



V-Soft Consulting Group Inc.,

Java Curriculum

- **Java Language Environment**

- Object Oriented
- Platform Independent
- Automatic Memory Management
- Compiled/Interpreted Approach
- Robust
- Secure
- Dynamic Linking
- Multithreaded
- Built-In Networking

- **Essentials Of Object-Oriented Programming**

- Object and Class Definition
- Using Encapsulation to Combine Methods and Data in a Single Class
- Inheritance and Polymorphism

- **Java Fundamentals**

- Data types
- Operators
- Control Statements
- Arrays
- Enhanced for Loop
- Enumerated types
- Static Import
- Auto Boxing
- C-Style Formatted I/O
- Variable Arguments

- **Generics**

- Introduction to Generics
- Using Built-In Generics Collections
- Writing Simple Generic Class
- Bounded Generics
- Wild Card Generics

- **Writing Java Classes**

- Encapsulation
- Polymorphism
- Inheritance
- OOP in Java
- Class Fundamentals
- Using Objects
- Constructor
- Garbage Collection
- Method Overloading
- Method Overriding
- Static Members
- Understanding Interface
- Using Interfaces

- **Packages**

- Why Packages
- Understanding Class Path
- Access Modifiers & Their Scope

- **Java Util Package**

- Date
- Enumeration
- Vector
- Properties
- Hashtable
- Collection & Iterator Interface
- Set Interface & Sortedset
- List and Arraylist

- **Exception Handling**

- Importance Of Exception Handling
- Exception Propagation
- Exception Types
- Using Try and Catch
- Throw, Throws, Finally
- Writing User Defined Exceptions

- **Inner Classes**

- Nested Top Level Classes
- Member Classes
- Local Classes
- Anonymous Classes

- **HTML**

- Introduction to Markup Language
- Formatting Tags
- Hyperlink
- Table
- Image
- Form and its Elements

- **Extensible Markup Language (XML)**

- Introduction to XML
- JAXP – Java API for XML Processing (DOM and SAX)
- Processing XML Documents

Course Content of Adv. Java Advanced Concepts of J2SE

- **I/O Operations in Java**

- Byte Oriented Streams
- File Handling
- Readers and Writers

- **Multithreaded Programming**

- Introduction to Multi-Threading
- Understanding Threads & its States
- Java Threading Model
- Thread class & Runnable Interface
- Thread Priorities
- Thread Synchronization
- Interthread Communication
- Preventing Deadlocks

- **Network Programming**

- Introduction to Networking
- InetAddress
- URL
- TCP Socket and Server Socket
- UDP Socket
- Developing a Chat Application

- **Abstract Window Toolkit**

- Graphics
- Color and Font
- AWT Components/Controls
- Event Handling & Layouts
- Frames and Menus
- Images and Animation
- Avoiding Flickering

- **Applet**

- Applet Basics
- Applet Life Cycle
- Applet Skeleton
- Applet Example

- **Swing Programming**

- Introduction to Swing and MVC Architecture
- Light Weight Component
- Swing Hierarchy
- Atomic Components e.g. JButton, JList and more
- Intermediate Container e.g. JPanel, JSplitPane and more
- Top-Level Container e.g. JFrame and JApplet
- Swing Related Events

Database Programming

- **Java Database Connectivity (JDBC)**

- Overview of RDBMS
- Introduction to Call Level Interface(CLI)
- Introduction to JDBC
- JCBC Architecture
- Types of JDBS Drivers
- Using Statement
- Using Prepared Statement
- Using Callable Statement
- Scrollable Result Set
- Editable Result Set
- Inserting & Fetching from BLOB type of Columns
- Managing Transactions in JDBC
- New Features Introduced in JDBC 3.0
- Fetching Generated Keys for Auto Increment Columns

Component Development Using JavaBeans

- **JavaBeans**
 - JavaBean Architecture
 - JavaBean Characteristics
 - Providing Properties & Methods
 - Event in JavaBeans

Distributed Application Development using Java EE Overview

Module 1 – Web Application Development

- **Common Gateway Interface**

- Introduction to CGI
- Understanding Environment Variables
- Disadvantages and Limitations of CGI

- **Java Servlets**

- Servlet as J2EE Web Component
- Servlet as an improved CGI
- Servlet Fundamentals / API
- What is a Web-Container
- Servlet Life Cycle / Architecture
- HTTP GET and POST Request Methods
- Processing Html Forms
- What is Name-Value pair
- Content Types and MIME
- Configuration of Web Application
 - ◆ Understanding the Deployment Descriptor (DD)
 - ◆ Writing the web.xml file
 - ◆ Specifying the Welcome file list
 - ◆ Servlet URL Pattern Mapping
 - ◆ Init Parameters
- Deploying Web Applications
- State Management
 - ◆ Using HTTP Session
 - ◆ Using Cookies
 - ◆ Using Application

- **Java Server Pages (JSP)**

- JSP Architecture
- JSP Standard/Implicit Objects
 - Request
 - Response
 - Out

- Config
- Application
- Session
- Page
- PageContext
- Exception
- JSP Page Implementation Class
- JSP Basics & Syntax
- JSP Directive Tags
 - ◆ Page Directive
 - Include Directive
 - Taglib Directive
- JSP Action Tags
 - Forward Action Tag
 - Include Action Tag
- JSP Script related Tags
 - ◆ Scriptlet Tags
 - ◆ Expression Tag
 - ◆ Declaration Tag
- Using Java Beans from JSP
 - ◆ UseBean Tag
 - ◆ Set Property Tag
 - ◆ Get Property Tag
- JSP Custom Tag Library
 - ◆ JSP 2.0 Tag Files
 - ◆ JSP 2.0 Simple Tag
 - Empty Tag
 - Tag with Body
 - (JSP Fragment)
- **JSP Expression Language (EL)**
 - ◆ Syntax
 - ◆ Operators
 - ◆ Using different scope objects
 - ◆ Calling Functions from EL
- **JSP Standard Tag Library (JSTL)**
 - ◆ General Purpose Actions
 - ◆ Conditional Actions
 - ◆ Iterator Actions

- **Web Application Security**

- ◆ Introduction
- ◆ Declarative Security
- ◆ Programmatic Security
- ◆ Roles
- ◆ Authentication
 - HTTP Basic Authentication
 - HTTP Digest Authentication
 - Form Based Authentication
 - Standard Login Form
- ◆ Security Constraints
 - Web Resource Collection
 - Authorization Constraint
 - User Data Constraint

- **Struts**

- Introduction To Struts and MVC Architecture
- Model-View-Controller (MVC) Architecture
- MVC Model 1 and Model 2 in Web Applications
- What is a Frame Work?
- Struts Frame work Components
 - ◆ Struts Controller Components
 - ◆ Struts Model Components
 - ◆ Struts View Components

Module 2

Distributed Application and Business Component Development.

Distributed Application Development

- **Remote Method Invocation (RMI)**

- Object Persistence and Serialization
- Introduction to Distributed Computing
- RMI Architecture
- Importance of RMI application
- Callback Implementation in RMI
-

- **COBRA**

- Introduction to COBRA
- COBRA for Distributed computing
- ORB & IIOP
- Interface Definition Language (IDL)
- Developing COBRA Client/Server application using JavaIDL

- **Java EE Application**

- Java EE Architecture
- Introduction to Java EE Components, Containers and Connectors
- Java EE Modules (Web App, EJB JAR, App Client)
- Structure of Java EE Application
- Packaging and Deploying Java EE Applications

- **JNDI**

- Introduction to Naming Services
- JNDI as Java API to Naming Services
- Using JNDI

- **JDBC Extension**

- Javax.sql package (Extension to JDBC)
- Data Source and Connection Pool
- Using JDBC and JNDI

- **Enterprise JavaBeans (EJB)**

- Introduction to Server-Side Components
- EJB Design Goals and Roles
- EJB Architecture
- Simplified EJB 3.0 API
- Metadata Annotations in place of XML
- RMI over IIOP
- Types of EJB
 - Session Bean
 - Message Driver Bean
 - Entity Bean
- EJB Container Services
 - Transactions
 - Security
 - Life Cycle Management
 - State and Persistence of EJB
- **Session Beans**
 - Role of Business Interface
 - Remote and Local Interfaces
 - Session Bean Lifetime
 - Developing Stateless Beans
 - Developing Stateful Beans
 - Standalone and Web Clients

- **Java Persistence API (JPA)**

- Designing Persistent Class
 - Entity Fields and Properties
 - Entity Instance Creation
 - Primary Keys and Entity Identity
 - Entity Relationships
- Entity Operations
 - Entity Manager
 - Entity Instance Life Cycle
 - Persistence Context
- Query API

- **Java Transaction Management (JTA)**

- The ACID Test for Transactions
- Introduction to JTS & JTA
- Container – Managed Transactions
- Bean – Managed Transactions
- Transaction Attributes
- Using JTA

- **Security Model**

- Role-Driver Access Control
- Security Identity
- Security and the Development Descriptor
- Using Security Roles
- Accessing Security Information via EJBContext
- Using JAAS to access secure EJB

- **Java Message Service (JMS)**

- Introduction to Messaging Systems
- Benefits of using JMS
- Pub / Sub Model
- Point-to-point Model
- Message Formats, Headers & Properties
- How JMS Fit into EJB system
- Developing Message Driven Beans (MDB)

- **Web Services**

- Introduction to Web Services
- Service Based Architecture
- Architecture and Advantages
- SOAP Significance
- WSDL

- Web Service Annotations
- Implementing a web Service
- Java API for XML based RPC
- Writing a web service Client
- **Java EE Design Patterns**
 - Introduction to Design Patterns
 - Java EE Design Patterns (Sun Blue Prints)
 - Identifying a pattern
 - Java EE Design Pattern Types
 - Presentation Tier Patterns
 - Intercepting Filter
 - Front Controller
 - View Helper
 - Composite View
 - Service to Worker
 - Dispatcher View
 - Business Tier Patterns
 - Business Delegate
 - Value Object
 - Session Façade
 - Composite Entity
 - Value Object Assembler
 - Value List handler
 - Service Locator
 - Integration Tier Patterns
 - Data Access Object
 - Service Activator