

Introduction

The Xcode Tools contain everything you need to create Mac OS X and iOS applications. As powerful as the Xcode Tools are, they can be difficult to learn. Most Mac OS X and iOS development books focus on Cocoa and Cocoa Touch programming, as they should. They teach enough Xcode for you to create the projects in the book. But there's more to writing real applications than writing code. Real applications must be tested and debugged to make sure they run properly and must be profiled to make sure they run fast enough. A Cocoa or iOS programming book isn't going to show you how to profile your program or find memory leaks in it.

Xcode Tools Sensei picks up where other books leave off. It teaches the Xcode Tools, not a particular language or programming framework. By reading this book you can spend more time writing, debugging, and profiling your applications and less time searching and reading documentation.

Getting Xcode 4

Xcode Tools Sensei covers Xcode 4, which requires an Intel Mac running Mac OS X 10.6 or later. You can download the latest version of Xcode from the Mac App Store. When you download Xcode from the App Store, it is installed in your Applications folder.

Use Apple's developer site to download older versions of Xcode. The most common reason to download an older version of Xcode is that you're running an older version of Mac OS X. If you're running Mac OS X 10.7 or later, you must sign up for a free ADC membership to download Xcode from Apple's developer site. If you are running Mac OS X 10.6, you must be a member of one of Apple's paid developer programs, iOS or Mac, to download Xcode 4. Xcode 4.2 is the latest version that runs on Mac OS X 10.6.

Xcode 4.3 and later versions are packaged as a single application that you install in your Applications folder. Older versions require you to run an installer. In most cases after running the installer, you can find Xcode in the following location:

`/Developer/Applications`

Accessing Other Developer Tools

Starting with Xcode 4.3, Apple packages Xcode as a single application instead of creating a `Developer` folder on your startup disk. The change in packaging makes finding the other developer tools more difficult. You can launch other developer tools, such as Instruments, by choosing Xcode > Open Developer Tool.

The Mac OpenGL tools I cover in the book are not packaged with Xcode. Choose Xcode > Open Developer Tool > More Developer Tools to go to Apple's developer downloads page. Download the Graphics Tools for Xcode package to install the OpenGL tools. If you want to compile code from the command line, download the Command Line Tools for Xcode package.

The Book's Contents

Chapter 1 covers Xcode projects. In this chapter you will learn how to create a project, add files to a project, create workspaces, and use the Organizer. You will also learn about the types of projects you can create in Xcode and Xcode's project window.

Chapter 2 covers code editing. Some of the topics covered in this chapter include Xcode's editor, code completion, code snippets, tabbed editing, refactoring tools, and reading developer documentation.

Chapters 3 and 4 cover Interface Builder, which is now integrated with Xcode in Xcode 4. In Chapter 3 you will learn how to use Interface Builder to create user interfaces for Mac applications. In Chapter 4 you will learn how to use Interface Builder to create user interfaces for iOS applications.

Chapter 5 covers Xcode's data modeling tools, which work with the Core Data framework. In this chapter you will learn how to create data models and mapping models, which migrate data from an older version of a data model to a newer version.

Chapter 6 covers building projects. Some of the topics covered in this chapter include the project editor, targets, build settings, schemes, configuration settings files, and static analysis.

Chapter 7 covers debugging. In this chapter you will learn how to set up your scheme for debugging, set breakpoints, view variables, step through code, and use the debug console with both GDB and LLDB.

24 Introduction

Chapter 8 covers version control using git and Subversion. Some of the topics covered in this chapter include creating a version control repository, adding an Xcode project to a repository, viewing changes in versions of a file, adding files to a repository, committing changes to a repository, branching, pushing to and pulling from remote git repositories, and snapshots.

Chapter 9 covers Instruments, which is a tool for tracing applications. With Instruments you can do things like check for memory leaks, find out how much memory your application is using, and determine where your application is spending its time.

Chapter 10 covers command-line debugging tools: `fs_usage`, `sc_usage`, `vmmmap`, `heap`, `leaks`, and `malloc_history`. If you've never heard of these tools before, don't worry. After reading Chapter 10 you'll become intimately familiar with them.

Chapter 11 covers the OpenGL tools: OpenGL Profiler, OpenGL Driver Monitor, OpenGL Shader Builder, and OpenGL ES Performance Detective. OpenGL Profiler profiles and debugs OpenGL applications. OpenGL Driver Monitor displays realtime statistics about your graphics card. OpenGL Shader Builder is the tool to create OpenGL shaders to give your OpenGL applications more control over drawing a scene. OpenGL ES Performance Detective investigates your OpenGL ES application for possible performance problems. If it finds a problem, OpenGL ES Performance Detective lists the most likely causes of the problem.

Xcode Preferences

Throughout the book I make references to Xcode's preferences, such as its Text Editing preferences. Choose Xcode > Preferences to open Xcode's preferences window. Xcode's preferences window has the following sections:

- General
- Behaviors
- Fonts and Colors
- Text Editing
- Key Bindings
- Downloads
- Locations

Older versions of Xcode may have Documentation and Source Trees preferences instead of Downloads preferences. Older versions of Xcode may also have Distributed Builds preferences.

What the Reader Needs to Know

I assume the reader has some experience writing Mac or iOS programs. Mac OS X and iOS development are large topics. There is no way I could adequately cover programming and the Xcode Tools in one book.

Those of you new to programming, Mac development, or iOS development should buy another book or two in addition to this book. You can find programming books and reviews at Amazon. I also provide links to publishers in the following post on my blog:

<http://meandmark.com/blog/2010/01/getting-started-with-mac-programming/>

Some Things to Keep in Mind as You Read This Book

The Xcode Tools have multiple methods to perform many tasks. Rather than detail each possible way to accomplish a task, I usually mention only one of the methods in the chapter text. Just because I mention one way to do something doesn't mean the other methods are worse. You may find it easier to complete a task differently than the way I explain in the book.

Apple is constantly making changes to the Xcode Tools. The constant changes mean that screenshots, labels, control names, and menu item names in the book may be different than what you see on your Mac.

I mention right-clicking many times in the book. Every Intel Mac should have either a mouse or a trackpad that allows right-clicking to open contextual menus. If your mouse or trackpad doesn't support right-clicking, hold down the Control key while clicking to open contextual menus.

Xcode 4.4 and 4.5 are available for both Mac OS X 10.7 and 10.8. Some Xcode 4.4 and 4.5 features may not be available in Mac OS X 10.7. Xcode 4.1 and 4.2 are available for both Mac OS X 10.6 and 10.7. Some Xcode 4.1 and 4.2 features may not be available in Mac OS X 10.6.

For the latest updates go to this book's official website.

<http://www.meandmark.com/xcodebook.html>

If you have any questions or comments about the book, feel free to email me at the address below. I'll do my best to answer any questions.

xcodebook@meandmark.com