

Swift™ for Programmers

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 Touch® with a Fully Coded iOS® 8 Tip Calculator App
- Overflow Operators, Attributes, Patterns
- More topics online

Visit www.deitel.com

- Download code examples
- For information on Deitel's Dive Into Series programming training courses delivered at organizations worldwide visit www.deitel.com/training or write to deitel@deitel.com
- Join the Deitel social networking communities on **Facebook**® at facebook.com/DeitelFan, **Twitter**® at @deitel, **Google+**™ at google.com/+DeitelFan, **LinkedIn**® at bit.ly/DeitelLinkedIn, **YouTube**™ at youtube.com/user/DeitelTV and subscribe to the **Deitel**® Buzz Online e-mail newsletter at www.deitel.com/newsletter/subscribe.html

informit.com/ph

\$39.99 US \$45.99 CANADA

PRENTICE HALL

The professional programmer's Deitel® guide to Apple's new Swift programming language for the iOS® and OS X® platforms

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.

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, you'll be ready to build industrial-strength object-oriented Swift applications.

Paul Deitel and **Harvey Deitel** are the founders of **Deitel & Associates, Inc.**, the internationally recognized programming languages authoring and corporate-training organization. Millions of people worldwide have used Deitel books, LiveLessons video training and online resource centers to master iOS® app development in Swift and Objective-C, and Java™, C++, Android™, C#, .NET, Visual Basic®, Visual C++®, C, Internet and web programming, JavaScript®, HTML, CSS, XML, Python®, PHP and more.

COMMENTS FROM THE REVIEWERS

"Apple took everyone by surprise when they announced a new programming language for developing Mac and iOS applications. Taking lessons from Objective-C and many other languages, Apple built a new language from the ground up. There is a lot to learn — new syntax, new idioms and more. It all seems daunting, but the Deitels have written a book that thoroughly explores Swift and Xcode 6 and guides you through what you need to know, regardless of which language you came from."

— **Robert McGovern, Independent Developer**

"An excellent introduction to Apple's new programming language. Line-by-line code explanations. Practical real-world abstractions throughout the code. Full of links to great resources. Features are introduced by comparison to established programming concepts making Swift easy to learn for developers new to Apple's platforms. A must-read."

— **René Cacheaux, iOS Architect, Mutual Mobile**

"It's surprising that a book of this quality, depth and breadth has appeared so soon after Swift was announced. The ideal accompaniment to Apple's reference documentation. This developers' book takes an in-depth look at Swift. Whether you're moving to the Apple ecosystem from a C++, C# or Java background or you're an Objective-C programmer looking to update your skills to this newest and most exciting of Apple's languages, this book is for you. Complements the Deitels' excellent book *iOS 8 for Programmers: An App-Driven Approach with Swift* and maintains their trademark high-quality approach, containing many interactive, nontrivial code examples with in-depth code walkthroughs and best practices. Uses the power of Swift with Cocoa's Foundation classes. A must-have for any serious Apple developer."

— **Rik Watson, Technical Team Lead for HP Enterprise Services (Applications Services)**

"Perfect for the Objective-C developer looking to quickly learn Apple's newest language. You'll learn how to incorporate new Swift features such as tuples, closures and generics into your existing Objective-C projects. You'll appreciate Swift's built-in error handling while working through real-world examples in Xcode playgrounds."

— **Scott Bossak, Lead iOS Developer, Thrillist Media Group**

(Continued Inside Back Cover)

ISBN-13: 978-0-13-402136-2
ISBN-10: 0-13-402136-3



DEITEL®
DEVELOPER
SERIES

Swift™
for Programmers

DEITEL
DEITEL

PRENTICE
HALL

DEITEL® DEVELOPER SERIES

The New Language for
iOS® and OS X® Developers

Swift™ for Programmers

PAUL DEITEL • HARVEY DEITEL

DEITEL® DEVELOPER SERIES

The DEITEL® DEVELOPER SERIES is designed for professional programmers. The series presents focused treatments on a growing list of emerging and mature technologies, including Swift™, iOS® and Android™, Java™, C++, C, C# and .NET, JavaScript™, Internet and web development and more. Each book in the series contains the same live-code teaching methodology used in the Deitel's HOW TO PROGRAM SERIES college textbooks—most concepts are presented in the context of completely coded, working apps with sample live executions.

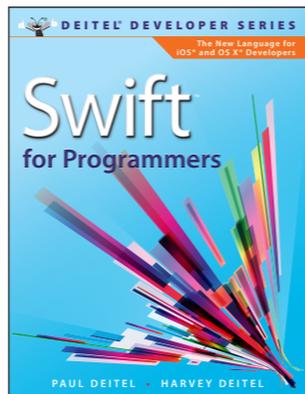
ABOUT THIS BOOK

The Swift™ programming language was arguably the most significant announcement at Apple's 2014 Worldwide Developers Conference. Although apps can still be developed in Objective-C®, Apple says that Swift is its applications programming and systems programming language of the future.

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, Java™, C#, Ruby, Python® 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. We've been able to develop apps more quickly in Swift than with Objective-C and the code is shorter, clearer and runs faster on today's multi-core architectures.

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 statement's body, assignment operators that do not return values, requiring initialization of all variables and constants before they'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 Cocoa®/Cocoa Touch® frameworks, which are largely written in Objective-C.

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 code's results as you write it, quickly find and fix errors, and conveniently experiment with features of Swift and the Cocoa/Cocoa Touch frameworks.



Cover graphic ©echo30051/Shutterstock



DEITEL & ASSOCIATES, INC.

Deitel & Associates, Inc., is an internationally recognized authoring and corporate training organization specializing in programming languages, object technology, Internet and web software technology, iOS® and Android™ app development, and Apple's new Swift language. The company offers instructor-led courses at client sites worldwide. The founders of Deitel & Associates, Inc., are Paul Deitel and Dr. Harvey Deitel. The company's training clients include many of the world's largest companies, government agencies, branches of the military and academic institutions. You can reach the authors at deitel@deitel.com.

COMMENTS FROM THE REVIEWERS *(Continued from Back Cover)*

"The chapters are comprehensive, covering simple use cases to complex challenges Swift is distinctly suited for. The code examples often represent day-to-day programming challenges. A wonderful learning tool and a handy reference for experienced developers."

— Ash Furrow, iOS Developer, Artsy

"With Swift-based iOS 8 and OS X development the Deitel magic continues. They guide you through Swift with increasingly complex projects. They also offer valuable software engineering tips, performance improvements and techniques for preventing common errors. Whether your programming background is Java, C#, C++ or Objective-C, you will benefit from this valuable book. There is no question that Swift is Apple's programming language of the future. This book by Paul and Harvey Deitel will be your guide to that future."

— Charles Brown, Independent Contractor affiliated with Apple and Adobe

"Fantastic, especially for those involved in iOS and OS X development. Complete examples help explain concepts clearly. Great combination of Swift topics and helpful real-world tips on working with Cocoa's Foundation classes, software engineering, performance, and error prevention. Highly recommended."

— Jack Watson-Hamblin, Programming Writer and Teacher, MotionInMotion

"The explanation of hash tables in the Dictionaries chapter is a plus; includes important tips, helping you avoid common roadblocks; every code sample is provided as a playground, letting you try out and modify the samples—cool! Great example on when to use an implicitly unwrapped failable initializer; liked how Chapter 8 kept building upon the Time class; validating property values via property observers is a nice practical tip. The Structs chapter includes great tips such as defining struct custom initializers in extensions so that you continue to get access to the autogenerated memberwise initializer, saving developers time. Great nested type example. Includes the all-important advice about when to use reference types vs. value types. Chapter 10 has a nice introduction to polymorphism; great real-world examples, including a practical illustration on how to use protocols. Perfect introduction to generics. Love the NSDecimalNumber arithmetic operator overload example in Chapter 12—it illustrates Swift's expressiveness while working with Objective-C code; this treatment of operator overloading and associated topics is very complete and includes great tips."

— René Cacheaux, iOS Architect, Mutual Mobile

"I loved how the Dictionary examples represent problems programmers actually face. The Classes chapter is very good; I love the breadth and helpful caveats. Chapter 9, Structures and Enumerations, is fantastic—gives readers a sense of how the different data types work and when each is appropriate; a thorough description of some of Swift's most awesome features. The Inheritance, Polymorphism and Protocols chapter includes the only explanation of polymorphism I've ever read in a textbook that makes sense; great use of case studies to solidify new concepts. I liked the discussion of how the Swift standard library uses generics and that the reader has been using them throughout this book. I appreciated the operator overloading examples using both custom types and Swift/Foundation ones—a really great chapter covering a super-cool feature of Swift whilst urging the appropriate level of caution."

— Ash Furrow, iOS Developer, Artsy

"A quick and enjoyable introduction to the Swift programming language. Covers Swift's strong typing, integration with Objective-C, use of the number and string primitives as well as the array and dictionary collections. I liked the way that mutability and immutability were explained, and where optionals were returned from subscripted access; the playgrounds were easy to experiment in. It was good that you created the array [Payable]—showing that protocols are also types. Good generics examples, particularly the safer Stack."

— Abizer Nasir, Freelance iOS and OS X Developer, Jungle Candy Software Ltd.

"The syntax shading really helps set the code apart. The introductory tour of Xcode was great. I liked the callouts for engineering tips and best programming practices. I thought the discussion about Hashing was particularly invaluable and informative. A good job of showing how to use structs and enums and their increased power compared to say Objective-C—I like that you returned to earlier examples and re-implemented them using structs to compare the different approaches. The Operator Overloading chapter felt like it was constantly teaching me something—it was thorough and all the tips felt appropriate."

— Robert McGovern, Independent Developer

"I'm happy with the pace, especially if I consider the target audience to be an existing Objective-C programmer. The Functions, Methods, enums and Tuples chapter is excellent. Loved the Arrays chapter—the book is worth it just for the performance tips alone; I really like the creating and initializing arrays example; sorting an Array with the method sorted and closures is a good example; the variadic parameters example is lovely. The Classes chapter is full of sound software engineering principles applied in a very understandable way to the new idioms exposed by Swift. Great enum examples. The Inheritance, Polymorphism and Protocols chapter is excellent. An excellent introduction to generics. Chapter 12 is a great intro to the more complicated aspects of String manipulation whilst showing some great, real-world operator overloading examples."

— Rik Watson, Technical Team Lead for HP Enterprise Services (Applications Services)