A Programmer's Introduction to C#
by: Eric Gunnerson
Second Edition Updates

If you already own the first edition of this book, you're probably asking yourself whether the second edition is worth the money. This section should give you enough information to make that choice.

The first thing I did was go through the book and update all the samples to conform to the Beta 2 release of the compiler. Most of the changes there are fairly minor, mostly due to naming changes in the Frameworks, though some of the samples did require a bit of re-architecting.

The second set of changes are the minor changes listed below. These changes typically involve the addition of small sections or new examples.
In the area of more major changes, I heavily revised the chapters on delegates and events and developed a sample application using Windows Forms. There is a new chapter on threading and asynchronous operations, which details two different ways of getting things to occur simultaneously. Finally, there's a chapter on execution-time code generation, which details how to write a self-modifying application.

The following are the major differences:

- Chapters 22 and 23 covering delegates and events have been rewritten.
- Chapter 29 covers threading and asynchronous operations.
- Chapter 30 covers execution-time code generation using several techniques.
- Chapters 33-35 describe the development of a Windows Forms (formerly know as "WinForms") application

There are also a number of minor differences in existing chapters:

- Chapter 8 covers deterministic finalization, IDisposable, and the using statement.
- Chapter 9 discusses immutable classes
- Chapter 10 covers structs that implement interfaces
- Chapter 11 provides more detail about versioning.
- Chapter 14 discusses checked and unchecked expressions.
- Chapter 17 covers string encodings, string interning, and regular expression options.
- Chapter 18 covers virtual properties.
- Chapter 19 now covers strongly-typed enumerators, disposable enumerators, and multi-dimensional indexers.
- Chapter 20 now covers some nice things enums get from the System.Enum class.
- Chapter 25 presents a complex number class.
- Chapter 27 has design guidelines for overriding object.Equals().
- Chapter 28 covers synchronized and case-insensitive collections.
- Chapter 31 provides interop design guidelines and more examples
- Chapter 32 contains information on starting processes, doing custom serialization, and accessing environment strings.
- Chapter 38 describes the new default compiler response file.