

Advanced Java Programming

Length: 4 days

Description:

This course presents several advanced topics of the Java programming language, including Servlets, Object Serialization and Enterprise JavaBeans. In order to build robust client server applications you need to understand the basics of many of these topics.

Course Objectives:

- Learn the basic concepts of Object-Orientation and how they are handled in Java
- Covers techniques for better class construction
- Understand Exceptions. How and when they should be handled
- An overview of database access and details for managing information using the JDBC API
- Examines the use of Object Serialization
- Addresses how to use Remote Method Invocation
- A general overview of Reflection and its uses
- Will be introduced to Java security
- Learn how to use Servlet and JSP and XML with JSP
- Be able to create and use custom JSP tags
- A presentation of Enterprise JavaBeans and how to use it

Audience:

The audience for this class should be Java programmers. This course contains the prerequisite material for developers interested in creating and maintaining client server systems written in Java.

Prerequisite:

Significant working knowledge of the Java language

Course Content:

Module 1 – About the Course

- Course Contents
- Overview
- Audience
- Course Objectives

Module 2 – Object Concepts in Java

- Objectives
- Object-Oriented Programming
- Objects
- Encapsulation
- Messages
- Classes
- Inheritance
- The Java Class Hierarchy

Module 3 – Creating Objects in Java

- Objectives
- Declaring Classes
- Deriving Classes
- Overriding Methods
- Overloading Methods
- Access Modifiers
- The default Access Modifier

- The public Access Modifier
- The private Access Modifier
- The static Modifier
- The final Modifier
- The synchronized Modifier
- The native Modifier
- Casting
- Object Creation
- The Constructor
- The New Operator

Module 4 – Abstracts, Interfaces and Packages

- Objectives
- Abstract Classes and Methods
- Class Visibility
- Inner Classes
- Interfaces
- Declaring Interfaces
- Implementing Interfaces
- Packages
- Declaring Packages
- Importing Packages

Module 5 – Exception Handling

- What is an Exception?
- Some Terminology
- Throw and Exception
- Throw, try and catch Blocks
- An Exceptional Example
- Multiple Catch Blocks
- Does Every Method Have to Catch Every Exception?
- The finally Clause
- The Throwable Class
- Types of Exceptions
- java.lang Exceptions
- Built-In Exceptions

Module 6 – Database Access with JDBC

- Remote Database Access
- ODBC and JDBC Drivers
- Microsoft's ODBC
- Enter JDBC
- Other Drivers
- Design Purpose
- Overview of Main Classes
- Database Names
- Overview of a Database Connection
- The JDBC Classes
- Other Functionality
- Extending the JDBC
- Designing a Database Application

Module 7 – Object Serialization

- Storing Objects to Streams
- The serializable Interface
- Using Sockets to Send a Serialized Object
- Special Requirements for Using Serializable
- The seralver Tool
- The Externalizable Interface
- Object Serialization and RMI

Module 8 – RMI

- RMI and Distributed Applications
- RMI Terminology
- The java.rmi Packages
- The java.rmi Package
- The java.rmi.registry Package
- The java.rmi.server Package
- The java.rmi.activation Package
- The java.rmi.dgc Package
- Implementing RMI
- Create the Remote Interface
- Create a Class that Implements the Remote Interface
- Create Stub and Skeleton Classes
- Start Up the Remote Registry
- Create and Register the Remote Object
- Implementing the RMI Client on the Local Host

Module 9 - Reflection

- Examining Classes
- Retrieving Class Objects
- Getting the Class Name
- Discovering Class Modifiers
- Finding Superclasses
- Identifying the Interfaces Implemented by a Class
- Examining Interfaces
- Identifying Class Fields
- Discovering Class Constructors
- Obtaining Method Information
- Creating Objects
- Using No-Argument Constructors
- Using Constructors that Have Arguments
- Getting Field Values
- Setting Field Values
- Invoking Methods
- Identifying Arrays
- Retrieving Component Types
- Creating Arrays
- Getting and Setting Element Values

Module 10 – The Java Security Model

- Configurable Security Policy
- Policy Permissions
- Extending the Sandbox
- Application Security

- Cryptographic Support
- Using Certificates

Module 11 – Applet Security

- Using JAR Files and Digital Signatures
- Using the JAR Tool
- Creating a JAR File
- Listing the Contents of a JAR File
- Viewing a JAR File
- Extracting the Contents of a JAR File
- Signing Applets
- Creating a Keystore
- Signing a JAR File
- Verifying the Signature of a JAR File
- Working with Certificates
- Exporting Your Certificates
- Importing the Certificates of Others
- Other keytool Commands
- The java.security Packages

Module 12 – Servlets and JSP

- What is the Advantage of Servlets over “Traditional” CGI?
- What is JSP?
- Obtain and Install the Servlet and JSP Development Kits
- Basic Servlet Structure
- A Simple Servlet Generating Plain Text
- A Servlet that Generates HTML
- Simple HTML-Building Utilities
- Form Data
- Example: Reading Three Parameters
- An Overview of Request Headers
- Reading Request Headers from Servlets
- Introduction to CGI Variables
- HTTP Status Codes
- Specifying Status Codes
- HTTP 1.1 Status Codes and Their Meaning
- Example: Search Engine Front End
- HTTP Response Headers
- Common Response Headers and their Meaning
- Example: Automatically Reloading Pages ad Content Changes
- Overview of Cookies
- The Servlet Cookie API
- Creating Cookies
- Reading and Specifying Cookie Attributes
- Placing Cookies in the Response Headers
- Reading Cookies from the Client
- Some Minor Cookie Utilities
- Getting the Value of a Cookie with a Specified Name
- Example: A Customized Search Engine Interface
- What is Session Tracking?
- The Session Tracking API
- Looking up the HttpSession Object associated with the current request
- Looking up Information Associated with a Session

- Associating Information with a Session
- Example: Showing Session Information
- JSP Overview
- Template Text: Static HTML
- JSP Scripting Elements
- JSP Expressions
- JSP Scriptlets
- JSP Declarations
- JSP Directives
- The JSP page Directive
- The JSP include Directive
- Example Using Scripting Elements and Directives
- Predefined Variables
- Actions
- More jsp:useBean Details
- The jsp:getProperty Action

Module 13 – JSP and XML

- Using XML Data Sources in JSP Pages
- Generating Markup Languages Using JSP
- Conclusion

Module 14 – What a Custom Tag Library Is

- The Composition of a Custom Tag Library
- The JSP

Module 15 – Enterprise JavaBeans

- Benefits of Middle-Tier Servers
- J2EE Architecture
- J2EE Server
- EJB Container
- Web Container
- Enterprise Beans
- Session Beans
- Entity Beans
- Comparing Session and Entity Beans
- JavaBean Components and Enterprise Beans
- Programming Restrictions for Enterprise Beans
- Database Access
- J2EE Applications
- Contents of a J2EE Application
- Development Phases of J2ee Applications
- Getting Started
- Creating the J2EE Application
- Building the Enterprise Bean
- Write the Remote Interface Code
- Write the Home Interface Code
- Write the Enterprise Bean Code
- Compile the Source Code Needed by the Enterprise Bean
- Create the EJB .jar File
- Deploying the J2EE Application
- Building the Client
- Write the Client Code

- Compile the Client Code
- Running the Client
- Solutions to Common Problems
- Servlets
- Use Servlets instead of CGI Scripts!
- Other Uses for Servlets
- Architecture of the Servlet Package
- A Simple Servlet
- Setting Up the Servlet's J2EE Application
- Coding the Servlet
- Compiling the Servlet
- Creating the Servlet's .war File
- Deploying the Servlet's J2EE Application
- Running the Servlet
- JavaServer Pages Components
- Other dynamic Content models
- Setting Up the JSP Component's J2EE Application
- Writing the .jsp File
- Coding the JavaBeans Component
- Creating the JSP Component's .war File
- Deploying the JSP Component's J2EE Application
- Running the JSP Component