

Foreword



Consider the following observations:

- Software continues to become ever more pervasive, ever more ubiquitous in our lives.
- Incompetence seems to be the only thing we can count on in today's world and, especially, in the domain of software.
- The Java programming language has become a lingua franca for programmers all over the world.

One can draw varied conclusions from these comments. One of them is that it is of great importance that programmers working with the Java programming language should be as competent as possible.

The Java certification program is an important effort aimed at precisely this goal. Practitioners looking to obtain such certification need good quality training materials, which brings us to this book.

Programming is still more of an art than a science, and will continue to be so for the foreseeable future. Mastering the intricacies of a large and complex programming language is a challenging task that requires time and effort, and above all experience.

Real programming requires more than just mastery of a programming language. It requires mastery of a computing platform, with a rich set of libraries. These libraries are designed to simplify the task of building realistic applications, and they do. Again, the practitioner is faced with a daunting task.

To address the clear need for professional training material, a plethora of books have been written purporting to tutor programmers in the programming language and platform skills they require.

The choice is as mind boggling as the material within the books themselves. Should one try *Java for Frontally Lobotomized Simians* or *Postmodern Java Dialectics*? The readership for these books is largely self selecting. I trust that if you, the reader,

have gotten this far, you are looking for something that is intelligent, yet practical. This book is one of the finest efforts in this crowded arena. It brings a necessary level of academic rigor to an area much in need of it, while retaining an essentially pragmatic flavor.

The material in this book is probably all you need to pass the Java certification exam. It certainly isn't all you need to be a good software engineer. You must continue learning about new technologies. The hardest part of this is dealing with things that are completely different from what you are familiar with. Yet this is what distinguishes the top flight engineer from the mediocre one. Keep an open mind; it pays.

Gilad Bracha

Computational Theologist

Sun Java Software

<http://java.sun.com/people/gbracha/>