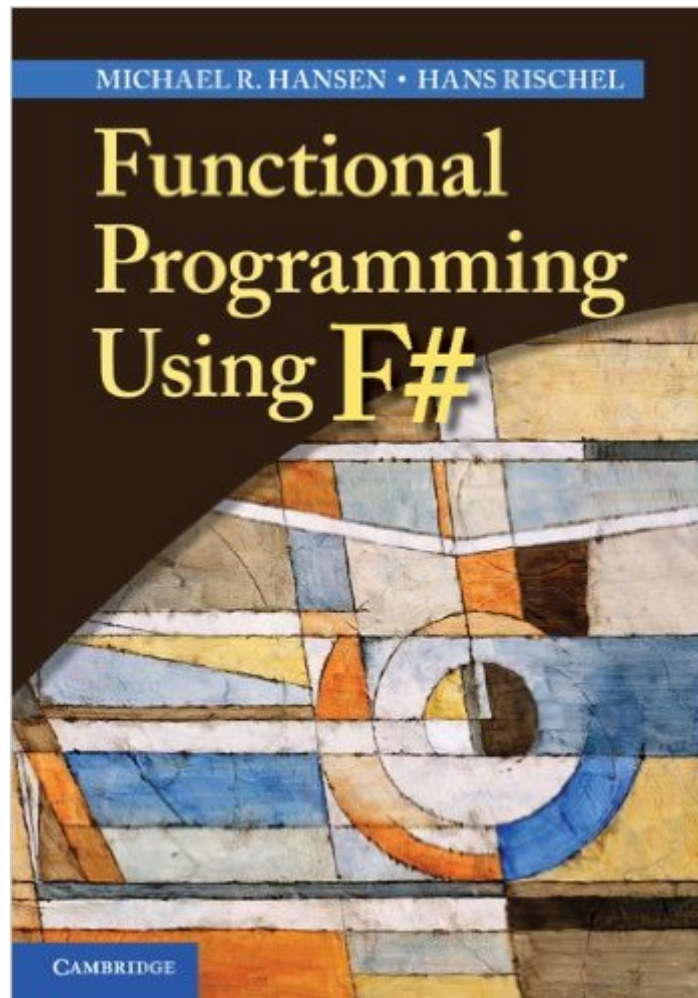


The book was found

# Functional Programming Using F#



## Synopsis

This comprehensive introduction to the principles of functional programming using F# shows how to apply basic theoretical concepts to produce succinct and elegant programs. It demonstrates the role of functional programming in a wide spectrum of applications including databases and systems. Coverage also includes advanced features in the .NET library, the imperative features of F# and topics such as text processing, sequences, computation expressions and asynchronous computation. With a broad spectrum of examples and exercises, the book is perfect for courses in functional programming and for self-study. Enhancing its use as a text is an accompanying website with downloadable programs, lecture slides, mini-projects and links to further F# sources.

## Book Information

File Size: 3376 KB

Print Length: 375 pages

Page Numbers Source ISBN: 1107019028

Simultaneous Device Usage: Up to 4 simultaneous devices, per publisher limits

Publisher: Cambridge University Press; 1 edition (April 30, 2013)

Publication Date: June 25, 2013

Sold by: Digital Services LLC

Language: English

ASIN: B00CARIB52

Text-to-Speech: Enabled

X-Ray: Not Enabled

Word Wise: Not Enabled

Lending: Not Enabled

Enhanced Typesetting: Not Enabled

Best Sellers Rank: #670,403 Paid in Kindle Store (See Top 100 Paid in Kindle Store) #87

in Books > Computers & Technology > Programming > Functional #1296 in Books > Computers

& Technology > Programming > Software Design, Testing & Engineering > Object-Oriented Design

#3391 in Kindle Store > Kindle eBooks > Computers & Technology > Programming

## Customer Reviews

This is a first rate survey of F# for academia. F# is in the ML family of languages. ML had already started supplanting Scheme as a lingua franca in academia in the early 80s, so a book on F# for academia makes a lot of sense. In fact, F#'s fastest path towards acceptance in industry might well

come via an indirect route through exposure in academia. If you're in academia, using F# instead of C# would typically be a no-brainer. (For instance, you'd have to be glutton for punishment to write a compiler in C# when you could use F# instead.) If you are a C# programmer, is this book for you? Or would an F# via C# book be better? That would depend on a lot of things; like personal preference, and how much of an academic background you have. One thing I like about academic books is how concise they are. (.Net books, by comparison, are typically verbose; with thousand page tomes being common.) Another thing I like about academic books are the examples; which are typically drawn from mathematics and computer science. (.Net books, by comparison, often have sophomoric examples with subject matter like zombies and cars; which I don't like.) The coverage of monads is excellent. The discussion on monadic parsers is great material. The fractal examples are fantastic; but showing them using Windows Forms (rather than XAML/WPF) seems schizophrenic to me. If you're cutting edge enough to want to use F#, why not also be cutting edge enough to use the latest and greatest GUI technology that Windows has to offer?! Conclusion: highly recommended. Eloquent, nicely done; but weak on applied aspects of F# (albeit that's not the focus nor purpose of this book).

This book is amazing. I just started learning F# in January for an independent study (I'm a grad student). I have Expert F# 3.0 and F# for Scientists - both are great but if I had to pick just ONE I would pick Functional Programming Using F#. I'm currently doing an internship, it's very hard/involved, and it's all in F#. This book connects a lot of dots in the "intermediate-expert" skill level range that were left unconnected by the other other two books. I had some serious nerd excitement over this book and I have a whole mess of books - nerd excitement is getting rarer. It was comparable to getting a new gadget or trying a new OS. It has examples, exercises, and an accompanying website - it is THE book to use for a class on the subject - not just because it fills a need for a functional intro with F# but because it is generally awesome as far as textbooks go. To give another analogy that should be relevant to some of you... Regarding my F# learning experience - reading this book was like the first time seeing calc based physics... "Ohhh! Well hell, I should have just started here first"

This is an excellent introduction to functional programming in F#. There are a lot of student exercises at the end of each chapter; a nice bonus for those using it as a college course text (let's hope many of you do just that). The order in which topics are covered builds in a very sensible way. Recursion and tail calls are done well early on. Everything relating to mutation is covered only in

chapter 8. The advantage this book has over most other F# books is (a) its concentration on purely functional programming (b) the authors have been teaching functional programming for quite some time and are able to teach the topics logically and well. If you are coming from an OO background you may want to make this your first F# book to help you rethink your programming habits. There are many other good F# books too but others don't have the same 1D-focus on functional programming as this. Move on to those other books after you're rewired your head a little with this. Although I say 'rewired' I don't want to give the impression that functional programming is hard. FP is no harder than object oriented programming. Initially, functional programming seems harder than imperative programming but, as your imperative code grows it has a tendency to turn into a minefield making it harder to understand; every mutable property you expose turns on you to become yet another expanded dimension of a combinatorial complexity hell dragging you down. Well that's what I think of imperative programming. There are two annoyances knocking off 1 star. It is let down by too many typos (which a clever reader will be able to correct for herself). The ebook version is somewhat hard for me to read because the many image files all look too small; OK if it's read on a PC though.

When learning a new language I tend to read a handful of books before I start doing anything serious. The other related books were either out of date, poorly written or strangely ordered such that I had to skip ahead to understand what I was currently reading. So I quickly tired of reading them. I enjoyed this book because the writing is straightforward with relevant examples for each topic, a final summary and a series of exercises at the end of each chapter. Clearly the authors know how to present the material in a way that makes it easy to learn. I found myself engaged in the material and liked the pacing which was perfect for chipping away at my imperative prejudices. Since I consider myself an expert in OO and imperative programming I was pleased that the authors stayed focussed on functional programming and kept imperative features to a minimum. However I'm subtracting 1 star for the following reasons: 1. I have the kindle version of the book and occasionally get annoyed by the typographical errors. I'm an experienced programmer so I didn't get roadblocked but it is annoying none the less. 2. The math does not render clearly on a Kindle tablet, however looks fine on an iPad. 3. I was unable to find an answer key for the exercises on the author's website. Although this is minor, if provided with exercises I would like to check my results. 4. The chapter on sequences (chapter 11) should have been introduced earlier, ideally just before or after the chapter on collections (chapter 5). I suppose this is because the book was translated from an earlier book on ML and the F# specific stuff was added at the end. Regardless of

my dislikes, I do strongly recommend this book.

[Download to continue reading...](#)

Programming #8:C Programming Success in a Day & Android Programming In a Day! (C Programming, C++programming, C++ programming language, Android , Android Programming, Android Games) Programming #57: C++ Programming Professional Made Easy & Android Programming in a Day (C++ Programming, C++ Language, C++for beginners, C++, Programming ... Programming, Android, C, C Programming) Programming #45: Python Programming Professional Made Easy & Android Programming In a Day! (Python Programming, Python Language, Python for beginners, ... Programming Languages, Android Programming) Functional Programming in JavaScript: How to improve your JavaScript programs using functional techniques Programming: Computer Programming for Beginners: Learn the Basics of Java, SQL & C++ - 3. Edition (Coding, C Programming, Java Programming, SQL Programming, JavaScript, Python, PHP) Raspberry Pi 2: Raspberry Pi 2 Programming Made Easy (Raspberry Pi, Android Programming, Programming, Linux, Unix, C Programming, C+ Programming) Android: Programming in a Day! The Power Guide for Beginners In Android App Programming (Android, Android Programming, App Development, Android App Development, ... App Programming, Rails, Ruby Programming) DOS: Programming Success in a Day: Beginners guide to fast, easy and efficient learning of DOS programming (DOS, ADA, Programming, DOS Programming, ADA ... LINUX, RPG, ADA Programming, Android, JAVA) ASP.NET: Programming success in a day: Beginners guide to fast, easy and efficient learning of ASP.NET programming (ASP.NET, ASP.NET Programming, ASP.NET ... ADA, Web Programming, Programming) C#: Programming Success in a Day: Beginners guide to fast, easy and efficient learning of C# programming (C#, C# Programming, C++ Programming, C++, C, C Programming, C# Language, C# Guide, C# Coding) FORTRAN Programming success in a day:Beginners guide to fast, easy and efficient learning of FORTRAN programming (Fortran, C++, C, C programming, ... Programming, MYSQL, SQL Programming) Prolog Programming; Success in a Day: Beginners Guide to Fast, Easy and Efficient Learning of Prolog Programming (Prolog, Prolog Programming, Prolog Logic, ... Programming, Programming Code, Java) R Programming: Learn R Programming In A DAY! - The Ultimate Crash Course to Learning the Basics of R Programming Language In No Time (R, R Programming, ... Course, R Programming Development Book 1) Parallel Programming: Success in a Day: Beginners' Guide to Fast, Easy, and Efficient Learning of Parallel Programming (Parallel Programming, Programming, ... C++ Programming, Multiprocessor, MPI) MYSQL Programming Professional Made Easy 2nd Edition: Expert MYSQL Programming Language Success in a Day for any Computer User! (MYSQL, Android programming, ... JavaScript,

Programming, Computer Software) Programming Raspberry Pi 3: Getting Started With Python (Programming Raspberry Pi 3, Raspberry Pi 3 User Guide, Python Programming, Raspberry Pi 3 with Python Programming) VBScript: Programming Success in a Day: Beginner's Guide to Fast, Easy and Efficient Learning of VBScript Programming (VBScript, ADA, ASP.NET, C#, ADA ... ASP.NET Programming, Programming, C++, C) Apps: Mobile App Trends in 2015 (iOS, Xcode Programming, App Development, iOS App Development, App Programming, Swift, Without Coding) ((Android, Android ... App Programming, Rails, Ruby Programming)) XML Programming Success in a Day: Beginner's Guide to Fast, Easy, and Efficient Learning of XML Programming (XML, XML Programming, Programming, XML Guide, ... XSL, DTD's, Schemas, HTML5, JavaScript) Functional Programming Using F#

[Dmca](#)