

TT3500: Mastering Test-Driven Development using JUnit (4 days)

Duration: 5 days

Skill Level: Intermediate

Focus: Java 5 or Java 6 Applications

Audience: Experienced Java programmers

Hands-On: Extensive hands-on programming labs; Expert lecture combined with open discussions and high-level demonstrations

Language / Tools: Java 5 or Java 6 delivered with most IDEs: Eclipse / Ganymede, Eclipse WTP, MyEclipse and more.

Delivery Format: Available for onsite private classroom presentation, or live online/virtual presentation

Customizable: Yes



Mastering Test Driven Development using JUnit is a four-day, comprehensive hands-on test-driven development / JUnit / TDD training course geared for developers who need to get up and running with essential Test-driven development programming skills using JUnit and various open-source testing frameworks. Throughout the course students learn the best practices for writing great programs in Java, using test-driven development techniques. This comprehensive course also covers essential TDD topics and skills.

► Course Objectives: What You'll Learn

Students who attend **Mastering Test-Driven Development using JUnit** will leave the course armed with the skills they require to develop solid Java programs, using sound coding testing techniques and best coding practices. This course quickly introduces developers to the features of JUnit and educates them regarding JUnit's strengths and weaknesses.

JUnit, and other testing frameworks based on JUnit such as Cactus, make it possible to write higher-quality Java code. It is a powerful tool designed to support robust, predictable and automated testing development in the Java enterprise application arena.

This course includes coverage of many of the essential JUnit capabilities, and can be tailored to focus exactly on the areas that you are interested in.

Working within in a dynamic, learning environment, guided by our expert TDD team, attendees will::

- Understand JUnit.
- Understand and use the JUnit Test Runner interface.
- Use JUnit to drive the implementation of Java code.
- Test applications using native IDE support.
- Best practices and patterns for test development.
- Understand JUnit's strengths and weaknesses
- Understand the role of debugging when done in conjunction with tests.
- Understand not only the fundamentals of the TDD using Java, but also its importance, uses, strengths and weaknesses.
- Understand the basics of JUnit, Cactus and other testing frameworks and how they relate to TDD.
- Learn to better control the development and quality of Java code.
- Understand how JUnit affects your perspective on

development and increases your focus on a task.

- Learn good JUnit coding style.
- Create well structured JUnit programs.
- Compile and execute programs using JUnit, Cactus, StrutsTestCase and DBUnit using the IDE of your choice.
- Understand how JUnit testing can be used for either state-based or interaction-based testing.
- How to extend testing with mock objects using EasyMock.
- Look at refactoring techniques available to make code as reusable/robust as possible.
- Discuss various testing techniques.

The following JUnit-based testing frameworks are examined:

- JUnit 3.8.1
- DBUnit
- jWebUnit
- StrutsTestCase
- EasyMock
- Cactus

► Experiential Learning – Course Structure

Throughout the course students will be led through a series of progressively advanced topics, where each topic consists of lecture, group discussion, comprehensive hands-on lab exercises, and lab review.

This workshop is about 50% hands-on lab and 50% lecture. **Multiple complete "mini-projects"** are laced throughout the course, designed to reinforce fundamental skills and concepts learned in the lessons, all working in the JUnit environment. Because these lessons, labs and projects are presented in a building block fashion, students will gain a solid understanding of not only the core concepts, but also how all the pieces fit together in a complete application.

► Audience & Pre-requisites: Who Should Attend

This is an **intermediate-to-advanced** level Java course, designed for developers who wish to get up and running on test-driven development immediately. Attendees should be familiar with Java and object-oriented technologies. Real world programming experience is a must.

► Related Courses – Suggested Options

Take Before: Students should have basic development skills and experience in the following topics, or attend these courses as a pre-requisite:

- **TT2100 Core Java Programming for OO Developers (C++, etc)**
- **TT5100 Building J2EE Web Applications (Servlets, JSPs, JDBC, Security, etc.)** (recommended but not required)

Take Instead: We offer other courses that provide different levels of knowledge or focus:

- Non-developers interested in test driven development should consider **TT3530 Implementing Agile Test Driven Development for Non-Developers**
- For a fast paced introduction to TDD essentials only consider **TT3510 Core Test Driven Development**

Take After: We offer a variety of introductory through advanced development, project management, engineering, architecture and design courses. Students may want to consider the following topics as a follow-on to this course. Please contact us for recommended next steps tailored to your longer term education, project or development objectives.

- **TT3520 Advanced Test Driven Development** or other TDD or Agile programming topics
- Advanced Java or JEE / J2EE topics such as EJB3.0; Spring; Hibernate; Design Patterns & more.
- Java / JEE Security topics
- Service-Oriented Analysis and Design
- Web Services – Intro through Advanced
- AJAX, XML or other Web Development topics
- Architecture & Analysis courses
- Software Engineering, Design or Project Management tracks

Please contact us for recommended next steps tailored to your longer term education, project or development objectives.

► Delivery Environment: Tools to Use

Although this training is skills-centric, this course can be delivered using a variety of software combinations, including but not limited to: Eclipse / Ganymede, MyEclipse, and HSQL. However, any IDE that includes JUnit can be used and any front-end web framework can be used as an implementation technology for the labs. This course may also run using Java 5 or Java 6. Please inquire for details

and options.

Our detailed lab workbooks are complete with software-specific instructions, screen shots and step-by-step tutorials for using the software you select. In most cases we can easily port our classes to run in the environment of your choosing.

► Easy Set Up! LoadNGo™ Instant Classroom Kit

For course deliveries or virtual presentation using open-source tools, we'll provide our unique **LoadNGo Instant Classroom Kit**, which enables students to run the entire course off of a DVD that hosts the entire course set up software, labs, and other pertinent useful educational resources, whitepapers and more. You only need to provide the hardware and appropriate O/S, and we'll do the rest. No installation needed. **Great for secure environments.** Minimum set up burden for your team or firm, with maximum results for your students.

No matter which set up option or software your firm requires, we're pleased to provide a detailed set up guide for all private or on-site courses, and as much assistance as you require to prepare your students or classroom for the course. Our support personnel and instructors can be contacted for any advice you may require to prepare your classroom and/or students for attendance.

► Student Materials: What You'll Receive

Our robust course materials include much more than a simple slideshow presentation handout. Student materials include a comprehensive hard-copy course manual, complete with detailed course notes, code samples, diagrams and current reference materials, all directly related to the course at hand, indexed for ease of use. Step-by-step lab instructions and project descriptions are clearly illustrated and commented for maximum learning.

In addition to everything students need for the course, the **LoadNGo Instant Classroom Kit** described above also includes of workshop labs and solutions; non-restricted workshop software, APIs, documentation, technical education papers, and specifications and tutorials pertinent to the training course. Our course kits are designed to serve as an excellent and useful reference set, long after we leave your classroom.

► Optional Pre / Post-Testing & Skills Assessment

We work with you to ensure that your resources are well spent. Through our basic course pre-testing and/or post-course assessments, we ensure your team is up to the challenges that this course offers. Our goal is to structure the best solution to ensure your needs are met, whether we customize the material, or devise a different educational path to prepare for this course.

► **Bridging the Gap: Collaborative Mentoring Services**

Our team of technical experts is also available for various project assistance services to help your team apply their newly-learned classroom skills to their real-world project in a meaningful, practical way, right after the training ends. This may be especially helpful for COBOL and Mainframe or non-programmer students moving to web based technologies, which can be an overwhelming prospect for some students, and their organizations.

Our custom **collaborative mentoring programs** integrate with or extend your team’s classroom training experience, to help bring

these skills into existing (or inherited) legacy projects, into new projects, or to simply keep your students sharp them in between projects. Our programs can be highly involved and closely integrated with your project timelines or group development efforts, or can be less involved, serving simply as an overarching educational framework or ‘spot check’ to keep your group skills moving forward in between projects or waiting for projects to begin.

Please contact us for details about this exciting custom service.

Workshop Topics Covered

Session: Test-Driven Development

- Overview of Test-driven Development
- The Problem
- The JUnit Solution
 - What is Unit Testing?
 - Purpose of Unit Testing
 - Successful Unit Testing
 - JUnit Overview
 - Reasons to Use JUnit
 - How JUnit works
 - Creating a TestCase
 - JUnit Methods to Override
 - Introducing Class Message
 - Creating Class MessageTest
 - The First Test Implementation Steps
 - The Initial Test Code
 - Testing the Constructor
 - Running a Test
 - Seeing the Results of a Test : JUnit View
 - Using the Results of a Test
 - Seeing the Results of a Successful Test
 - Defining a TestSuite
 - An Example TestSuite
- Test, code, refactor, repeat
- The ROI of TDD
 - Rationale for Test-driven Development
 - The Process of TDD
 - Advantages to TDD
 - Side-effects of TDD
 - Observations About Tests
- Tools to support TDD
- Automation and Coverage
- Working With Coverage Analysis
- Rationale
- Advantages

- Tools

Session: Testing Frameworks

- Integration Testing: jWebUnit/HttpUnit
- Presentation testing
- Integration testing
- jWebUnit
- jWebUnit/HttpUnit
- Testing Struts: StrutsTestCase
- Design of a Struts application
- StrutsTestCase
- Testing strategies

Session: Advanced TDD Topics

- Mock Objects and EasyMock
- Decoupling with Mock Objects
- Mock object frameworks
- EasyMock and JUnit
- Dependency Injection, Spring and Testing
- Dependency Injection and IoC
- The Spring Framework
- Mock Objects and Spring
- State-based vs. Interaction-based Testing
- State-based testing
- Interaction-based testing
- Dependencies vs. mock objects
- Interaction-based Testing
- JUnit and Ant
- Running JUnit Tests from Ant
- Generating a JUnitReport

Session: Improving Code Quality Through Refactoring

- Refactoring Overview
- Sample of Refactorings
- Refactoring and Testing

- Suggested Refactoring
- The Impact of Refactoring

Session: Additional Testing Frameworks

- Database Testing: DBUnit
- Issues related to database testing
- Persistence mechanisms
- DBUnit

Session: Introduction to Spring (optional)

- The Spring Framework
- Spring Basics
- Configuring a Spring bean
- Property Editors
- Constructor injection vs. Setter injection
- Wiring the collaborators

Session: Advanced Refactoring (optional)

- Design Patterns
- Code That Feels Wrong
- Refactoring to Design Patterns
- Abstract Factory Design Patterns
- Sample Refactorings
- Adapter Design Patterns
- Sample Refactorings
- Strategy Design Patterns
- Sample Refactorings

Session: Advanced Topics (optional)

- Testing Business Rules
 - Fit
 - Fitnesse
 - Selenium
- Adding Testing to Your Build Process
- The Ant JUnit Tag
- Example Ant Build File
- Continuous Integration

Session: Additional Testing Frameworks (optional)

- Server-side Testing: Cactus
- Server-side testing
- Cactus: mock-container testing
- Cactus: in-container testing

***Need more details?** Please note that a more detailed outline of the course table of contents, lists of lab exercises and project descriptions is available. Please contact us at Training@triveratech.com for info.*

***Need courseware?** This course is fully customizable, and also available for license with complete support for qualified organizations. Please contact Courseware@triveratech.com for details.*

► Why Work with Trivera Technologies?

- **We provide a solid application development and testing foundation.** Students will learn how to develop (and reuse!) essential design and testing skills and concepts properly, using best practices, grounding them for advanced curriculum. Students will be prepared for designing and implementing real solutions, right after the class ends. Students will learn the importance of developing and testing well-designed applications.
- **Our courses are focused - no "fluff" included.** We offer more than a “laundry list” approach to teaching. All lessons have clear objectives, are fundamental to core Agile and testing practices, and are reinforced by hands-on labs and solid practical examples. Each lesson has performance driven objectives that ensure students will learn technologies and skills core to fundamental testing and design – nothing more, nothing less.
- **Our materials are comprehensive, and current.** Our comprehensive manuals include not only a hard copy of the course presentation, but also detailed reference notes, pertinent diagrams and charts, current lists of suggested online resources and articles, and often technical tutorials or white papers geared to the topics at hand. Our dedicated course development team keeps everything as current as possible with both industry trends and software editions to ensure your team is getting the most current information available.
- **We set you up!** Hands-on courses also include our unique materials for each student, complete with our **LoadNGo Instant Classroom** course set up DVD, software, and a multitude of learning resources that complement the course. Run the course right off the DVD – minimal set up for your company – maximum results for your students.
- **We foster "Learning by Doing".** Progressive labs are designed in such a way that students get a firm grasp on fundamental skills while they work toward designing a complete application. All labs are take-home, and all solution code is presented in an easy to use self-study format for future use and review.
- **True content ownership gives us flexibility & quality above the rest.** These course materials are wholly-owned by our company and fully customizable - at little or no cost- to help you best meet your learning objectives. We have many dedicated experts available worldwide to instruct your team, and can provide services around the globe, either locally or virtually. We work closely with you to produce the most effective events and materials for your team, within your desired timeframe and budget.
- **We have to adhere to higher standards.** As a courseware provider, our content and hands-on lab materials are licensed internationally by dozens of firms, and are therefore subject to very stringent quality requirements. Not only will your organization benefit from our own technical team’s technical expertise, but also the feedback of hundreds of students and trainers using these materials, worldwide, on a regular basis. This unique fact guarantees that our materials are not only robust and interesting, but also technically correct, current and of the highest quality and usability.
- **We bring years of practical, current experience into the classroom and content.** Our instructors and course authors are also skilled mentors, Java, JEE/ JavaEE, J2EE, .Net, SOA, Agile and web services developers, architects and security-oriented professionals. We believe that learning, using and maintaining solid software execution and delivery methods are as important as gaining sharp coding skills. Best Practices for software development and execution, beyond technical coding skills, are enforced throughout all of our courses and discussions. Our team brings this extensive experience into every classroom and engagement.
- **We're skills-centric.** Although our team has extensive experience using a variety of tools and solutions, our core content is “technology-centric”. Our aim is to teach you the best skills and solutions out there – not to sell you software from any particular vendor.
- **We're Java & JEE / J2EE authors and industry speakers.** Our team was selected to write the online *J2EE, EJB, EJB CMP-CMR and Web Services Tutorial Series for IBM developerWorks®* (www.ibm.com) These are the same instructors who train our classes and author the courseware. Most of our trainers/consultants have also authored additional articles on web services, EJB, Struts, J2EE and advanced Java topics, and are recognized speakers and presenters on the industry technical seminar circuit. Our team is comprised on several successful published authors. Members of our team have written or contributed to: *Eclipse Kick Start, Mastering Eclipse; Professional Jakarta Struts; Using Java Tools for Extreme Programming; Mastering Resin; Mastering TomCat and others.*

- **Our services are guaranteed.** Whether you're a stakeholder organizing your firm's educational services, a student in our live or virtual classroom or a trainer using our materials to educate your own client or team – **Our core mission is to make YOU a success in the classroom.**

► **For Additional Information**

Need dedicated training? All courses can be brought onsite or produced virtually for a **private presentation**, customized to suit your unique requirements or goals. Please contact Training@triveratech.com for course details, Public Schedule dates and locations, and Special Discount Offers.



*Trivera Technologies is a 100%
Female-Owned Small Business Concern
GSA Schedule #GS-35F-0188T
Please contact us for details & pricing.*

Need courseware? Let us take the risk out of your classroom delivery! All materials are also available for corporate license with complete instructor support and free corporate branding. We guarantee our pricing and service. Samples of our course materials, as well as live client references for all of our services are available upon request. Please contact Courseware@triveratech.com for details.

For more information about our training, mentoring or courseware licensing options, or to see our complete list of course offerings and services, please visit us at www.triveratech.com, email Training@triveratech.com or call 609.953.1515.

•