



Java Fundamentals

About This Course

Java is more than a programming language. Java is a platform that includes an extensive library to simplify the creation of applications and applets. Java is designed specifically for to run on a network, including, of course, the Internet. This has made Java a favored solution for the web-based applications and e-commerce.

Java offers the advantages of object-oriented programming, a robust run-time that creates a secure platform, the ability to run on a variety of hardware and operating system platforms, and an extensive library of packages.

This course is designed to enable computer programmers to take advantage of the Java platform. In addition to covering the elements of the language, the course covers the object-oriented concepts that underlie the language and the network-oriented run-time.

Course Duration

Three days.

Who This Course Is For

This course is designed for the programmer who would like to broaden his/her skill-set into any (or all) of the following:

- the Java platform
- client-side web-based applications
- the object-oriented programming paradigm

Prerequisites

Participants should already be established programmers who are comfortable with a high-level programming language (such as C++, C, Pascal, Fortran, Cobol, Python, or Perl). Note: HTML is a mark-up language and does not qualify as a programming language.

What Participants Will Learn

The basic elements of the language and the corresponding syntax are covered first, along with the basic tools used to create and execute a Java program. Then the object-oriented paradigm on which Java is based is developed. The course then explores the standard libraries (i.e., packages) that provide Java with a graphical user interface and which can be used for applications and applets.



After completing the course participants will be able to:

- Organize Java programs from a development perspective
- Effectively use the basic tool-set to develop Java programs
- Develop Java classes, applets, and applications
- Present a graphical user interface elements in applications and applets
- Use layout managers to organize the graphical user interface
- Write programs driven by graphical user interface events
- Incorporate applets into web pages
- Utilize the packages that form the Java API
- Exploit the advantages of the object-oriented programming paradigm

Course Outline

- Day 1
 1. Introduction
 - (a) History of Java
 - (b) Advantages Java Offers
 - (c) Online Java Resources
 - (d) First Example
 - (e) The Standard Java Tool-Set
- 2. A Few Basics
 - (a) File Extensions
 - (b) Identifiers
 - (c) Comments
 - (d) Java Applications
 - (e) Java Applets
- 3. Types and Operators
 - (a) Numeric Types, Literals, and Operators
 - (b) Accessing Mathematical Functions
 - (c) Boolean Type, Literals, and Operators
 - (d) Character Type, Literals, and Operators
 - (e) String Type, Literals, and Operators
 - (f) Operator Precedence
- 4. Flow of Control
 - (a) The if/else Statement
 - (b) The switch Statement
 - (c) The Conditional Operator
 - (d) The for Loop
 - (e) The while Loop
 - (f) The do-while Loop
 - (g) The break and continue Statements



-
- (h) Nesting and Naming Loops
 - 5. Arrays
 - (a) Declaring Arrays
 - (b) Allocating Arrays
 - (c) Initializing Arrays
 - (d) Arrays of Multiple Dimensions
 - (e) Array Assignment, Copying, and Cloning
 - Day 2
 - 1. Object-Oriented Programming
 - (a) The Structure of a Class
 - (b) The Instance
 - (c) Variables and Methods
 - (d) Scoping Issues
 - (e) Default Values of Variables
 - (f) The Uses of this
 - (g) Constructors
 - (h) Class Self-Reference
 - (i) Relationships between Classes
 - 2. Inheritance
 - (a) The Role of Inheritance
 - (b) Vocabulary of Inheritance
 - (c) Single and Multiple Inheritance
-
- (d) Class Composition v. Class Inheritance
 - (e) The Uses of super
 - 3. Packages and Access Attributes
 - (a) The Package Hierarchy
 - (b) Standard Packages
 - (c) Using Packages
 - (d) Creating Packages
 - (e) Access Attributes
 - 4. Exploring Applets
 - (a) The Applet Class
 - (b) Drawing in an Applet
 - (c) Referencing Applets from HTML
 - (d) Passing Data to Applets
 - Day 3
 - 1. Graphical User Interface Components
 - (a) Components and Containers
 - (b) Declaring, Creating, and Placing Components
 - (c) Buttons
 - (d) Labels
 - (e) TextFields and TextAreas
 - (f) Checkboxes and CheckboxGroups
-



2. Layout Managers
 - (a) The Power of Layout Managers
 - (b) Using a Layout Manager
 - (c) Nesting Containers
 - (d) FlowLayout
 - (e) GridLayout
 - (f) BorderLayout
 - (g) GridBagLayout
3. Graphical User Interface Events
 - (a) Events
 - (b) Listeners
 - (c) Writing a Listener
 - (d) Using a Listener
 - (e) High- and Low-Level Events
 - (f) Event Class Hierarchy
 - (g) Listener Class Hierarchy
4. Graphics Based Applications
 - (a) Comparing Applications and Applets
 - (b) The Frame
 - (c) Dialog Boxes

- (d) MenuBars, Menus, and MenuItems
- (e) Hierarchical Menus

Hands-On Workshops

Roughly 50% of the time is spent with hands-on workshops. Numerous Java applets and applications are written and improved throughout the course. At the course's conclusion participants will have written both applets and applications offering users a graphical user interface, responding to multiple types of events, and using the object-oriented programming paradigm.

Materials Provided

Each participant receives:

- A comprehensive course textbook/workbook
- Solutions to all exercises
- A CD-ROM containing examples, exercise solutions, and other items as allowed by license
- A certificate of completion

© 2002 J. Eric Ivancich, all rights reserved.