J2SE 1.4: Java Fundamentals

Meet the expert: Noah Kriegel is a software engineer and instructor with The DSW Group, Ltd. in Atlanta, Ga., specializing in the development of custom Web-based enterprise applications. During his career in the software industry, he has also developed many database and other distributed applications for clients of all sizes and in a variety of industries. Over the years, he has also instructed hundreds of developers with the languages that he has used in creating applications. His primary language proficiencies include Java, C#, C, and Pascal/Delphi.

Terry Dietzler is president and co-founder of The DSW Group, Ltd., a nationally recognized application development and training firm based in Atlanta, Georgia. Terry has been a Borland Certified Delphi Trainer since 1996 and is co-author of the book >JBuilder3 Unleashed>. Since the inception of DSW in 1987, Terry has managed and developed on many software projects for a wide range of companies. During that time, he has become an accomplished developer and trainer familiar with several development products & languages.

Runtime: 10:02:20

Course description: In this course, you will learn the fundamentals of the Java language. It begins by showing you how to obtain, setup, and configure the Java Developers Kit, including all the provided tools. You will learn the basic syntax of Java, which includes the language syntactic elements as well as the object-oriented features of Java. You will learn about exception handling, database connectivity, and deployment. As the course progresses, you will build on your knowledge of Java fundamentals as you learn about the advanced object-oriented features in Java. You will learn about threading, building graphical and event-driven applications using the Swing framework, advanced JDBC, files and streams, and applet development. You will also learn the basic first steps for web development in Java.

Course outline:

Introduction to Java
- Introduction
- What is Java?
- What is NOT Java
- Common Misconceptions
- Java Versions
- Advantages of Java
- Platform Design Goals
- Language Design Goals
- Java"s Beginning
- History of JSE
- Summary

Installation / Setup
- Introduction
- JSE Acronyms
- Obtain JDK / Install JDK
- Demo: Install the JDK
- View the Installed Files
- JDK Configuration
- Demo: PATH
- Demo: rt.jar
- Demo: VERSION
- Compile / Run Java Programs
- Tools: javac
- Demo: javac
- Tools: java / javaw
- Demo: java / javaw
- Tools: jdb / jar / javadoc
- Demo: javadoc
- Explore the Documentation
- Summary

Syntax
- Introduction
- Intro to jEdit Editor
- Demo: Create a Simple App
- Demo: Compile / Test App
- Data Types: Primitives
- Data Types: References
- Type: Integers
- Type: Floating Point Numbers
- Type: Character / Boolean
- Naming Conventions
- Variables
- Operators
- Arithmetic Operators
- Increment/Decrement
- Compound Operators
- Relational Operators
- Bitwise Operators
- Other Operators
- Conversions: Implicit/Explicit
- Summary

Flow Control
- Introduction
- Variable Scope
- Looping
- Demo: While Loop
- Demo: Do...While Loop
- Demo: For Loop
- Decisions
- Demo: if ...else
- Demo: switch
- Demo: Ternary Operator
- Summary

Strings / Arrays
- Introduction
- Using Strings
- Creating Strings
- Concatenating Strings
- Strings are Immutable
- StringBuilder class
- String Behavior
- equals Method
- Sub-Strings
- Getting Arrays
- Using Arrays
- Declare/Dimension Arrays
- Referencing Arrays
- Working with Arrays
- Summary

Introduction to OOP
- Introduction
- Object-Oriented Primer
- Classes vs. Objects
- Explore Existing Class
- Create Objects w/Class
- Create a New Class
- Create a Class To Test
- Compile the Class
- Modify the Constructor
- Create Methods
- Compile/Test Class
- Add Additional Methods
- Compile/Test Modified Class
- Summary

Intermediate OOP
- Introduction
- Add Class Functionality

(Continued on page 2)

J2SE 1.4: Java Fundamentals

- Test Functionality
- Create Overloaded Method
- Test Overloaded Method
- Add Static Methods
- Test Static Methods
- Overload the Constructor
- Test the New Constructor
- Use Multiple Constructors
- Create Initialization Block
- Summary

Intermediate OOP (cont.)
- Introduction
- Overview: Packages
- Import Packages
- Packages and CLASSPATH
- Compile with CLASSPATH
- Attributes/Encapsulation
- Specify Access Attributes
- Packages and Access Levels
- Accessor / Mutator Methods
- Packages
- Use a Package
- Overview: Inheritance
- extends Keyword
- Subclass Constructors
- Override Subclass Methods
- Demo: Inheritance
- Overview: Polymorphism
- Overview: Object Superclass
- instanceof Keyword
- getClass() Method
- Cast an Object
- Demo: Polymorphism
- Summary

Exception Handling
- Introduction
- Overview: Exception Handling
- Syntax: Exception Handling
- Demo: Simple try...catch
- Generic Exception
- Handle Multi Exception Types
- Exception Propagation
- Re-throwing Exceptions
- Your Own Exceptions
- Exception Hierarchy
- Throws Clause
- Demo: throws Clause
- Finally Block
- Demo: finally Block
- Summary

Intro to JDBC
- Introduction
- Architecture of JDBC
- JDBC Overview
- Driver Types
- JDBC Drivers
- Obtaining a Driver
- Setup Sample Database
- View Database Example
- JDBC Classes
- More Complex DB Example
- Examine mysql.jdbc driver
- Get a Connection
- Create the ResultSet
- Create the Output
- View the Output
- Summary

Deployment
- Introduction
- Java Client Environment
- CLASSPATH Settings
- Deploy Command Line Apps
- Deploy GUI Apps
- CLASSPATH Variable
- CLASSPATH Default Setting
- Perm. CLASSPATH Setting
- JAR Utility
- JAR Command Line Tool
- JAR Utility Options
- Creating a JAR File
- View JAR File Structure
- Run App w/Cmd Line Switch
- Using the Extensions Directory
- Self Executing JAR File
- View the Manifest File
- Summary

Advanced Syntax
- Introduction
- Immutable Strings
- StringBuilder
- String Methods
- String Techniques
- String Demo / StringBuilder
- Demo: String Tokenizer
- Re-Dimensioning Arrays
- Multi-Dimensional Arrays
- Demo: Ragged Arrays
- Parameter Passing Semantics
- Summary

Advanced OOP
- Introduction
- Review: final
- Abstract Class
- Interfaces
- Inner Classes
- Create an Inner Class
- Anonymous Classes
- Anonymous Class Syntax
- Cloning
- Shallow Copy vs Deep Copy
- Create a Shallow Copy
- Extend Shallow Copy
- Create a Deep Copy
- equals / hashCode Methods
- Summary

Threading
- Introduction
- Overview: Threading
- Define Threads
- Subclassing / Runnable
- Start a Thread
- Thread Types
- Stop a Thread
- Demo: Implement Threads
- Demo: Runnable Interface
- Demo: Use Threads in an App
- Thread Execution / Priorities
- Manage Threads
- Synchronization
- Methods / Code Blocks
- Asynchronous Thread Access
- Use wait to Control Execution
- Use synchronized keyword
- wait() Method
- notify() and notifyAll() Methods
- Demo: Use notifyAll() Method
- Summary

Introduction to Frames
- Introduction
- Frame Basics
- Abstract Windowing Toolkit
- Frames in Swing
- JFrame
- Create a Frame
- Close a Frame
- Frame Title / Location
- Demo: Add Title / Set Location
- Containership
- Demo: Add Button to Frame
- Layout Managers
- BorderLayout
- Demo: BorderLayout Manager
- FlowLayout
- Demo: FlowLayout Manager
- GridLayout
- Demo: GridLayout Manager
- BoxLayout
- Demo: BoxLayout Manager
- Summary

Advanced Frames
- Introduction
- GridBagLayout
- GridBag: gridx / gridy
- GridBag: gridwidth / gridheight
- GridBag: weightx / weighty
- GridBag: anchor and fill
- GridBag: insets
- GridBag: ipadx / ipady
- GridBagLayout Manager
- Create Multi-Window Apps
- Parent Frame
- Dialogs
- Dialogs: Modality
- Disposing Frames / Dialogs
- Multi-Frame Application
- InputNameDialog code
- Summary

Introduction to Swing
- Introduction
- Swing vs AWT Controls
- Overview: SwingSet
- JLabel/JTextField/JTextArea
- Demo: JLabel
- Demo: JTextField
- Demo: JTextArea
- JButton/JProgressBar/Jslider
- Demo: JButton
- Demo: JProgressBar
- Demo: JSlider
- JRadioButton/ButtonGroup
- JPanel / JScrollPanel
- Demo: JRadioButton
- Demo: JPanel
- Demo: JScrollPanel
- JTabbedPane / JSplitPane
- JOptionPane
- Demo: JTabbedPane
- Demo: JSplitPane
- Demo: JOptionPane
- Summary

Event Handling
- Introduction
- Event Delegation Model
- Adapters
- External Classes
- Demo: Simple Event Handler
- External Classes with Params

(Continued on page 3)

J2SE 1.4: Java Fundamentals

- Event Handler w/ Params
- Frame and Inner Classes
- Anonymous Inner Classes
- Demo: Frame Class
- Demo: Inner Class
- Demo: Anonymous Inner Class
- Summary

Update Swing Components
- Introduction
- Model-View-Controller Model
- Creating Models
- ListModel Class
- Demo: Create a JList
- Update List Models
- Create an Updatable JList
- Overview: JTable
- AbstractTableModel Class
- Create a Jtable
- More Complex Jtable
- Summary

Advanced JDBC
- Introduction
- Use ResultSetMetaData
- Data Aware Table Model
- Efficiency Concerns
- PreparedStatements
- Handling Keys
- Demo: Handling Keys
- Summary

Files and Streams
- Introduction
- Overview: Files and Streams
- File and Stream Hierarchy
- Byte Streams
- FileInput/Output Stream
- BufferedInput/Output Streams
- Text Streams
- Read / Write Text Files
- Work with ZIP/JAR Files
- Demo: Create a ZIP File
- Demo: Read a Zip File
- Summary

Utility Classes
- Introduction
- Type Wrappers
- Collections API
- List Interface
- Vector Class
- ArrayList Class
- Map Interface
- Hashtable
- Set Interface
- Demo: Collections API
- Calendars and Dates
- Demo: Use the Calendar
- Math Class
- NumberFormat
- Demo: NumberFormat Class
- Summary

Generics and Enumerations
- Introduction
- Use Collections
- Generics
- Type Erasure
- Generic Subtypes
- When to Use Generics
- Using Non Generic Code
- Generics Class
- Rewrite Using Generics
- Enumerations
- Java enum Type
- Use int EnumType
- Rewrite u/new EnumTypes
- Enhance enum Declaration
- Summary

Applets
- Introduction
- Overview: Applets
- Applet Restrictions
- Build Applets
- Applet Methods
- Test Applets
- Deploy Applets
- Build / Test / Deploy Applet
- Summary

Intro to Web Development
- Introduction
- Overview
- Web Basics
- Static / Dynamic Content
- Using Servlets
- Servlet Lifecycle
- Servlet Containers
- Building Servlets
- Requests and Responses
- Create a Simple Servlet
- Deploy the Servlet
- Test the Servlet
- Statelessness
- Session Management
- Using Session State
- Test the Session App
- Java Server Pages (JSP)
- JSP Architecture
- Code in JSP
- Scriplets / Expressions
- Deploy JSP Page
- Test the JSP Page
- Summary