

Intensive and hands-on, this three day course emphasizes becoming productive quickly in the Java language. This course quickly covers the Java 5.0 language syntax and then moves into more advanced features of the language such as abstract classes, interfaces, generics, packages, and exception handling. Students will then learn two foundational API libraries: I/O streams and collections. Additional appendices on threads, J2EE, and Eclipse are provided for further study.

Course Objectives:

- Write stand-alone applications using the Java language.
- Accurately implement Object-Oriented concepts using Java features, such as classes, interfaces, and references.
- Create well-scoped classes using packages and inner classes.
- Write programs which both handle and create exceptions.
- Read and write data, using input and output streams.
- Use the Java 2 Collections Framework to work with groups of objects.

Audience: Programmers moving to the Java language.

Prerequisites: Professional programming experience in C, C++, or C# required. Knowledge of Object-Oriented concepts is required.

Number of Days: 3 days

- | | |
|---|---|
| <p>1. Course Introduction
 Course Objectives
 Overview
 Suggested References</p> <p>2. Getting Started with J2SE
 What is Java?
 How to Get Java
 A First Java Program
 Compiling and Interpreting Applications
 The JDK Directory Structure</p> <p>3. Language Fundamentals
 A Java Program
 If Statements
 Switch Statements
 Loop Statements
 Syntax Details
 Primitive Datatypes
 Variables
 Expressions in Java
 Strings
 Arrays
 Enhanced for Loop</p> | <p>4. Objects and Classes
 Defining a Class
 Creating an Object
 Instance Data and Class Data
 Methods
 Constructors
 Access Modifiers
 Encapsulation</p> <p>5. Using Java Objects
 Printing to the Console
 printf Format Strings
 StringBuilder and StringBuffer
 Methods and Messages
 toString
 Parameter Passing
 Comparing and Identifying Objects
 Destroying Objects
 Using the Primitive-Type Wrapper Classes
 Autoboxing</p> <p>6. Inheritance in Java</p> |
|---|---|

- Inheritance
- Inheritance in Java
- Casting
- Method Overriding
- Polymorphism
- super
- The Object Class
- 7. Advanced Inheritance and Language Constructs**
- Enumerated Types - Pre-Java 5.0
- Enumerated Types Today
- More Enumerated Types
- Abstract Classes
- Interfaces
- Using Interfaces
- Comparable
- Collections
- Generics
- 8. Packages**
- Packages
- The import Statement
- Static Imports
- CLASSPATH and Import
- Defining Packages
- Package Scope
- 9. Exception Handling**
- Exceptions Overview
- Catching Exceptions
- The finally Block
- Exception Methods
- Declaring Exceptions
- Defining and Throwing Exceptions
- Errors and RuntimeExceptions
- Assertions
- 10. Input/Output Streams**
- Overview of Streams
- Bytes vs. Characters
- Converting Byte Streams to Character Streams
- File Object
- Binary Input and Output
- PrintWriter Class
- Reading and Writing Objects
- Basic and Filtered Streams
- 11. Core Collection Classes**
- The Collections Framework
- The Set Interface
- Set Implementation Classes
- The List Interface
- List Implementation Classes
- The Queue Interface
- Queue Implementation Classes
- The Map Interface
- Map Implementation Classes
- 12. Appendix A – Introduction to Threads**
- Non-Threaded Applications
- Threaded Applications
- Creating Threads
- Thread States
- Runnable Threads
- Coordinating Threads
- Interrupting Threads
- Runnable Interface
- ThreadGroups
- 13. Appendix B – J2EE Overview**
- Introduction to J2EE
- J2SE Building Blocks
- Servlets, JSPs, and Web Applications
- Web Services
- Enterprise JavaBeans
- Additional J2EE APIs
- J2EE Clients
- The J2EE Platform
- 14. Appendix C – Eclipse**
- Introduction to Eclipse
- Installing Eclipse
- Running Eclipse for the First Time
- Editors, Views, and Perspectives
- Setting up a Project
- Creating a New Java Application
- Running a Java Application
- Debugging a Java Application
- Shortcut Key Sequences
- More Shortcut Key Sequences
- Setting the Classpath
- Importing Existing Java Code into Eclipse