SilverMark Course Outline

Title	**T201: Agile Testing for Java™ Developers**
Duration	2 Days
Audience	Software developers
Prerequisites	Knowledge of the Java™ programming language. Some knowledge of object-oriented design is helpful.
Objectives	Arm the developers on your team with the best practices, patterns and tools (many of them free) for delivering error free code. On completion of this course developers will write code that is more often correct, simpler, more maintainable and just plain better.
Description	This course introduces developers to techniques, tools, and patterns for using testing to deliver good, defect-free code as fast as possible. There are a great number of tools and practices for delivering defect-free code available to Java™ developers. Many of the tools are free but poorly documented and it is not often obvious how best to use them together. This course introduces the best and most useful tools and practices, where to apply them and how to use them. Some tools include JUnit, Cactus, EJBUnit, HTTPUnit, Clover, Jester, MockMaker, EasyMock, JFCUnit, Jemmy, Abbot, NoUnit, JUnit-addons, JUnitPerf, Enhanced JUnit, DBUnit, SQLUnit, FIT, FITnesse, WATT, ANT, Test Mentor, and JTest. This course also introduces an important technique called *Test-Driven Development*, which enables developers to write the tests that prove their code is correct before they write their code. Test-driven development is more than a testing technique. It is a way to drive design and implementation from the point of view of implementing running code that satisfies the tests. You will learn to use Test-driven development to rapidly evolve well-designed, easy to maintain code. \
Skill level	Intermediate
Prerequisites	Working knowledge of the Java™ programming language and a desire to do no more than the simplest thing that can possibly work.

Outline

Day 1:

- Agile development and agile testing
  - Introduction and agile manifesto
  - Relationship to eXtreme programming
  - Automation vs. manual testing
  - Continuous integration
- Developer and tester roles and collaboration
  - Unit and component testing vs. acceptance testing
- What to automate
  - Test whatever can break
  - Applying risk models
  - Taking context into account
- Automated testing Tools
  - JUnit
    - Architecture
    - Assertions
    - Writing tests and aggregating tests
    - Running tests
      - Differences between test runners
  - Enhanced JUnit
  - Test Mentor
  - JTest
- Test-driven development
  - Writing tests first
  - Growing code
  - Live demo of test-driven development
- Testability and component interface design
  - Architectural considerations for testability
  - Taking advantage of Java Interfaces
- Refactoring
  - Definition
  - When and how to refactor
  - Using tests to enable refactoring
- Adding automated testing to builds
  - Introduction to ANT
  - ANT's JUnit task
  - Reporting results
- Coverage Tools
  - Test Mentor and Enhanced Junit, Clover, JCoverage, Jester, NoUnit

Day 2:

- Test organization

	- Where to put tests - Projects, packages, and resources - Test initialization and cleanup - Sharing information between tests - Test implementation patterns - Configured instance,Aggregate state, Validation - Server-side testing - Tests on the server vs. tests as remote client - Cactus, HTTPUnit, Test Mentor - Load testing - JUnitPerf, Enhanced JUnit - Testing asynchronous processes - Database testing - DBUnit, SQLUnit - Testing private methods - JUnit extension, Inner classes,Test Mentor - Using mock objects - MockMaker, EasyMock, MockObjects framework, AgileTest - GUI testing - Thin GUIs and testability - GUI playback tools and frameworks - AWT Robot, JFCUnit, Jemmy, Abbot - Web browser testing - HTTPUnit - Logical capture/replay - Developer role in acceptance testing - FIT, FITNesse, WATT - Developing fixtures as a way to enable others to write acceptance tests - Further reading - Books and web sites
Labs	Each day is punctuated by hands-on labs, as well as live demonstrations of tools and techniques
Note	SilverMark provides consulting and mentoring services that help you apply the above tools and practices to your application. Save $200/day off of our standard consultant rate by signing up for these services with this course.

