.
- At a high level describe the most popular open source and where they fit in. Struts, Spring, Hibernate, Axis, MyFaces etc.

## Core Java

### Core Java fundamentals

- Basic process of how a Java program is written compiled and run.
- Describe a very basic Java source file. Class, methods, properties, package, imports etc.
- Glance at the language keywords, literals, primitive data types.
- Details of java operators and assignments used on primitive types.
- Arrays - declaration, initialization and usage
- Method calls, by value, by reference semantics for primitive types.
- If statement, loops, switch, assert statement.
- Regular Expressions
- Details of class declarations and the allowed class modifiers.
- Method & variable declarations and their allowed modifiers.
- General Object Oriented principles.
- Overloading, overriding, constructors.
- Basic J2SE classes, their hierarchy and String, Math, Wrappers.
- Object Serialization.

### Java IO, Java NIO.

- Streams, Readers and Writers
- Accessing files
- Formatting text output

### Files and Directories

- Creating, deleting and renaming files
- Obtaining directory and file information

### Other important J2SDK API classes.

- Property files, Resource Bundles

### Internationalization and Localization

### Nested classes.

## **Collections.**

## **Multithreading concepts and Java Threads.**

## **JVM structure, heap, stack, garbage collection.**

## **Java Generics API, annotations.**

- Apache Commons and Collections API
- Design Patterns
- Introduction to design patterns
- All design patterns(GoF) normally used in J2SE

## **Advanced Java Topics**

- Java Reflection API.
- Java Networking API.
- Java Security API – JCE, JSSE.

## **JDBC API details, different types of drivers.**

- Connecting to a database
- Connection pooling
- Submitting SQL statements
- Retrieving and processing data
- Introduction to Datasources.
- A high level introduction to ORM using Hibernate (details will be later on)

## **J2EE**

- Bird's eye view of a typical J2EE architecture.

## **Java Web tier**

- HTTP basics. GET, POST requests, common HTTP headers.
- Java Servlets
  - J2EE Servlet container description. Role of Servlets.

- A basic Servlet example to show all the pieces involved in developing a Servlet, configuration files, code etc.
- The Servlet Model
- Servlet life cycle
- Servlet API details.
- ServletConfig, ServletContext,
- RequestDispatcher
- Servlet Deployment Details
- Servlet Container model
- Servlet "event" listeners.
- Servlet Filters
- Servlet Session Management
- Developing secure web applications

## JSP

- JSP technology basics
- JSP syntax, tags
- JSP page life cycle
- JSP page directive attributes
- JSP translation process
- JSP implicit variables and objects
- JSP page scopes
- Expression Language (EL)
- Using Java Beans in JSP
- Using custom tags
- Understanding tag libraries
- Using custom tags in jsp pages
- Understanding JSTL

## Advanced JSP

- Developing custom tag libraries
- Web tier deployment using standard Servlet containers (Tomcat, Weblogic, Websphere).
- EJB precursor

## JNDI implementation and usage details

- How to make a JNDI call
- How to bind to JNDI

## RMI implementation and usage details

- How to build a RMI client and server.

## EJBs

- Understanding the EJB architecture
- EJB container overview
- Session Beans
- Client's view of stateless and stateful session beans
- Life cycle of stateless and stateful session beans

## Entity beans

- Client's view of stateless and stateful session beans
- Life cycle of stateless and stateful session beans
- Entity Bean relationships
- EJB query language

## JMS & MDBs

- JMS overview
- Client's view of stateless and stateful session beans
- Life cycle of stateless and stateful session beans

## EJB services

- Transaction service, JTA, JTS.
- Security service

## **EJB distributed architecture**

### **Implementing EJBs in Weblogic/WebSphere/Jboss**

### **Introduction to EJB 3.0 (mainly JPA)**

### **Java – XML**

- Introduction to XML, XSD, DTD
- XML SAX, DOM parsing, xerces parser usage
- XML parsing using xstream/jdom
- JAXB introduction (Implementation details later on during JAXWS)

### **Introduction to J2EE design patterns**

- Along with the other patterns, introduction to the MVC model.

### **Struts**

#### Struts 1.3.x

- Description of how Struts adheres to the MVC model
- Building model components
- Building view components
- Building controller components
- Struts configuration details, how to configure Struts modules etc.
- Struts plugins

### **Tiles plugin, velocity plugin**

#### Tiles & Velocity

- What are tiles and how do we implement presentation layer with tiles.

- More details of how struts view components are built using

tiles

- Velocity Templates

### **Struts 2.x**

- Introduction to Struts 2.x

## Java – Web Services

- Introduction to web services architecture

## SOAP

- SOAP Basics
- SOAP Messages Layer

## RPC Style, Document Style

- SOAP Data Layer
- SOAP Transport Layer
- SOAP Bindings

## WSDL

## SAAJ – Sending SOAP messages programmatically.

## JAX – RPC

## Apache AXIS SOAP implementation

- How to call a web service using Apache Axis
- How to implement a web service using Apache Axis
- JAXB implementation details
- Introduction to UDDI
- Web Services Security

## SOA

- Introduction to Service Oriented Architecture and how JWS is used to realize it.
- WS-I Basic Profile description
- Introduction WS-BPEL.

## UML

- Introduction to UML,
- Class diagrams, sequence diagrams, collaboration diagrams.

## AJAX

- Introduction to XmlHttpRequest and asynchronous processing

- DWR implementation of AJAX
- Introduction to JSON & Google web toolkit

## **Hibernate**

- ORM Basics
- Hibernate Architecture
- Deploying and configuring Hibernate
- Creating Hibernate Applications
- Basic Hibernate O/R mapping
- Developing the persistent class
- Defining the Hibernate mapping rules
- Storing and retrieving Java objects

## **Hibernate Collection & Component mapping**

- Persisting and retrieving collections
- Preserving collection order for data integrity
- Specifying one-to-many and many-to-many relationships
- Controlling association life cycle

## **Hibernate inheritance mapping**

- Applying class rules for inheritance
- Techniques for class-database mapping

## **Hibernate session management**

- Establishing a thread-safe session object
- Defining object states: transient, persistent, detached

## **Hibernate transaction management and concurrency HQL & Criteria queries**

- Selecting and filtering queries
- Improving structure with named queries
- Augmenting HQL with native SQL

## **Hibernate performance tuning ,Spring, Spring & AOP introduction**



- IOC/Dependency injection concepts
- Spring architecture
  - Spring Beans, bean factory, application context
  - Spring typical configuration and an example Spring config file
  - Spring AOP details
  - Point cuts, Advices, interceptors
  - Creating Spring proxies
  - Transaction Management with Spring
  - Using an external app server transaction manger.
  - Configuring Spring for appropriate transaction isolation and propagation.
  - Details of transaction synchronization with Spring.

### Data persistence with Spring

- Spring's support for JDBC

### Using JdbcTemplate

- Spring Hibernate Integration
- Using HibernateTemplate

### Spring MVC Framework

- Building on the Spring controllers
- Resolving views with ViewResolvers
- Spring integration with Struts (and tiles)
- Using and implementing EJBs with Spring.
- Spring's support for JMS.

### JSF

- Introduction to JSF specs
- Using Apache MyFaces

### JMX

- Introduction to JMX

- Implementing Simple MBeans.

## Ant/Maven

- How to write Ant scripts and the built-in tasks
- Introduction to Maven, plug-in and the standard layout structure