Oracle - Java SE: Exploiting Modularity and Other New Features Ed 1.1

Code:	D99949GC11
Length:	2 days
URL:	View Online

Java SE: Exploiting Modularity and Other New Features introduces the Java module system and other new features, including JShell, convenience methods, new techniques for working with streams, and managing deprecated APIs. These features were introduced in versions after Java 8, and therefore new to developers migrating to Java 11.

Learn To:

- Design applications to take advantage of the module system and its more reliable configuration, improved security and performance, and more easily scalable applications. Migrate existing applications to a modular applications in a step-by-step manner, choosing which parts of the application to migrate first. Deal with common problems encountered in migrating an application, including, cyclic dependencies and split packages. Use services to make modularized applications more robust and easily extensible. Create multi-release JAR files that can be run on different Java releases. Use convenience methods to reduce code that seems verbose, inefficient or boilerplate, and increase readability. Use JShell to quickly run small code experiments and test new APIs.

Benefits To You

By enrolling in this course, you'll learn how to use the module system to design applications with explicit dependencies and encapsulation at the JAR level, ensuring more reliable configuration, improved security and enhanced performance. You'll also get a chance to experiment with new features that ease development. These include convenience methods that make your code more readable and succinct, and JShell, an easy way to test code snippets and APIs.

Skills Gained

- Identify deprecated APIs and possible alternatives
- Swap sub-optimal or tedious coding with convenience methods
- Create a modular Java application
- Run applications that combine modularized libraries and non-modularized libraries
- Create a custom runtime image
- Build Multi-release JAR files
- Design interfaces which implement methods
- Process stream data using new convenience methods
- Leverage JShell for fast code experiments
- Identify and apply new methods to more conveniently work with collections and arrays
- Identify and address common requirements in migrating older applications to modularity

Who Can Benefit

- Developer
- Manager

Prerequisites

Suggested Prerequisite

- Java SE 8 Fundamentals Ed 1
- Java SE 8 Programming Ed 1
- Develop applications using the Java programming languages
- Ability to use object-oriented programming techniques
- Ability to use classes commonly found in Java programs
- Administer operating systems from the command line

Required Prerequisite

- Familiarity with NetBeans or similar IDE
- Familiarity with Object-oriented programming concepts
- Familiarity with Java Collections and Enumerators
- Familiarity with JDK 8 features

Course Details

Topics

- Why Modules?
- Working with the Module System
- Modular JDK
- Creating Custom Runtime Images
- Migration
- Services
- Multi-release JAR files
- Private Interface Methods
- Enhancements to the Stream API
- JShell
- Convenience Methods for Collections
- Convenience Methods for Arrays
- Enhanced Deprecations for APIs

Download Whitepaper: Accelerate Your Modernization Efforts with a Cloud-Native Strategy

Get Your Free Copy Now

ExitCertified® Corporation and iMVP® are registered trademarks of ExitCertified ULC and ExitCertified Corporation, respectively
Copyright ©2024 ExitCertified ULC & ExitCertified Corporation. All Rights Reserved.