



# Foreword

*by Michael Kölling*

Teaching to program well is a hard challenge. Writing a book about it is a difficult undertaking.

The bulk of my own experience in programming teaching is at the introductory level. I see my students leave the first programming course, many of them thinking they are good programmers now. Most of them are not. Only the good ones realise how much they have yet to learn. Learning how to build good quality software is much more than mastering the syntax and semantics of a language. This book is about the next phase of learning they are about to face.

Most academic discussion about the teaching of programming revolves around introductory teaching in the first semester of study, and by far the largest number of books on programming cover the beginners' aspects. Many fewer books are available that cover more advanced aspects—as this one does—and even fewer do it well.

The reason is just that introductory programming is easier to handle, and still so difficult that for many years we—as a teaching community—could not agree how to approach the teaching of modern, object-oriented programming in a technically and pedagogically sound manner. It has taken well over 10 years and more than one hundred published introductory textbooks on learning object orientation with Java alone to get to where we are now: a state where introductory texts are available that are not only a variation of a commented language specification, but that follow sound pedagogical approaches, that are written with learners in mind, that emphasise process over product, and that deal with real problems from real contexts.

For more advanced programming books—usable in advanced programming courses—the situation is less rosy. There is much less agreement about the topics that such a course should cover, and fewer authors have taken the difficult step to write such a book. Many programming books at this level are in character where introductory books were ten years ago: Descriptions of techniques and technologies, written with great emphasis on technical aspects, but with little pedagogical consideration.

This book is a refreshing change in this pattern. This book brings together a careful selection of topics that are relevant, indeed crucial, for developing good quality software with a carefully designed pedagogy that leads the reader through an experience of active learning. The emphasis in the content is on practical goals—how to construct reliable and flexible software systems—covering many topics that every

software engineer should have studied. The emphasis in the method is on providing a practical context, hands on projects, and guidance on *process*.

This last point—process—is crucial. The text discusses not only what the end product should be like, but also how to get there.

I know that this book will be a great help for many of my students on the path from a novice programmer to a mature, professional software developer.

—Michael Kölling  
Originator of the BlueJ and Greenfoot Environments.  
Author and coauthor of the best-selling books  
*Objects First with Java* and  
*Introduction to Programming with Greenfoot*.