Where Do You Go from Here?

Now that you’ve mastered the fundamentals of C, you’re next step is to dig into the specifics of Mac programming. As you ran the example programs in the previous chapters, you probably noticed that none of the programs sport the look and feel that make a Mac program a Mac program. For one thing, all of the interaction between you and your program focuses on the keyboard and the console window. None of the programs take advantage of the mouse. None offer color graphics, pull-down menus, buttons, checkboxes, scrolling windows, or any of the thousand things that make OS X applications so special. These things are all part of the Mac user interface.

In short, the book has, so far, only taught basic C programming on a Mac computer. This is fundamentally different from creating programs that have a Mac GUI interface. That’s the next logical step in the process, and that’s what this chapter is about.

The Mac User Interface

The user interface is the part of your program that interacts with the user. So far, your user interface skills have focused on writing to and reading from the console window using functions such as `printf()`, `scanf()`, and `getchar()`. The advantage of this type of user interface is that each of the aforementioned functions is available on every machine that supports the C language. Programs written using the standard library are extremely portable.
On the down side, console-based user interfaces tend to be limited. With a console-based interface, you can’t use an elegant graphic to make a point. Text-based interfaces can’t provide animation or digital sound. In a nutshell, the console-based interface is simple and, at the same time, simple to program. OS X’s graphical user interface (GUI) offers an elegant, more sophisticated method of working with a computer.

Learning Objective-C

Your Mac programs just wouldn’t be the same without windows, drop-down and pop-up menus, icons, buttons, and scroll bars. You can and should add these user interface elements to your programs. Fortunately, the set of Apple developer tools you downloaded and installed at the beginning of this book includes everything you need to build world-class applications with all the elements that make the Mac great!

The key to working with these elements lies in understanding Objective-C and Cocoa. The Objective-C language is a superset of C. This means that everything you just learned about C will work in Objective-C! There are a number of excellent resources available for learning Objective-C. One of them is just a mouse click away.

Choose the Documentation and API Reference command from the Xcode Help menu. This will open Xcode’s documentation browser. In the search field, type in “Programming Objective-C” and press Return. In the list of results, locate two documents: Programming with Objective-C and The Objective-C Programming Language. Start with the first if you want to get a feel for what Objective-C has to offer. Jump to the second if you want to start learning the language. If you’ve never done any object oriented programming, the second document has a link to the Object Oriented Programming with Objective-C guide, which is a great primer on the advantages of objects.

We love these documents. They are very well written and detailed, and best of all, they’re free! Take a few minutes to read through the first few pages. If you feel comfortable with the language and the tone, you’ve found your path to learning Objective-C.

If this document makes your eyes glaze over and you start to feel a bit queasy, there are plenty of other ways to learn Objective-C. If you like the experience you had reading this book, check out its companion book from Apress called Learn Objective-C on the Mac by Mark Dalrymple and Scott Knaster (2009). Mark and Scott are two of the smartest people we know, and they do an excellent job explaining the concepts behind the Objective-C language.