

Java Programming (Java SE 6)

Course Specifications

Course number: 085662

Course length: 5.0 day(s)

Course Description

Course Objective: You will work with the advanced features of Java.

Target Student: This course is intended for existing Java SE 6 programmers who have worked with databases and with object-oriented programming techniques, who are now ready to create more complex and advanced programs using Java SE 6. It will also benefit students who are preparing for the Oracle Certified Professional, Java SE 6 Programmer examination.

Prerequisites: Students should be skilled and experienced in using Java SE 6. They should have knowledge of database structure and terminology. Also, the students should understand object-oriented programming. To ensure your success, we recommend you first take the Element K courses: Introduction to Programming using Java and Object-Oriented Programming and Design Using Java (Third Edition) or have equivalent knowledge.

Hardware Requirements

On each student's machine, you need the following hardware:

- Intel® Pentium® 1 GHz or higher processor.

- 512 megabytes (MB) of RAM or more.

- 10 gigabytes (GB) of available hard-disk space or more.

- CD-ROM drive or DVD-ROM drive.

- Sound card.

- Headphones or speakers to hear sound effects.

- Color monitor with a 16-bit or greater video card.

- Monitor with 1024 x 768 resolution.

- A mouse or other pointing device.

- Internet connection.

- Display system to project the instructor's computer screen.

Software Requirements

Each computer requires the following software:

- Java Development Kit (JDK) 6

- Eclipse Java EE IDE for Web Developers

Course Objectives

Upon successful completion of this course, students will be able to:

- create and manage custom classes.

- control program flow by writing code to respond to specific criteria.

implement object-oriented programming techniques to create reusable and reliable programs.

work with Java utility class libraries.

use the capabilities of the Java I/O package to read and write data to external devices.

use the collection APIs in Java to manage data.

use generics to enforce compile-time type checking.

use multi-threaded programs that can handle multiple tasks concurrently.

manage Java applications.

Course Content

Lesson 1: Working with Classes

Topic 1A: Create Classes

Topic 1B: Create Variables

Topic 1C: Write an Expression

Topic 1D: Work with Arrays

Topic 1E: Work with Static Class Members

Topic 1F: Define Methods

Topic 1G: Use Enumerated Data Types

Lesson 2: Controlling Program Flow

Topic 2A: Work with Conditional Statements

Topic 2B: Work with Looping Statements

Topic 2C: Handle Exceptions

Topic 2D: Handle Chained Exceptions

Topic 2E: Write and Enable Assertions

Lesson 3: Implementing Object-Oriented Programming Concepts

Topic 3A: Extend a Class

Topic 3B: Overload and Override Methods

Topic 3C: Work with Interfaces

Topic 3D: Create Inner Classes

Topic 3E: Work with Object-Oriented Design Concepts

Lesson 4: Working with Java Utility Class Libraries

Topic 4A: Work with Strings

Topic 4B: Format and Parse Strings

Topic 4C: Work with Dates, Numbers, and Currencies

Lesson 5: Using the Java I/O Package

Topic 5A: Work with the File Class

Topic 5B: Work with Byte Streams

Topic 5C: Work with Character Streams

Topic 5D: Read Files

Topic 5E: Write to a File

Topic 5F: Manipulate I/O Objects

Lesson 6: Using Collections

Topic 6A: Work with the Collection Interface

Topic 6B: Work with Set Collection

Topic 6C: Work with List Collection

Topic 6D: Work with Queues

Topic 6E: Work with Map Collection
Topic 6F: Work with Collection Utilities

Lesson 7: Working with Generics

Topic 7A: Declare Generics
Topic 7B: Implement Generics

Lesson 8: Using Multi-Threaded Programs

Topic 8A: Examine a Multi-Threaded Program
Topic 8B: Deploy Threads
Topic 8C: Synchronize Threads

Lesson 9: Managing Java Applications

Topic 9A: Implement Garbage Collection
Topic 9B: Work with JAR Files

Appendix A: Java Standard Edition 6 Programmer Certified Professional Exam Mapping