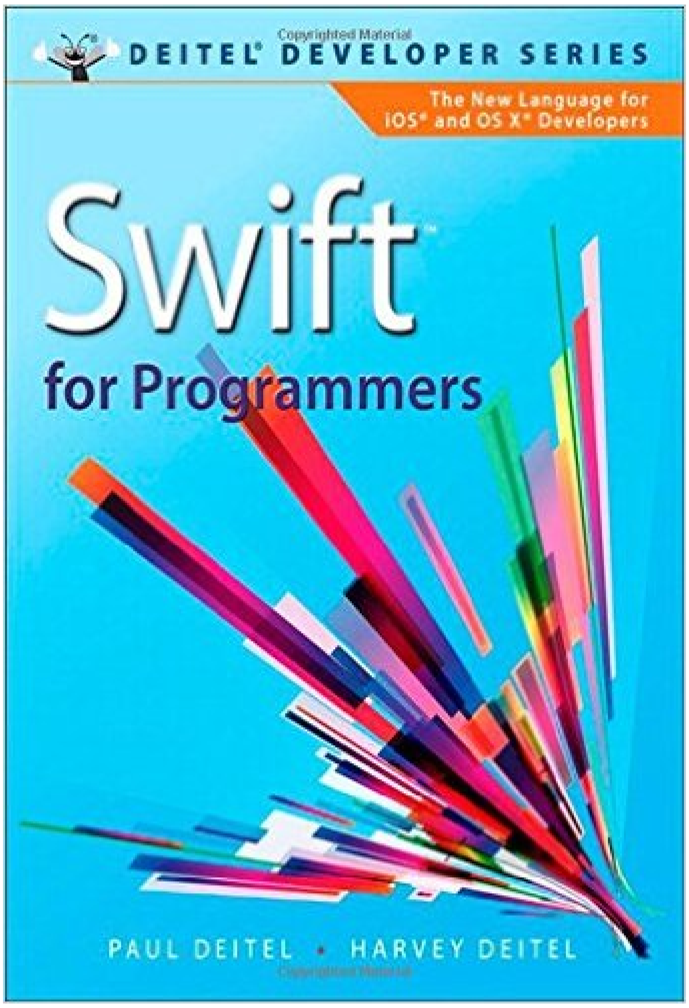
The book was found

Swift For Programmers (Deitel Developer Series)



PDF
Adobe
DOWNLOAD EBOOK

Synopsis

The professional programmerâ ™s DeitelÂ® guide to Appleâ ™s new Swift programming language for the iOSÂ® and OS XÂ® platforms Â¿ Written for programmers with a background in object-oriented programming in a C-based language like Objective-C, Java, C# or C++, this book applies the Deitel signature live-code approach with scores of complete, working, real-world programs to explore the new Swift language in depth. The code examples feature syntax shading, code highlighting, rich commenting, line-by-line code walkthroughs and live program outputs. The book features thousands of lines of proven Swift code, and tips that will help you build robust applications. Â¿ Start with an introduction to Swift using an early classes and objects approach, then rapidly move on to more advanced topics. When you master the material, youâ ™ll be ready to build industrial-strength object-oriented Swift applications.   About This Book Â¿ The Swiftâ„¢ programming language was arguably the most significant announcement at Appleâ ™s 2014 Worldwide Developers Conference. Although apps can still be developed in Objective-CÂ®, Apple says that Swift is its applications programming and systems programming language of the future. Â¿ Swift is a contemporary language with simpler syntax than Objective-C. Because Swift is new, its designers were able to include popular programming language features from languages such as Objective-C, Javaâ„¢, C#, Ruby, PythonÂ® and many others. These features include automatic reference counting (ARC), type inference, optionals, String interpolation, tuples, closures (lambdas), extensions, generics, operator overloading, functions with multiple return values, switch statement enhancements and more. Weâ ™ve been able to develop apps more quickly in Swift than with Objective-C and the code is shorter, clearer and runs faster on todayâ ™s multi-core architectures. Â¿ Swift also eliminates the possibility of many errors common in other languages, making your code more robust and secure. Some of these error-prevention features include no implicit conversions, ARC, no pointers, required braces around every control statementâ ™s body, assignment operators that do not return values, requiring initialization of all variables and constants before theyâ ™re used, array bounds checking, automatic checking for overflow of integer calculations, and more. You can combine Swift and Objective-C in the same app to enhance existing Objective-C apps without having to rewrite all the code. Your apps will easily be able to interact with the CocoaÂ®/Cocoa TouchÂ® frameworks, which are largely written in Objective-C. Â¿ You can also use the new Xcode playgrounds with Swift. A playground is an Xcode window in which you can enter Swift code that compiles and executes as you type it. This allows you to see and hear your codeâ ™s results as you write it, quickly find and fix errors, and conveniently experiment with features of Swift and the Cocoa/Cocoa Touch frameworks. Â¿ Practical, Example-Rich Coverage of: Classes, Objects, Methods, Properties Initializers, Deinitializers, Bridging Tuples, Array and Dictionary Collections Structures, Enumerations, Closures, ARC Inheritance, Polymorphism, Protocols Type Methods, Type Properties Generics; Strings and Characters Operator Overloading, Operator Functions, Custom Operators, Subscripts Access Control; Type Casting and Checking Nested Types, Nested Methods Optionals, Optional Chaining, Extensions Xcode, Playgrounds, Intro to Cocoa TouchÂ® with a Fully Coded iOSÂ® 8 Tip Calculator App Overflow Operators, Attributes, Patterns More topics online Â¿ IMPORTANT NOTE ABOUT XCODE AND SWIFT: With Xcode 6.3 and Swift 1.2, Apple introduced several changes in Swift that affect the book&#39;s source code. Please visit www.deitel.com/books/iOS8FP1 for updated source code. The changes do not affect Xcode 6.2 users. You can download Xcode 6.2 from developer.apple.com/downloads/index.action (youâ ™ll have to log in with your Apple developer account to see the list of downloads). Â¿Visit www.deitel.com Download code examples For information on Deitelâ ™s Dive IntoÂ® Series programming training courses delivered at organizations worldwide visit www.deitel.com/training or to deitel@deitel.com Join the Deitel social networking communities on FacebookÂ® at facebook.com/DeitelFan, TwitterÂ® at @deitel, Google+â„¢ at google.com/+DeitelFan, LinkedInÂ® at bit.ly/DeitelLinkedIn, YouTubeâ„¢ at youtube.com/user/DeitelTV and subscribe to the DeitelÂ® Buzz Online e-mail newsletter at www.deitel.com/newsletter/ subscribe.html Â¿

Book Information

Series: Deitel Developer Series

Paperback: 400 pages

Publisher: Prentice Hall; 1 edition (February 1, 2015)

Language: English

ISBN-10: 0134021363

ISBN-13: 978-0134021362

Product Dimensions: 6.9 x 1.1 x 9 inches

Shipping Weight: 1.5 pounds (View shipping rates and policies)

Average Customer Review: 4.1 out of 5 starsÂ Â See all reviewsÂ (19 customer reviews)

Best Sellers Rank: #841,612 in Books (See Top 100 in Books)   #100 inÂ Books > Computers & Technology > Programming > Languages & Tools > Swift   #512 inÂ Books > Computers & Technology > Mobile Phones, Tablets & E-Readers > Programming & App Development   #1061 inÂ Books > Computers & Technology > Programming > Software Design, Testing & Engineering >

Customer Reviews

Having read all fourteen chapters in this book, I now feel competent in this new language that is very different from all other object-oriented compilable languages I know. I have worked extensively in Java, C#, C++, and Objective-C. Swift seems to be a distant relative to all those languages. In fact, Swift seems to be so distinct as to be in a language family of its own. So there is a greater learning curve to mastering this language coming from those languages than there is in going from one to another among those languages, such as from Java to C++. However, this book is titled correctly: "Swift for Programmers". It assumes you are already a programmer in one of those languages, and thus you do not start from programming 101. Instead, this book gets you up-to-speed fast, and is quick paced, giving just enough coverage to enable comprehension of a language feature before moving onward.The concepts covered are clear and well written. The book as a whole is well written, and can easily be read from cover to cover. The last two chapters are a quick intro to iOS programming with Swift, but the rest of the book applies to development in the language itself, whether the target is OS X or iOS. The book is a quick read but comprehensive, and I highly recommend this book for programmers who want to get into this new technology coming from another technology, whether it is from iOS in Objective-C or from C++,Java, or C#.Once you master this language, if you knew Objective-C, you will understand why Apple is moving over to this new technology for all their development. Swift is better, more powerful, and faster to develop in than Objective-C and you can directly access libraries written in Objective-C and C from your Swift code. However, there is no equivalent of Objective-C++ in Swift yet, but the technology is quite new, and we don’t know how it will change in coming years. But the book makes clear that rapid changes are occurring in the language. So what you learn today may be different in the soon future. However, now is the time to come onboard with Swift, in its infancy, so you can ride the wave and have the advantage over those still working in Objective-C.There are a lot of unusual features in Swift that require an in-depth explanation to understand, and this book provides that. For example, the book covers the automatic conversion of common types between Swift and Objective-C libraries that it uses. It mentions how NSString, NSDictionary, and NSArray are converted to Swift’s value based struct equivalents transparently to the programmer. It also mentions the memory management of Swift. Swift does not have automatic garbage collection like Java or C#, but neither does it have raw pointers like C++, C, and Objective-C. Instead, it accomplishes memory management through ARC (Automatic Reference Counting), which is a feature started in

Objective-C and perfected in Swift. Basically, the Swift compiler handles all the ARC code; all the programmer needs to do is prevent circular references, which is done by using weak references. ARC is used for reference types only. Everything else is passed by value on the stack, and these are called structs. Even numbers are value objects with methods and properties.Swift is an object-oriented language without a common super object from which all classes descend. Swift classes do not need to have a super class. However, your Swift classes can descend from Objective-C classes and inherit and change their methods. This book covers these object-oriented features, including extensions, which is the Swift version of Objective-C categories, only much more powerful. With extensions and operator overloading, the book shows how you can add overloaded operators to existing Objective-C classes in Foundation and Cocoa and Cocoa-Touch without access to the source code of those classes! C++ and C# are not quite as capable in their operator overloading functionality.This book also covers how they totally avoid the problems of Null Pointer Exceptions, which is a problem with most other compilable languages. This is accomplished through optionals. Note that there are in fact no such thing as Exceptions in Swift. Instead you have failable initializers, which return optionals containing nil if they fail.These details just cover the tip of the iceberg in this new language. I strongly recommend buying this book while it is new. I bought it the day it was released, and I have never regretted it. It was a good read and serves as a good reference with good pointers to go back to. In addition, almost every section points to URLs that you find more detailed discussions on the internet for further understanding. In summary, this is a very good book.

I really didnÃ¢Â Â™t like this book. Because of the title, I assumed that this book would get me up and coding in no time. Instead, most of the book is dedicated to code snippets and short examples introducing the language. If this book is really for programmers, why are you explaining with paragraphs and code examples how the Ã¢Â ÂœorÃ¢Â Â• operator works. I already know how to program, and I already know all this stuff. It isnÃ¢Â Â™t until Chapter 13 that the book starts explaining stuff that a programmer new to the language wouldnÃ¢Â Â™t know. So I do not recommend this book. A few hours online looking at documentation would give you the same amount of knowledge.

Buy an expensive computer language book with a copyright the same year you bought it (2015) you would expect it to be completely relevant. A lot of the code will not compile with the latest Swift because of small syntax changes.Just to list a few things wrong that prevent compiling:1. "println()"

used in almost every example needs to be changed to "print()"2. "#" prefix for givng an external parameter name the same name as the internal has been removed from the language.3. "Printable" protocol has been replaced "CustomStringConvertible" . This is the way to make the object print themselves.The book is pretty good otherwise. If Deitel updates the book and Apple stop breaking compatibility with syntax changes it probably would be 4 or five star book. But you can start to see why the internet is replacing computer books with this example.

"Swift for Programmers" is a great resource for anyone that is currently trying to pick up Swift from another C based language. I’m coming from a Java/C# background and was able to pick up the basics of the language pretty easily using the book. I feel like I will be going back to this book often if I have questions or issues with any code I’m writing. I recommend downloading the code examples from the website and going through Xcode as the book is explaining the new material.I also have the book "iOS8 for Programmers An App-Driven Approach with Swift" which shares a little content between the books. The first chapters seem to be identical, and "Swift for Programmers" last two chapters are projects from this book.Besides the shared content, both books are great and recommended for anyone trying to pick up iOS or MacOS programming.

I just finished the book and I regret I didn’t find it sooner. It was the fourth book on swift I purchased. I have been a dyslexic programmer for many years and usually shied away from reading, but I found swift way too structured to just look at code and to see how it works. I taught myself VB.net and php just by looking up examples. I now have a good foundation to go from here.

Really well written, and it’s very easy to understand. I would’ve preferred more chapters written on apps and how swift integrates with Xcode but it’s understandable. I loved the Computer Engineering tips, they were so helpful in each section.Now I wouldn’t recommend this for beginners of programming but if you know a language already this is a great book

Download to continue reading...

Swift for Programmers (Deitel Developer Series) Android 6 for Programmers: An App-Driven Approach (3rd Edition) (Deitel Developer Series) Taylor Swift: The Ultimate Fan Book 2015: Taylor Swift Facts, Quotes and Quiz (Taylor Swift Fan Books) (Volume 2) Taylor Swift: The Ultimate Fan Book 2015: Taylor Swift Facts, Quiz and Quotes (Taylor Swift Fan Books) The iOS 5 Developer’s Cookbook: Core Concepts and Essential Recipes for iOS Programmers (3rd Edition) (Developer’s Library) An Illustrated Guide for z/Architecture Assembler Programmers: A compact reference for application programmers Delphi Nuts & Bolts for Experienced Programmers: For Experienced Programmers iOS Animations by Tutorials Second Edition: Updated for Swift 2.2: iOS 9 and Swift 2.2 Edition iOS 10 Programming Fundamentals with Swift: Swift, Xcode, and Cocoa Basics Swift Walker: A Space Adventure: Swift Walker Science and Geography Books for Kids, Book 3 Programming Groovy: Dynamic Productivity for the Java Developer (Pragmatic Programmers) Programming Groovy 2: Dynamic Productivity for the Java Developer (Pragmatic Programmers) QuickTime for Java: A Developer Reference (QuickTime Developer Series) Delphi 6 Developer’s Guide (Sams Developer’s Guides) Delphi 5 Developer’s Guide (Developer’s Guide) Delphi 4 Developer’s Guide with CDROM (Sams Developer’s Guides) Delphi Developer’s Guide to XML (Wordware Delphi Developer’s Library) Client/Server Developer’s Guide with Delphi 3 with CDROM (Sams Developer’s Guides) Java for the Web with Servlets, JSP, and EJB: A Developer’s Guide to J2EE Solutions: A Developer’s Guide to Scalable Solutions Book 1: Diving In - iOS App Development for Non-Programmers Series: The Series on How to Create iPhone & iPad Apps

Dmca