



Java: Intermediate-Advanced Java

Course ID #: 7000-492-ZZ-Z

Hours: 35

Course Content

Course Description:

Intermediate/Advanced Java includes an accelerated, yet thorough hands-on review of Java foundational concepts, with attention given to OO design and implementation principles. It then moves on to comprehensive coverage of more advanced topics in Java and OO development to provide participants with a strong grounding to use Java in a sophisticated and productive manner. This course covers far more than an introductory course, including important topics such as UML and Design Patterns, and using composition vs. inheritance, which are all key to creating well-structured OO systems. After these important areas, it moves on to advanced Java topics such as inner classes, reflection, writing generic classes, and functional programming with lambdas and streams. It teaches a number of useful techniques that enhance productivity and good system design - which may otherwise take Java developers years to absorb on their own. The course concludes with build tools and logging. Unit testing is stressed throughout the course, with most labs implemented as JUnit tests. All labs are done with the Eclipse IDE Java EE version, and the lab instructions include detailed directions for using it.

Course Objectives:

Upon successful completion of this course, students will:

- Understand UML and design patterns
- Know how to use composition and inheritance effectively
- Know Advanced language features: inner classes, writing generic classes, custom annotations, reflection
- Know Modern language features: lambda expressions and functional programming, Stream API, Java modules, type interference
- And so much more!

Prerequisites:

Basic knowledge of Java

Target Audience:

- Java developers
- OO developers



Java: Intermediate-Advanced Java

Course ID #: 7000-492-ZZ-Z

Hours: 35

Topics:

Lesson 1: Review - Basics

- Java Environment
- Classes and Objects
- Packages, Enums, Arrays
- Exceptions
- Data and Time API
- New Language Features

Lesson 2: Review – Inheritance and Interfaces

- UML Overview
- Inheritance
- Interfaces
- Guidelines

Lesson 3: JUnit

- Overview
- Tests, Assertions, and Fixtures
- Best Practices and TDD Overview [read for homework]
homework lab]

Lesson 4: Collections and Generics

- Collections Overview
- Lists, Sets, and Maps
- Writing Generic Classes

Lesson 5: Techniques and Object Creation

- Design Patterns Overview
- Controlling Object Creation
- Singleton Pattern
- Simple Factory
- Factory Method Pattern
- Other Techniques

Lesson 6: Using Composition and Inheritance Effectively

- Inheritance and Composition Pros and Cons
- Strategy Patterns
- Decorator Pattern
- Façade and Other Patterns

Lesson 7: Inner Classes

- Overview and Motivation
- Defining and Using Inner Classes
- Static Nested Classes

Lesson 8: Annotations

- Overview
- Using Annotations
- Writing Custom Annotations

Lesson 9: Reflection

- Overview and API
- Working with Objects Reflectively

Lesson 10: Lambda Expressions

- Overview
- Using Lambda Expressions
- Method References

Lesson 11: Streams

- Overview
- Understanding the Stream API
- Stream Processing
- Collectors



Java: Intermediate-Advanced Java

Course ID #: 7000-492-ZZ-Z

Hours: 35

Lesson 12: Introduction to Modules

- Motivation and Overview
- Types of Modules
- Modular JDK
- Our Approach

Lesson 13: Working with Modules

- Defining and Using Modules
- Services
- Compatibility and Migration
- Conclusion
- Recap

Register for this class by visiting us at:
www.tcworkshop.com or calling us at 800-639-3535