



Robert C. Martin Series

Functional Design

Principles, Patterns, and Practices

Foreword by Janet A. Carr, Independent Clojure Consultant



Robert C. Martin

Functional Design

Functional Design: Principles, Patterns, and Practices

Table of Contents

Cover

Half Title

Contents

Title Page

Copyright Page

Foreword

Preface

Acknowledgments

About the Author

PART I: Functional Basics

Chapter 1 Immutability

What Is Functional Programming?

The Problem with Assignment

So Why Is It Called Functional?

No Change of State?

Immutability

Chapter 2 Persistent Data

On Cheating

Making Copies

Structural Sharing

Chapter 3 Recursion and Iteration

Table of Contents

Iteration

Very Brief Clojure Tutorial

Iteration

TCO, Clojure, and the JVM

Recursion

Chapter 4 Laziness

Lazy Accumulation

OK, but Why?

Coda

Chapter 5 Statefulness

When We MUST Mutate

Software Transactional Memory (STM)

Life Is Hard, Software Is Harder

PART II: Comparative Analysis

Chapter 6 Prime Factors

Java Version

Clojure Version

Conclusion

Chapter 7 Bowling Game

Java Version

Clojure Version

Conclusion

Chapter 8 Gossiping Bus Drivers

Java Solution

Driver

Route

Stop

Rumor

Simulation

Clojure

Table of Contents

Conclusion

Chapter 9 Object-Oriented Programming

Functional Payroll

Namespaces and Source Files

Conclusion

Chapter 10 Types

PART III: Functional Design

Chapter 11 Data Flow

Chapter 12 SOLID

The Single Responsibility Principle (SRP)

The Open-Closed Principle (OCP)

Functions

Objects with Vtables

Multi-methods

Independent Deployability

The Liskov Substitution Principle (LSP)

The ISA Rule

Nope!

The Representative Rule

The Interface Segregation Principle (ISP)

Dont Depend on Things You Dont Need

Why?

Conclusion

The Dependency Inversion Principle (DIP)

A Blast from the Past

A DIP Violation

Conclusion

PART IV: Functional Pragmatics

Chapter 13 Tests

But What about the REPL?

Table of Contents

What about Mocks?

Property-Based Testing

A Diagnostic Technique

Functional

Chapter 14 GUI

Turtle-Graphics in Quil

Chapter 15 Concurrency

Conclusion

PART V: Design Patterns

Chapter 16 Design Patterns Review

Patterns in Functional Programming

Abstract Server

Adapter

Is That Really an Adapter Object?

Command

Undo

Composite

Functional?

Decorator

Visitor

To Close, or to Clojure?

The 90-degree Problem

Abstract Factory

90 Degrees Again

Type Safety?

Conclusion

Postscript: OO Poison?

PART VI: Case Study

Chapter 17 Wa-Tor

Table of Contents

Scratch That Itch

Showers Solve Problems

Its Time to Wildly Reproduce

What about the Sharks?

Conclusion

Afterword

Index