Get started building your very first
iPhone or iPad apps

iPhone and iPad Apps for Absolute Beginners

THIRD EDITION

Dr. Rory Lewis | Chad Mello
Foreword by Ben Easton

Apress®

iPhone and iPad Apps for Absolute Beginners, 3rd Edition

Dr. Rory Lewis and Chad Mello

Apress®

iPhone and iPad Apps for Absolute Beginners, 3rd Edition

Copyright © 2013 by Dr. Rory Lewis and Chad Mello

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

ISBN 978-1-4302-4617-6

ISBN 978-1-4302-4618-3 (eBook)

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image, we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

President and Publisher: Paul Manning
Lead Editor: Steve Anglin
Development Editor: Matthew Moodie
Technical Reviewer: Matthew Knott
Editorial Board: Steve Anglin, Mark Beckner, Ewan Buckingham, Gary Cornell, Morgan Engel, Jonathan Gennick, Jonathan Hassell, Robert Hutchinson, Michelle Lowman, James Markham, Matthew Moodie, Jeff Olson, Jeffrey Pepper, Douglas Pundick, Ben Renow-Clarke, Dominic Shakeshaft, Gwenan Spearing, Matt Wade, Tom Welsh
Coordinating Editor: Christine Ricketts
Copy Editor: Corbin Collins
Compositor: SPi Global
Indexer: SPi Global
Cover Designer: Anna Ishchenko

Distributed to the book trade worldwide by Springer Science+Business Media, LLC., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com.

For information on translations, please e-mail rights@apress.com, or visit www.apress.com.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales–eBook Licensing web page at www.apress.com/bulk-sales.

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author(s) nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

To my best friend, my wife, my life, my light, my Kera.

—Dr. Rory Lewis

To my wife, Imelda. I can see nothing but beauty, eclipsed only by your grace. You have all of my love, admiration, and respect; Ab imo pectore.

—Chad Mello

Contents at a Glance

Foreword: About the Author xiii

About the Contributing Authors xvii

Acknowledgments xix

Preface xxi

Chapter 1: Preliminaries 1

Chapter 2: Blast-Off! 25

Chapter 3: Keep on Truckin' 49

Chapter 4: Buttons and Labels with Multiple Graphics 83

Chapter 5: Touches 123

Chapter 6: Switches 171

Chapter 7: Storyboards 221

Chapter 8: Introducing MapKit 257

Chapter 9: MapKit with a Little More Effort 285

Chapter 10: Storyboarding to Multimedia Platforms 329

Index 359

Contents

Foreword: About the Author ..... **xiii**

About the Contributing Authors ..... **xvii**

Acknowledgments ..... **xix**

Preface ..... **xxi**

Chapter 1: Preliminaries **1**

Necessities and Accessories 1

Getting a Mac 2

Getting OS X 3

Becoming a Developer 6

Getting Ready for Your First iPhone/iPad Project 17

Installing DemoMonkey 20

Chapter 2: Blast-Off! **25**

Running your App on the iPhone Simulator 25

Running Your iPhone App on the iPad Simulator 42

Running Your iPad App on the iPad Simulator 46

Digging the Code 48

Chapter 3: Keep on Truckin' 49

helloWorld_03: An Interactive Single View App 49

Creating the User Interface 53

Connecting to the Code 59

Digging the Code: -(IBAction)button: (id)sender; 64

Setting up the Coding Environment 64

Creating a Programming Roadmap 67

Digging the Code 75

Nibs, Zibs, and Xibs 76

Methods 78

Header Files 78

The Inspector Bar 79

Chapter 4: Buttons and Labels with Multiple Graphics 83

Roadmap Recap 84

helloWorld_04: A View-Based Application 84

Understanding IBOutlets and IBActions 103

Using pointers 109

Properties: Management and Control 111

Adding IBActions 112

Digging the Code 118

IBOutlets and IBActions 118

More About Pointers 119

Chapter 5: Touches 123

Redefining "Giving Up" 123

Roadmap Recap 124

Touches: A View-Based Application 125

CGAffineTransform Structs 125

Coding the Implementation File 148

Running the Code 167

Digging the Code 170

Chapter 6: Switches 171
Don't Freak Out: Let's Look at Your Brain! 171
switches: A Tabbed Application 174
Obtaining the Resources 174
Creating the App 175
Adding the Images to the Project 177
Running the App 181
Customizing the Views 183
Customizing the Buttons 189
Creating the Second View 194
Connecting and Coding the Buttons 199
Digging the Code 219
Compile-time and Runtime 219
Chapter 7: Storyboards 221
Storytelling 221
Roadmap Recap 222
Evolve: A View-Based Storyboard Application 222
Phase I: Creating Core Storyboarding Configurations 223
Phase II: Setting Up and Establishing the Content of the View Controllers 237
Phase III: Working on Closure and Coding 248
Chapter 8: Introducing MapKit 257
A Little About Frameworks 257
MapKit Showcase: Preinstalled Apps 258
Turn-by-Turn Navigation 259
Traffic: A Smarter Way to Cope 260
Local Search 261
See More, Do More 262
Innovative and Popular MapKit Apps Inspire Us 263
MyMapKit_01: A Universal Single View Application 265
Preliminaries 265

A New Single View Template 265
Preliminaries: Adding the MapKit Framework 268
Bring in the Images! 270
Finishing the View Controllers 276
Making It Go: Running Your First MapKit App 279
Time to Show Off! 283
Chapter 9: MapKit with a Little More Effort 285
Understanding Map Geometry and Coordinate Systems 285
Important Things to Know 288
myMapkit_02: A Single View iPad Application 288
Possible Prepping for the App 289
Preliminaries 290
A New Single View Template 290
Preliminaries: Bringing in the Images 292
Preliminaries: Adding MapKit and Arranging the Project 294
Preliminaries: Adding the Annotation Class 297
Coding the myPos Annotation Class 303
The User Interface 306
Coding the View Controller 311
Dealing with the View Controller's Implementation 314
Modifying the AppDelegate.m Implementation file 320
The End Result: Running Your iPad MapKit App 321
Source for ViewController Implementation 325
Chapter 10: Storyboarding to Multimedia Platforms 329
myiTunes: A Master-Detail Application (iPad) 330
Preliminaries 330
A New Master-Detail Template 331
Bring in the Images! 332

Organize the Popover in Storyboard 334
Coding the myiTunes App 342
Coding MasterViewController 344
Coding DetailViewController 352
Finalizing the Storyboard 353
In Ending 357

Index 359

Foreword: About the Author

"Rory and I met in L.A. in 1983. He reminds me of one of my favorite film characters, Buckaroo Banzai—always going in six directions at once. If you stop him and ask what he's doing, he'll answer comprehensively and with amazing detail. Disciplined, colorful, and friendly, he has the uncanny ability to explain the highly abstract in simple, organic terms. He always accomplishes what he sets out to do, and he'll help you do the same."

Why You'll Relate to Dr. Lewis

While attending Syracuse University as a computer engineering student, Rory scrambled to pass his classes and make enough money to support his wife and two young daughters. In 1990, he landed a choice on-campus job as a proctor in the computer lab in the LC Smith College of Engineering. Even though he was struggling with subjects in the electrical engineering program, he was always there at the help desk. It was a daunting experience for Rory because his job was only to help his fellow students with computer lab *equipment* questions, but he invariably found his classmates asking deeper and harder questions: *"Dude, did you understand the calculus assignment? Can you help me?!"*

These students assumed that, because Rory was the proctor, he knew the answers. Afraid and full of self-doubt, he sought a way to help them without revealing his inadequacies. Rory learned to start with: *"Let's go back to the basics. Remember that last week the professor presented us with an equation. . .?"* By going back to the fundamentals, restating them, and rebranding them, Rory began to develop a technique that would, more often than not, lead to working solutions. By the time his senior year rolled around, there was often a line of students waiting at the help desk on the nights Rory worked.

Fast-Forward 17 Years

Picture a long-haired, wacky professor walking through the campus of the University of Colorado at Colorado Springs, dressed in a stunning contrast of old-school and drop-out. As he walks into the Engineering Building, he is greeted by students and faculty who smile and say hearty hellos, all the while probably shaking their heads at his tweed jacket, Grateful Dead t-shirt, khaki pants, and flip flops. As he walks down the hall of the Computer Science Department, there's a line of students standing outside his office. Reminiscent of the line of students that waited for him at the help desk in those early years as a proctor in the computer lab, they turn and greet him, *"Good morning, Dr. Lewis!"* Many of these students at UC Colorado Springs are not even in his class, but they know that Dr. Lewis will see them and help them anyway.

Past—Present—Future

Dr. Lewis holds three academic degrees. He earned a Bachelor of Science in Computer Engineering from Syracuse University. Syracuse's LC Smith College of Engineering is one of the country's top schools. It is there that Intel, AMD, and Microsoft send their top employees to study for their PhDs.

Upon completing his BS (with emphasis on the mathematics of electronic circuitry in microprocessors), he went across the quad to the Syracuse University School of Law. During his first summer at law school, Fulbright & Jaworski, the nation's most prolific law firm, recruited Rory to work in its Austin office, where some of the attorneys specialize in high-tech intellectual-property patent litigation. As part of his clerking experience, Lewis worked on the infamous *AMD v. Intel* case; he helped assess the algorithms of the mathematics of microprocessor electrical circuitry for the senior partners.

During his second summer in law school, Skjerven, Morrill, MacPherson, Franklin & Friel—the other firm sharing the work on the *AMD v. Intel* case—recruited Rory to work with them at their Silicon Valley branches (San Jose and San Francisco). After immersing himself in law for several years and receiving his JD at Syracuse, Lewis realized his passion was for the *mathematics* of computers, not the legal ramifications of hardware and software. He preferred a learning and creative environment rather than the fighting and arguing intrinsic in law.

After three years away from academia, Rory moved south to pursue his PhD in Computer Science at the University of North Carolina at Charlotte. There, he studied under Dr. Zbigniew W. Ras, known worldwide for his innovations in data-mining algorithms and methods, distributed data mining, ontologies, and multimedia databases. While studying for his PhD, Lewis taught computer science courses to computer engineering undergraduates, as well as e-commerce and programming courses to MBA students.

Upon receiving his PhD in Computer Science, Rory accepted a tenure-track position in Computer Science at the University of Colorado at Colorado Springs, where his research is in the computational mathematics of neurosciences. Most recently, he co-wrote a grant proposal on the mathematical analysis of the genesis of epilepsy with respect to the hypothalamus. However, with the advent of Apple's revolutionary iPhone and its uniquely flexible platform—*and market*—for mini-applications, games, and personal-computing tools, he grew excited and began experimenting and programming for his own pleasure. Once his own fluency was established, Lewis figured he could teach a class on iPhone apps that would include *non*-engineers. With his insider knowledge as an iPhone beta tester, he began to integrate the parameters of the proposed iPad platform into his lesson plans—even before the official release in April 2010.

The class was a resounding success, and the feedback was overwhelmingly positive, from students and colleagues alike. When approached about the prospect of converting his course into a book to be published by Apress, Dr. Lewis jumped at the opportunity. He happily accepted an offer to convert his course outlines, class notes, and videos into the book you are now holding in your hands.

Why Write This Book?

The reasons Dr. Lewis wrote this book are the same reasons he originally decided to create a class for both engineering and non-engineering majors: the challenge and the fun! According to Lewis, the iPhone and iPad are "*…some of the coolest, most powerful, and most technologically advanced tools ever made—period!*"

He is fascinated by the fact that, just under the appealing touchscreen of high-resolution images and fun little icons, the iPhone and iPad are programmed in Objective-C, an incredibly difficult and advanced language. More and more, Lewis was approached by students and colleagues who wanted to program apps for the iPhone and would ask his opinion on their ideas. It seemed that, with every new update of the iPhone, not to mention the advent of the expanded interface of the iPad, the floodgates of interest in programming apps were thrown open wider and wider. Wonderful and innovative ideas just needed the proper channel to flow into the appropriate format and then out to the world.

Generally speaking, however, the people who write books about Objective-C write for people who know Java, C#, or C++ at an advanced level. So, because there seemed to be no help for the average person who, nevertheless, has a great idea for an iPhone/iPad app, Dr. Lewis decided to launch such a class. He realized it would be wise to use his own notes for the first half of the course and then explore the best existing resources he could find.

As he forged ahead with this plan, Dr. Lewis was most impressed with *Beginning iPhone 3 Development: Exploring the iPhone SDK*, the best-selling instructional book from Apress written by Dave Mark and Jeff Lamarche. Lewis concluded that their book would provide an excellent, high-level target for his lessons—a stepping-stones approach to comprehensive and fluent programming for all Apple's multitouch devices.

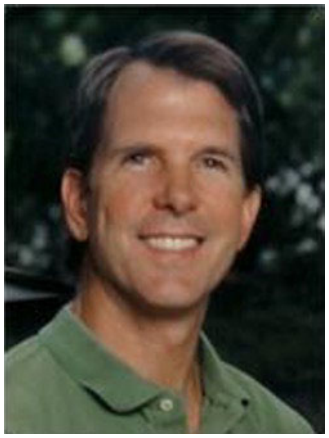
After Dr. Lewis's course had been successfully presented, and during a subsequent conversation with a representative from Apress, Lewis happened to mention that he'd only started using that book about halfway through the semester, as he had to bring his non-engineering students up to speed first. The editor suggested converting his notes and outlines into a primer—an introductory book tuned to the less-technical programming crowd. At that point, it was only a matter of time and details, like organizing and revising Dr. Lewis's popular instructional videos to make them available to non-engineers excited to program their own iPhone and/or iPad apps.

So, that's the story of how a wacky professor came to write this book. We hope you are inspired to take this home and begin. Arm yourself with this knowledge and begin now to change your life!

—Ben Easton
Author, teacher, editor

About the Contributing Authors

Chad Mello is currently working on his PhD in Computer Science at the University of Colorado at Colorado Springs, with Dr. Rory Lewis as his mentor. Several research projects in machine learning keep him busy, such as research into epilepsy identification and predication models. Possessing Top Secret clearance, he is currently a freelance software consultant and architect for the U.S. Air Force, Army, and a number of private companies. His specialties are software architectures and framework design as well as geospatial applications. Chad currently lives with his wife and three children in the Rocky Mountains and loves every minute of it.



Ben Easton is a graduate of Washington & Lee University and has a BA in Philosophy. His eclectic background includes music, banking, sailing, hang gliding, and retail. Most of his work has involved education in one form or another. Ben taught school for 17 years, mostly middle school mathematics. More recently, his experience as a software trainer and implementer reawakened his long-time affinity for technical subjects. As a freelance writer, he has written several science fiction stories and screenplays, as well as feature articles for magazines and newsletters. Ben resides in Austin, Texas and is currently working on his first novel.

Acknowledgments

When I arrived in America in 1981 at the age of 20, I had no experience or money and didn't even know how to use an American payphone. Since then it's been a wonderful road leading to this book and my life as an assistant professor at two University of Colorado campuses. I am such a lucky man to have met so many wonderful people.

First, to my wife, Kera, who moved mountains to help with graphics, meals, dictations, keeping me working, and sustaining a nominal level of sanity in our house. Thank you, Kera.

To my mother, Adeline, who was always there to encourage me, even in the darkest of times when I almost dropped out of electrical engineering. To my sister, Vivi, who keeps me grounded, and my late brother Murray, a constant reminder of how precious life is. To Keith and Nettie Lewis, who helped me figure out those American payphones. To Ben Easton, Brian Bucci, and Dennis Donahue, all of whom invited me into their families when I had nobody.

A special thanks to Dr. Zbigniew Ras, my PhD advisor, who became like a father to me, and to Dr. Terry Boult, my mentor and partner in the Bachelor of Innovation program at UCCS.

Last, but not least, to Clay Andres at Apress—he walked me through this process and risked his reputation by suggesting to a bunch of really intelligent people that I could author such a book as this.

Many thanks to you all.

Preface

What This Book Will Do For You

Let me get this straight: you want to learn how to program for the iPhone or the iPad and you consider yourself to be pretty intelligent—but whenever you read computer code or highly technical instructions, your brain seems to shut down. Do your eyes glaze over when reading gnarly instructions? Does a little voice in your head chide, *"How about that! Your brain shut down six lines ago, but you're still scanning the page—pretending you're not as dense as you feel. Great!"*

See if you can relate to this: you're having an issue with something pretty technical, so you decide to Google it and troubleshoot the problem. You open the top hit, and somebody else has asked the exact same question! You become excited as the page loads, but, alas, it's only a bulletin board (a chat site for all those geeks who yap at one another in unintelligible code). You see your question followed by . . . but it's too late! Your brain has already shut down, and you feel the tension and frustration as knots form in your belly.

Sound Familiar?

Yes? Then this book's for you! My guess is that you're probably standing in a bookstore or in the airport, checking out a magazine stand for something that might excite you. Because you're reading this in some such upscale place, you can probably afford an iPhone, a Mac, a car, and plane tickets. You're probably intrigued by the burgeoning industry of handhelds and the geometric rate at which memory and microprocessors are evolving . . . and how quickly ideas can be turned into startlingly new computing platforms, powerful software applications, helpful tools, and clever games—perhaps even into greenbacks! And now you're wondering if you can get in on the action, using your intellect and technical savvy to serve the masses.

How Do I Know This About You?

Easy! Through years of teaching students to program, I know that if you're still reading this, you're both intelligent enough and sufficiently driven to step onto the playing field of programming, especially for a device as sweet as the iPhone or as sexy as the iPad. If you identify with and feel connected to the person I've just described, then I know you. We were introduced long ago.

You're an intelligent person who may have mental spasms when reading complex code—even if you have some background in programming. And, even if you do have a pretty strong background in various programming languages, you are a person who simply wants an easy, on-point, no-frills strategy to learn how to program the iPhone and iPad. No problem! I can guide you through whatever psychological traffic jams you typically experience and help you navigate any technical obstacles, real or imagined. I've done this a thousand times with my students, and my methodology will work for you too.

The Approach I Take

I don't try and explain everything in minute detail. Nor do I expect you to know every line of code in your iPhone/iPad application at this stage. What I will do is show you, step by step, how to accomplish key actions. My approach is simultaneously comprehensive and easygoing, and I take pride in my ability to instruct students and interested learners along a wide spectrum of knowledge and skill sets.

Essentially, I will lead you, at your own pace, to a point where you can code, upload, and perhaps sell your first iPhone/iPad app, whether it is simple or complex. Good news: the most downloaded apps are *not* complex. The most popular ones are simple, common-sense tools for life . . . finding your car in a parking lot, making better grocery lists, or tracking your fitness progress. However, when you complete this book, you may want to graduate to other books in the Apress and friends of ED series. You have quite a few options here, and down the road I'll advise you regarding the best ways to move forward.

May you experience great joy and prosperity as you enter this amazing and magical world.

Peace!

—Rory A. Lewis, PhD, JD

Chapter 1

Preliminaries

This introductory chapter aims to make sure you have all the required tools and accessories to proceed with full confidence.

Three types of readers are likely to read this book. One group can skip to Chapter 2 immediately without reading Chapter 1. Another group may only need to read one small section in Chapter 1 before moving on to Chapter 2. The third group should read Chapter 1 very carefully before moving on.

- *Group 1*: You own a Mac. You have experience coding with Xcode on your Mac. You have an up-to-date iOS SDK and an up-to-date version of Xcode. You also have experience with DemoMonkey, and it's installed on your machine. If all this is true, meet me in Chapter 2.
- *Group 2*: You own a Mac. You have experience coding with Xcode on your Mac. You have an up-to-date iOS SDK and an up-to-date version of Xcode. But you have no experience with DemoMonkey or it's not installed on your machine. Please check out the section "Installing DemoMonkey" in this chapter and then meet me in Chapter 2.
- *Group 3*: You are a seeker of knowledge and have begun travelling down a wonderful road. We need to check your backpack and make sure you have all the tools you'll need for your journey. So let's start right here.

Necessities and Accessories

In order to program for the iPhone and/or iPad, and to follow along with the exercises, tutorials, and examples presented in this book, you'll need to meet six minimal requirements. You may not completely understand these requirements right now, but that's okay—just roll with me for a second, and I'll explain everything as we go through these steps.

> **Note** Whenever I say *iPhone* or *iPad*, I'm referring to any iPhone or iPad OS device, including the iPod touch. And when I say *Macintosh HD*, yours may be named something different.

Briefly, you'll need six things:

- An Intel-based Macintosh
- The correct operating system for your Mac (OS X 10.8.1 Mountain Lion or later)
- To be a registered developer or be simulator-based (discussed in detail later in this chapter)
- To have the correct operating system for your iPhone (iOS 6 or later)
- To have the correct Software Development Kit (SDK) for your iPhone that runs a program called Xcode (version 4.4 and later)
- To have installed DemoMonkey

Let's go into each of these in a bit more detail.

Getting a Mac

As far as your Mac is concerned, I'll explain what you need in two ways—first a geeky way and then a newbie way.

- *In geek terms*: You need a Mac that has a 64-bit Intel-Core 2 duo processor or better.
- *In newbie terms*: You need a MacBook Pro that was manufactured from mid-2009 or later, or an iMac that was manufactured from mid-2007.

All the videos on the net that accompany this book are screencast from my MacBook Pro, bought in 2011; if I broadcast from my 2010 iMac, I first run the app on my MacBook Pro, bought in 2011.

You don't need the latest revved-up Mac. If you haven't bought one yet, get a basic, no-frills MacBook Air. See Figure 1-1.

Figure 1-1. *The author uses the cheapest 2011 MacBook Pro on the market to perform all the coding and compiling in this book. Many of the authors' students purchase the MacBook Air for $999, as illustrated here*

Or buy a secondhand MacBook Pro on eBay or another such site. If you do that, or own an older Mac, you may be able to add some RAM. I suggest you make a free appointment at the nearest Apple Store's Genius Bar and ask whether they can increase the RAM on your older model Mac. If so, ask about the maximum the RAM can be increased. Then ask explicitly: "Can this old computer run Mountain Lion, *at least* 10.8.1, and Xcode 4.4.1 or later?"

If you don't have a Mac, you'll need to buy one if you want to follow along with this book or program Objective-C to create iPhone apps. Keep in mind that, as mentioned, I've made a point to code and run every program in this book on Apple's smallest and cheapest model, the MacBook Air—many of my newbie students buy the MacBook Air for $999.

Getting OS X

You'll need the correct version of OS X. At the time of this writing, that version is OS X 10.8.1. You need to have the latest greatest operating system inside your Mac. I see a lot of emails and forum questions revealing that many people think: "*Ah, my code probably didn't compile correctly because Dr. Lewis has a different version of OS X or/and iOS on his machine…*"

> **Note** Even if you think your system is up-to-date, I suggest you follow along in this section to make sure your system has the latest OS X and the latest iOS inside it. I say this because as you follow along in this book and tackle all the programs, there will be times when your code doesn't work the first time you run it.

To make sure your system is recent enough to follow along with the book, please do the following:

1. Close every program running on your Mac so that only the Finder is running.
2. Click the little apple in the upper left-hand corner of your screen and select About This Mac. You'll see the window shown in Figure 1-2. Make sure it says OS X 10.8.1 (or later).

Figure 1-2. *Here you can see that Dr. Lewis's iMac is using OS X 10.8.1. and Xcode 4.4.1*

Now make sure your Mac has the latest software:

1. With all your programs closed except for the Finder, click the apple in the upper left-hand corner again and select Software Update, as illustrated in top image in Figure 1-3.

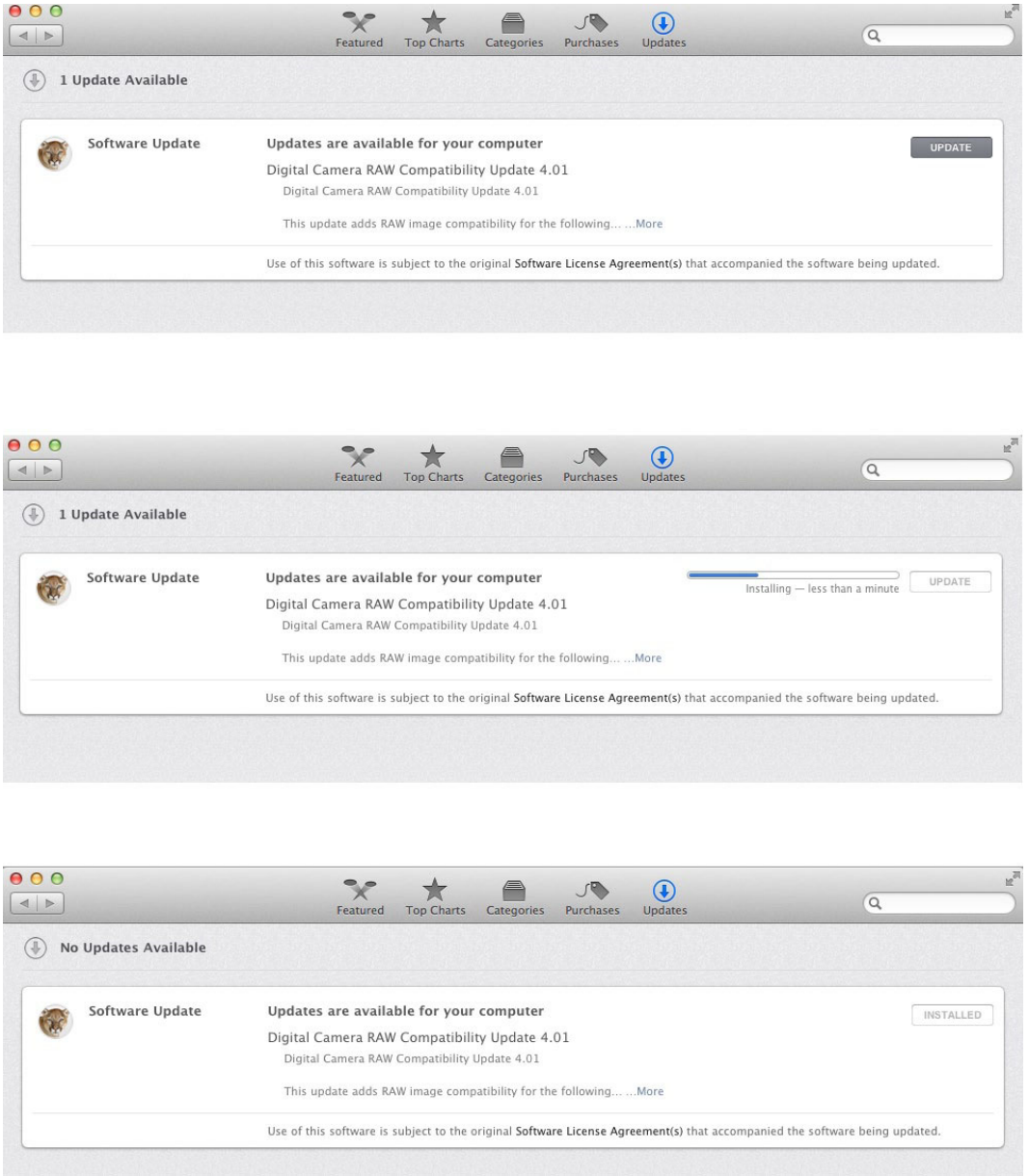


***Figure 1-3. Top:** Checking for new software. **Middle:** Download any new software and wait for software to download. **Bottom:** All software updated*

2. If updates are available, click Continue and follow the instructions and screen prompts, as shown in two bottom figures of the three Figures 1-3.

If your version of OS X or iOS makes my pictures seemed dated, don't freak out. Along with volunteers, I maintain a forum that has news about recent updates of OS X and iOS. You can visit the forum at www.rorylewis.com/ipad_forum/.

Becoming a Developer

If you're not already one, you need to become a registered developer via the iPhone/iPad DK for $99. Or you can get an introductory set of bells and whistles for free.

Making Your Choice

If you're a student, your professor has probably already taken care of this, and you may already be registered under your professor's name. If you're not a student, you need to decide which type of developer you'd like to be. Here are your options:

- *$0 option*: You can go to the App Store and download Xcode for free. That's fine, but bear in mind that unless you become a developer ($99), you'll only be able to see the apps you code and program in this book running on the iPhone or iPad *Simulator*. You won't be able to run them on a real physical iPad or iPhone. And you won't be able to sell your apps on the iTunes store. Nor will you be able to log in to the developer site to view code snippets and updates, beta-test new products, or be a part of the Apple online community. Still, the free route may be a very good choice for people who aren't sure whether they want to continue with Xcode and programming. If that's the case, then download the latest version of Xcode from https://developer.apple.com/xcode/ and meet me at Figure 1-10.
- *$99 option*: If you do want to run your apps on a real iPad or iPhone, sell apps on the iTunes store, and be a part of the developer group at Apple, simply continue reading.

Installing Xcode

Let's get started installing Xcode.

1. Go to http://developer.apple.com/programs/ios/. You'll see a page similar to the one shown in Figure 1-4. Click the Enroll Now button.

Figure 1-4. Click the Enroll Now button

2. Click Continue, as illustrated in Figure 1-5.

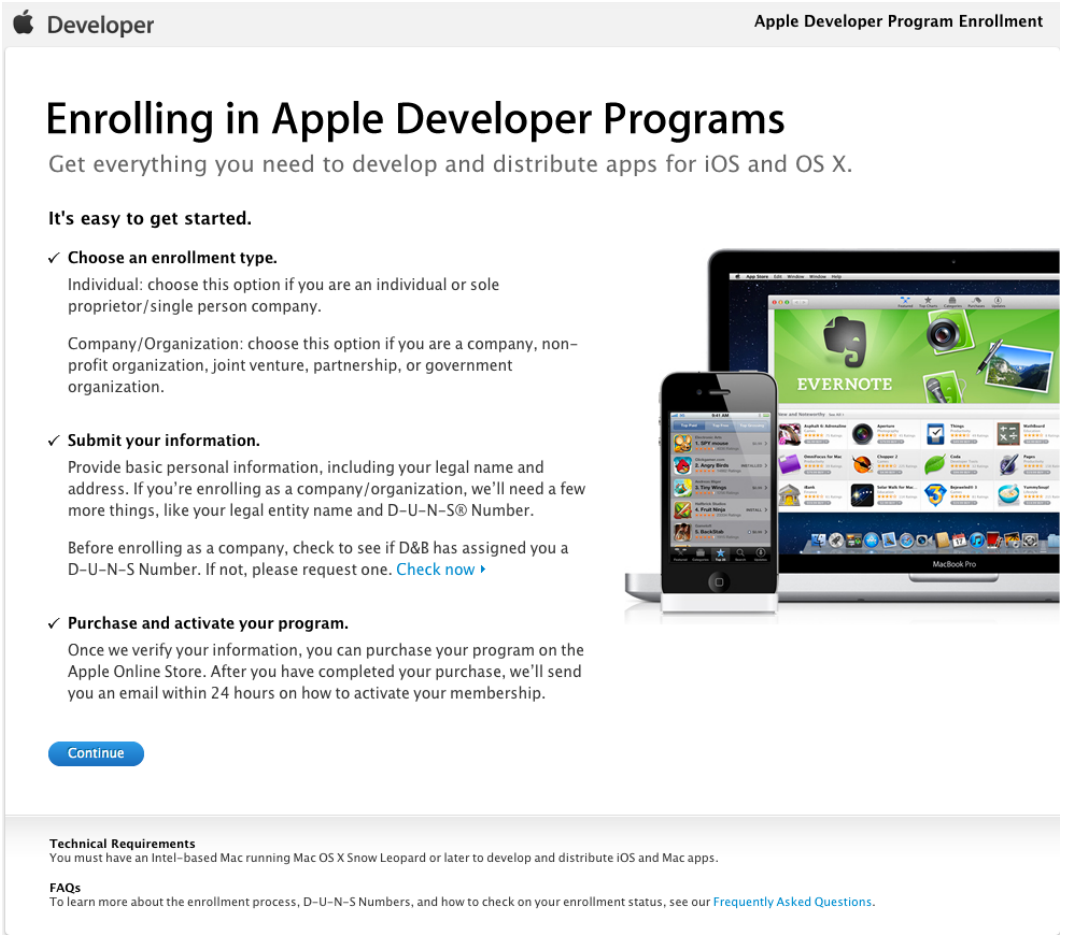


Figure 1-5. *Click the Continue button*

3. Most people reading this book will select the "Create an Apple ID" option (arrow 1 in Figure 1-6) and then click Continue (arrow 2). If you already have an account, then you've been through this process before; go ahead with the process beginning with the "Sign in with your Apple ID" option, and I'll meet you at step 6, where you'll log in to the iPhone/iPad development page and download the SDK.

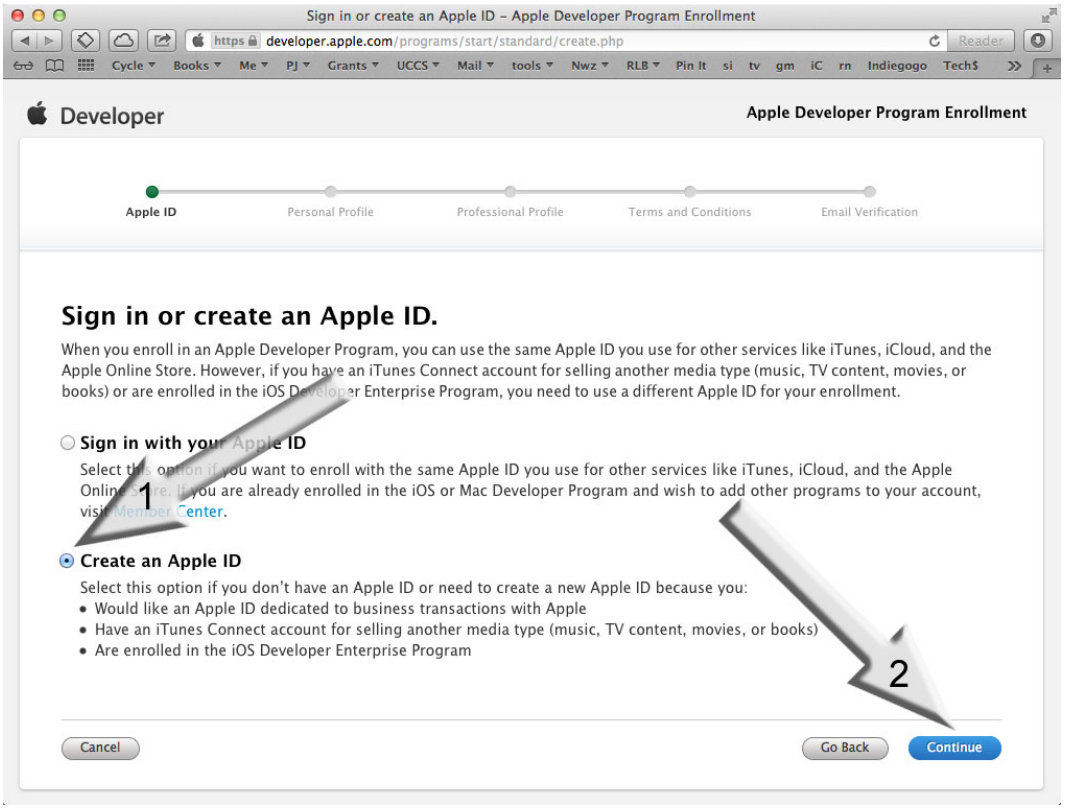


Figure 1-6. *Click the "Create an Apple ID" option to proceed*

4. You're probably going to be enrolling as an individual, so click the Individual button shown in Figure 1-7. If you're enrolling as a company, click the Company button to the right, follow the appropriate steps, and then skip to step 6.

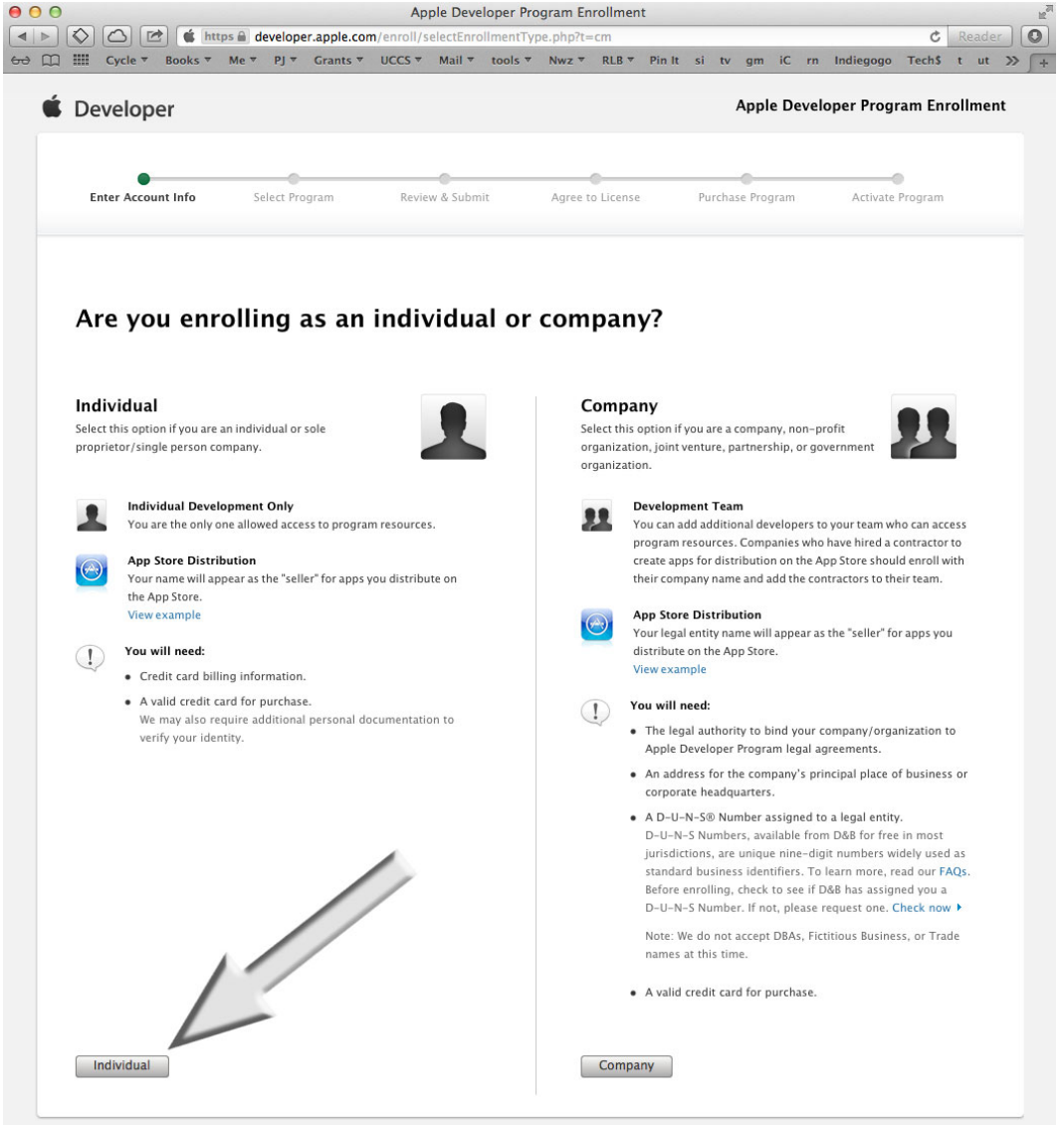


Figure 1-7. *Click the Individual option*

5. Enter all your information as shown in Figure 1-8 and pay the Standard program's $99 fee. The fee provides all the tools, resources, and technical support you'll need. (If you're reading this book, you don't buy the Enterprise program at $299—it's for commercial in-house applications.) After paying, save your Apple ID and username; you'll get a confirmation email.

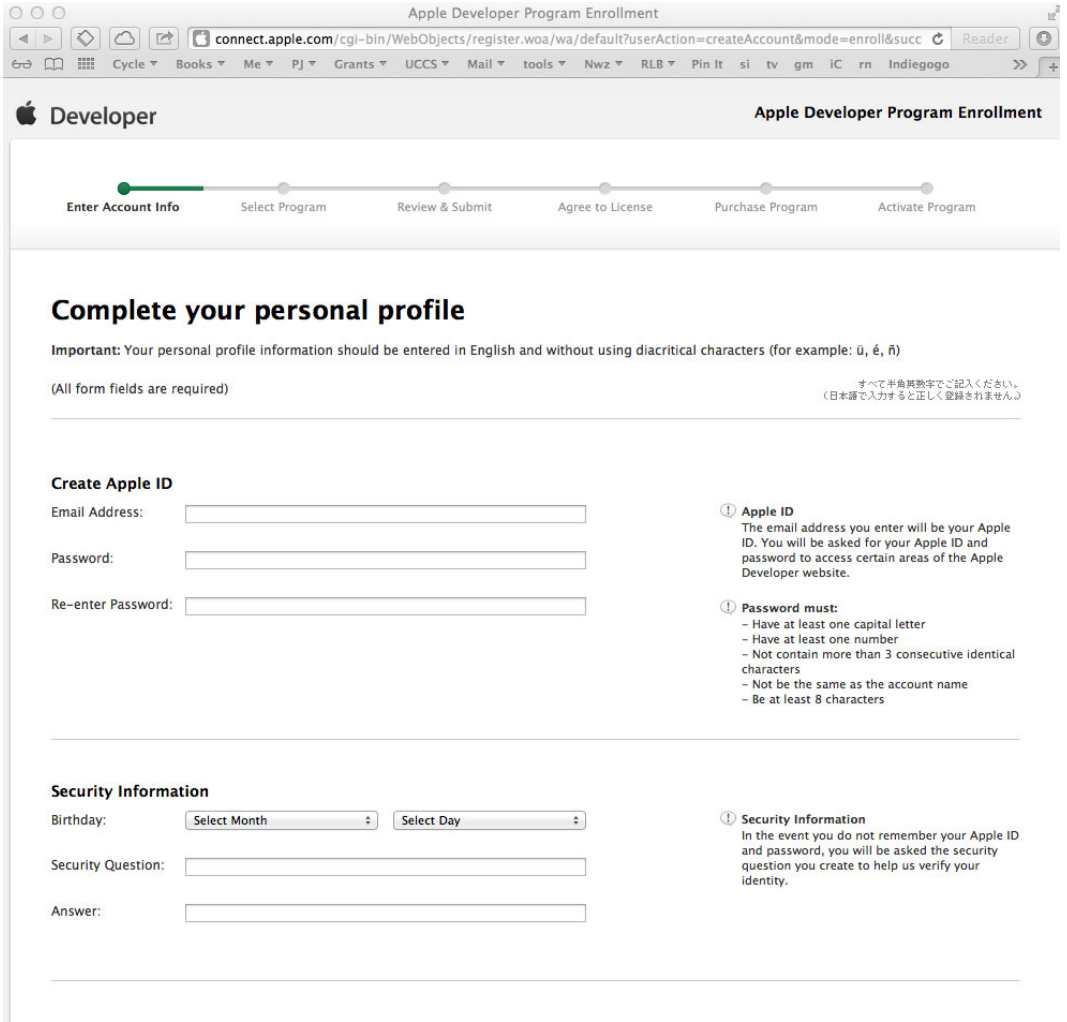


Figure 1-8. *Enter all your information accordingly*

> **Note** Before you move on to step 6, make sure you have received your confirmation email and chosen a password to complete the last step of getting set up as a bona fide registered Apple developer. Congratulations!

6. Use your Apple ID to log in to the main iOS development page at `http://developer.apple.com`. This page has three icons for the three types of Apple programmers. As shown on Figure 1-9, click the iOS Dev Center icon, which leads to the download page for iOS development software.

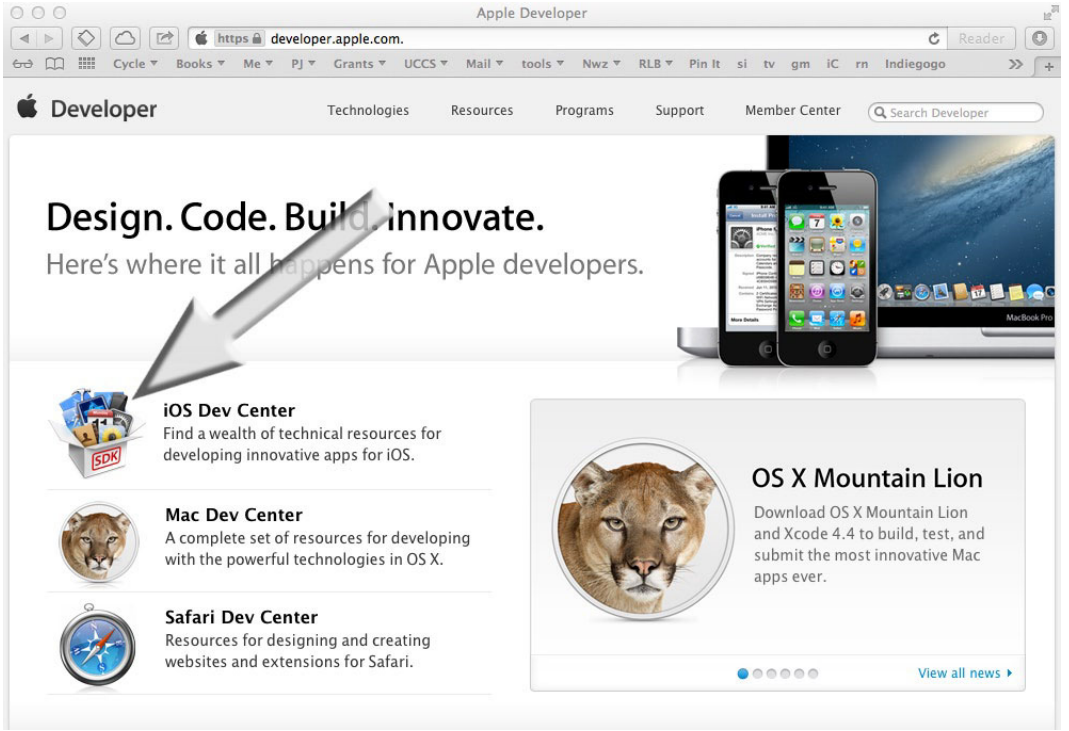


Figure 1-9. *For now click the iOS Dev Center icon as indicated by the arrow. Later you may want to program applications for Mac computers or the Safari web browser*

7. After logging in with your username and password as described in step 6, you'll see a screen similar to Figure 1-10. The iOS Dev Center contains all the tools necessary to build iOS apps. You'll spend more time here later, but for now just go to the Developer Page of the latest build of the iOS SDK by clicking the icon indicated by the arrow.

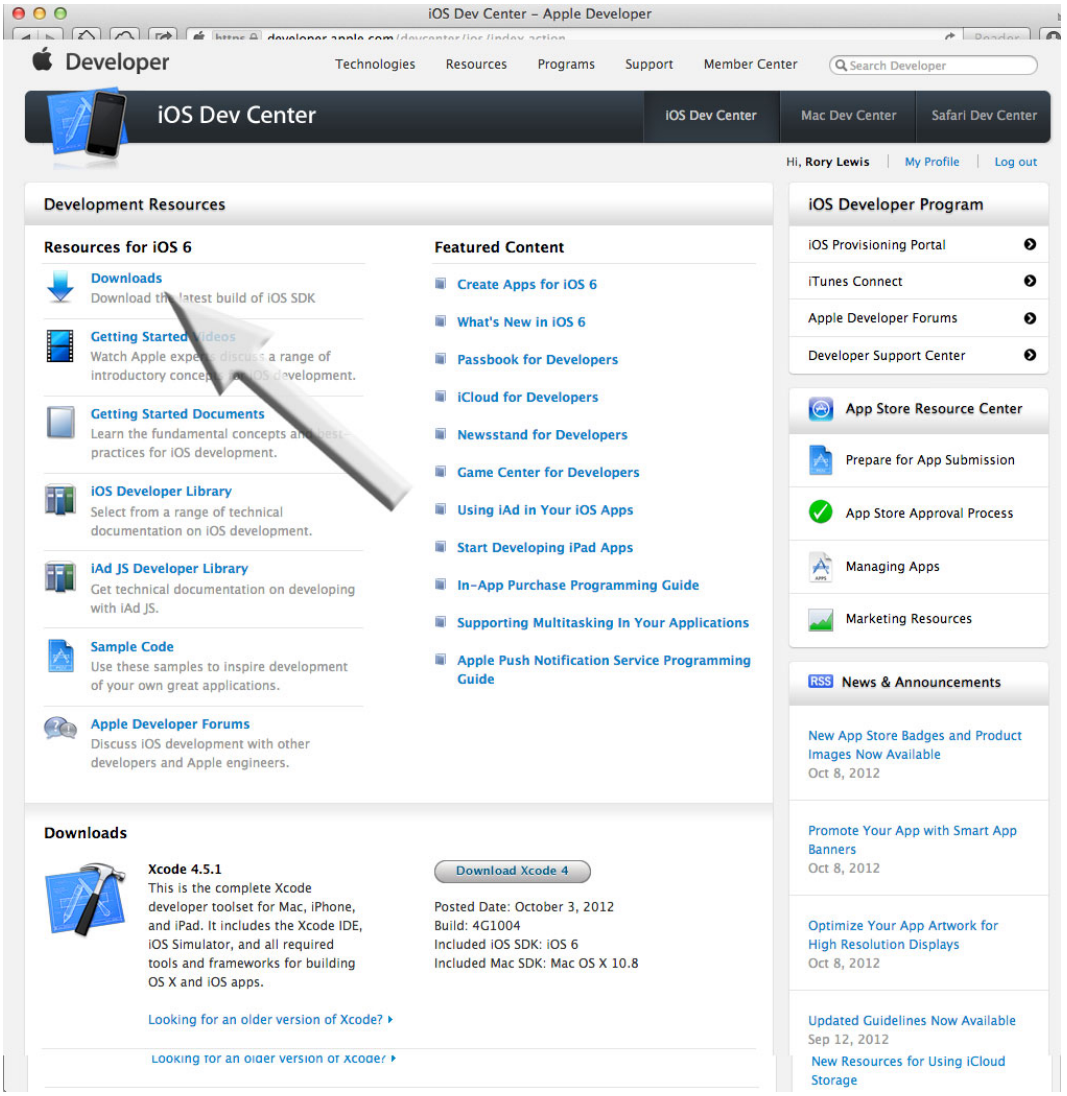


Figure 1-10. *The Downloads link takes you to the bottom of the page*

> **Note** At the time of writing, Xcode 4.4 and iOS SDK 6 are the latest environments. By the time you read this book these may well have larger numbers. That's not a problem—just go on to step 8. If by chance something has really thrown us a curve ball, it will be discussed and solved for you in our forum located at `www.rorylewis.com/ipad_forum/` or `http://bit.ly/oLVwpY`.