

- [Table of Contents](#)

Test-Driven Development By Example

By [Kent Beck](#)

Publisher	: Addison Wesley
Pub Date	: November 08, 2002
ISBN	: 0-321-14653-0
Pages	: 240

[Copyright](#)

[Preface](#)

[Courage](#)

[Acknowledgments](#)

[Introduction](#)

[Part I. The Money Example](#)

[Chapter 1. Multi-Currency Money](#)

[Chapter 2. Degenerate Objects](#)

[Chapter 3. Equality for All](#)

[Chapter 4. Privacy](#)

[Chapter 5. Franc-ly Speaking](#)

[Chapter 6. Equality for All, Redux](#)

[Chapter 7. Apples and Oranges](#)

[Chapter 8. Makin' Objects](#)

[Chapter 9. Times We're Livin' In](#)

[Chapter 10. Interesting Times](#)

[Chapter 11. The Root of All Evil](#)

[Chapter 12. Addition, Finally](#)

[Chapter 13. Make It](#)

[Chapter 14. Change](#)

[Chapter 15. Mixed Currencies](#)

[Chapter 16. Abstraction, Finally](#)

[Chapter 17. Money Retrospective](#)

[What's Next?](#)

[Metaphor](#)

[JUnit Usage](#)

[Code Metrics](#)

[Process](#)

[Test Quality](#)

[One Last Review](#)

Part II. The xUnit Example

Chapter 18. First Steps to xUnit

Chapter 19. Set the Table

Chapter 20. Cleaning Up After

Chapter 21. Counting

Chapter 22. Dealing with Failure

Chapter 23. How Suite It Is

Chapter 24. xUnit Retrospective

Part III. Patterns for Test-Driven Development

Chapter 25. Test-Driven Development Patterns

Test (noun)

Isolated Test

Test List

Test First

Assert First

Test Data

Evident Data

Chapter 26. Red Bar Patterns

One Step Test

Starter Test

Explanation Test

Learning Test

Another Test

Regression Test

Break

Do Over

Cheap Desk, Nice Chair

Chapter 27. Testing Patterns

Child Test

Mock Object

Self Shunt

Log String

Crash Test Dummy

Broken Test

Clean Check-in

Chapter 28. Green Bar Patterns

Fake It ('Til You Make It)

Triangulate

Obvious Implementation

One to Many

Chapter 29. xUnit Patterns

[Assertion](#)
[Fixture](#)
[External Fixture](#)
[Test Method](#)
[Exception Test](#)
[All Tests](#)

[Chapter 30. Design Patterns](#)

[Command](#)
[Value Object](#)
[Null Object](#)
[Template Method](#)
[Pluggable Object](#)
[Pluggable Selector](#)
[Factory Method](#)
[Imposter](#)
[Composite](#)
[Collecting Parameter](#)
[Singleton](#)

[Chapter 31. Refactoring](#)

[Reconcile Differences](#)
[Isolate Change](#)
[Migrate Data](#)
[Extract Method](#)
[Inline Method](#)
[Extract Interface](#)
[Move Method](#)
[Method Object](#)
[Add Parameter](#)
[Method Parameter to Constructor Parameter](#)

[Chapter 32. Mastering TDD](#)

[How large should your steps be?](#)
[What don't you have to test?](#)
[How do you know if you have good tests?](#)
[How does TDD lead to frameworks?](#)
[How much feedback do you need?](#)
[When should you delete tests?](#)
[How do the programming language and environment influence TDD?](#)
[Can you test drive enormous systems?](#)
[Can you drive development with application-level tests?](#)
[How do you switch to TDD midstream?](#)
[Who is TDD intended for?](#)
[Is TDD sensitive to initial conditions?](#)
[How does TDD relate to patterns?](#)
[Why does TDD work?](#)

[What's with the name?](#)

[How does TDD relate to the practices of Extreme Programming?](#)

[Darach's Challenge](#)

[Appendix I. Influence Diagrams](#)

[Feedback](#)

[Appendix II. Fibonacci](#)

[Afterword](#)