Java™ Programming Language Workshop
SL-285

The *Java™ Programming Language Workshop* course provides students with practical experience in designing a vertical solution for a distributed, multi-tier application. Students use graphical user interface (GUI) design principles and network-communications capabilities to code a functional Java™ application that interacts with a networked database server. The significant amount of lab time illustrates the workshop nature of this course.

Who Can Benefit

Students who can benefit from this course are programmers already familiar with the basic structure and syntax of the Java programming language who need to further expand this knowledge to develop complex, production-level applications.

Completing this course also assists individuals preparing for the Sun Certified Developer for the Java™ Platform examination.

Prerequisites

To succeed fully in this course, students should be able to:

- Develop applications using the Java programming language
- Understand basic Unified Modeling Language (UML) diagrams
- Understand basic System Query Language (SQL) statements
- Understand how to implement interfaces and handle Java programming exceptions
- Use object-oriented programming techniques
- Understand GUI design
- Understand basic Transmission Control Protocol/Internet Protocol (TCP/IP) communication
- Program with sockets or streams

Skills Gained

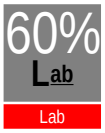
Upon completion of this course, students should be able to:

- Implement and test a "from-the-ground-up" program that could be used in a commercial intranet application
- Develop classes to connect programs to Structured Query Language (SQL) database systems using the core aspects of the Java™ Database Connectivity [JDBC™] application programming interface (API)
- Apply design patterns to create reusable classes
- Organize and set up the infrastructure to support a Java technology project
- Create two- and three-tier Java technology applications
- Create remote objects using Java™ Remote Method Invocation (Java RMI) Create a multi-threaded server

Related Courses

Before:

- SL-275: *Java™ Programming Language*



November 2001
SL285_revC_CD_1101.fm

- SL-265: *Java™ Technology for Structured Programmers*
- OO-226: *Object-Oriented Application Analysis and Design for Java™ Technology (UML)*
- *SL 255: Java™ Technology for Visual Basic Programmers*

After:

- SL-301: *Distributed Programming With Java™ Technology*

Course Outline

Module 1 – *Introducing the BrokerTool Project*

- Explain the problem statement of the BrokerTool Project

Module 2 – *The Model View Controller (MVC) Design Pattern*

- Explain design patterns
- Explain the Model View Controller (MVC) design pattern
- Analyze how the MVC design pattern can be used in applications

Module 3 – *The BrokerTool Design*

- Begin the analysis and design of the BrokerTool project
- Apply the MVC design pattern
- Develop a build plan for the project

Module 4 – *Introduction to Java Database Connectivity (JDBC)*

- Describe Java Database Connectivity (JDBC)
- Explain how using the abstraction layer provided by JDBC can make a database front end portable across platforms
- Describe the five major tasks involved with the JDBC programmer's interface
- State the requirements of a JDBC driver and its relationship to the JDBC driver manager

Module 5 – *Creating Client Software*

- Implement a view class
- Implement a controller class

Module 6 – *Multiple-Tier Design*

- Compare the BrokerTool two-tier design with the BrokerTool three-tier design
- Explain how you can use the Java technology package `java.net` to implement networking applications
- Show how you can use the Command design pattern in the BrokerTool application
- Apply the Strategy design pattern to create reusable code
- Describe how you can implement the BrokerTool network client
- Describe how you can implement the BrokerTool network server

Module 7 – *Advanced Multiple-Tier Design*

- Create a multi-threaded server
- Examine a thread pool
- Identify integrity problems in multi-threaded servers

Module 8 – *Remote Objects*

- Create remote objects
- Use Java Remote Method Invocation (Java RMI) to create a multi-tier application

Module 9 – *Building GUIs*

- Apply the principles of good GUI design
- Differentiate, at a high level, between the new Java™ Foundation Classes (Swing components) and the Abstract Window Toolkit (AWT) model
- Explain how you can create the class structure needed for an object-oriented GUI

- Design and implement a GUI for the BrokerTool project using your choice of containers, components, and layout managers

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303

Telephone: 800 422-8020