

Java for the Cobol Programmer

Course Summary

Description

This course will introduce the student to the Java programming language with the emphasis on the COBOL language. The student will learn how to develop applications for an intranet as well as applets for the web. Through hands-on exercises on any platform, the student will become familiar with the Java syntax and the object oriented approach that this language utilizes.

Objectives

At the completion of this course, the student will be able to:

- Understand the Java language syntax
- Understand object orientation and its utilization in Java
- Create, compile and run Java applications
- Develop web pages with embedded Java applets
- Build a GUI interface
- Create multithreaded applications
- JAVA versus COBOL

Topics

- Introduction to Java
- Java syntax (data types, operators, control statements)
- Introduction to object-oriented programming in Java
- Java and the Web (building applets)
- Arrays and vectors
- Creating a graphical user interface in Java
- Multithreading in Java
- Navigating Sun's API documentation
- Running applications on OS/390 Enterprise Server
- Introduction to WebSphere

Audience

This course is intended for programmers who will be using Java as an application-development language or who want to use Java in conjunction with the creation of a web page or web site.

Prerequisites

The student must be familiar with the Internet and must have experience programming in at least one mid to high level language (COBOL, Visual Basic, C, C++, etc.).

Duration

Five days

Java for the Cobol Programmer

Course Outline

I. Java and OO Programming

- A. Object-Oriented Fundamentals
- B. Objects, Attributes, Methods
- C. Classes and Instances
- D. UML Notation
- E. Association
- F. Inheritance
- G. Aggregation
- H. Encapsulation
- I. Polymorphism

II. The Java Environment

- A. Portability and Compilation
- B. Class Programs
- C. Code Blocks
- D. Comments
- E. Identifiers
- F. Naming Rules
- G. Naming Conventions

III. Variables

- A. Primitive Variables
- B. Reference Variables
- C. Data Types
- D. Declaring and Initializing
- E. Constants
- F. Variable Scope
- G. Variable Accessibility

IV. Methods

- A. Object interaction
- B. Method Structure
- C. Method Signature
- D. Method Accessibility
- E. Standard Methods
- F. Custom Methods
- G. Constructors
- H. Accessor Methods
- I. Mutator Methods

V. Classes and Instances

- A. Static Variables
- B. Instance Variables
- C. Static Methods
- D. Instance Methods
- E. The String Class

VI. Computation

- A. Assignment

- B. Simple Math
- C. Shortcut Code
- D. Type Casting
- E. The Math Class
- F. Rounding
- G. Wrapper Classes
- H. Integer Class
- I. NumberFormat Class

VII. Decisions

- A. Java's Logical Operators
- B. if Logic
- C. if...else
- D. if Blocks
- E. Nested if
- F. Compound Expressions
- G. Conditional Operator
- H. switch Statement

VIII. Loops

- A. Pre-test vs. Post-test Loop
- B. while Loop
- C. do Loop
- D. for Loop

IX. Arrays

- A. COBOL Tables
- B. Subscript vs. Index
- C. Declaring Java arrays
- D. Populating Java arrays
- E. Accessing Java arrays
- F. Two-dimensional arrays

X. GUIs

- A. Applets and Applications
- B. Java GUI Packages
- C. AWT Classes
- D. Event-driven Programming
- E. Source and Listener
- F. Event Class Instances
- G. Listening for Events
- H. Frames and Windows
- I. Defining an Application Window
 - 1. Imports
 - 2. Class Header
 - 3. main Method
 - 4. Constructor
 - 5. Event Handler•Application Components
- J. Application with a Button

XI. Reference Variables

- A. Associations Between Classes
- B. Reference Variable Pointers
- C. Creating Your Own Classes
- D. Declaring Reference Variables
- E. Creating Instances Of Classes