Book Review

Mattias Felleisen and Daniel P. Friedman
The Little MLer. MIT Press. 1998. ISBN 0-262-56114-X. £14.95. 182pp. softbound.

What is ML?	ML is what people call a *functional* programming language. Many universities use it to teach programming.
Does this book teach programming?	Mainly it talks about types, pattern-matching and functional programming techniques.
May I see the table of contents?	Don't bother. The chapters have titles like 'Oh My, It's Full of Stars!'
Then what does the book cover?	Types including recursive types; pattern-matching and recursion; building recursive data structures; tuples; trees and mutual recursion; functions as values and lazy lists; curried and higher-order functions; exception handling; ML modules.
That sounds like a lot. But why is the book written in this silly style? Is it one of those programmed learning guides?	Heavens no. The dialogue format is good for bringing out tricky points and for walking you through mechanical tasks like type inference.
I see. It's another example of dumbing-down.	That isn't fair. The book covers hard concepts such as type identity with functors, using sharing constraints and **where type** specifications.
But does this writing style really work?	Who knows, but this book is just the latest of a series that includes *The Little LISPer* and *The Little Schemer.*
What do you like about the book?	Concepts are introduced with great care. Although I don't see the point of the *combine_s* example of Chapter 8, most of the examples work well.
And what don't you like?	It never gives the impression that ML could do something useful. I'm afraid that when readers want to solve an actual problem, they will just turn to C.
Maybe the book is intended to teach fundamental issues rather than programming.	Yes, there is little on efficiency, debugging or any notion of software correctness. Although the programs are developed step-by-step, the book doesn't really teach design.

I think the book is still valuable as a supplementary text. It covers some tricky concepts.	Yes, it can't be a course's only text, but some students would benefit from reading it.

L. C. Paulson
University of Cambridge