Object Oriented Programming In Swift Ray Wenderlich

There's a new language in town. Swift is Apple's new, native, fast, and easy to learn programming language for iOS and OS X app development. It's their "Objective-C without the C". If you are an iOS developer or planning to become one, learning Swift is your #1 priority, and Learn Swift on the Mac tells you everything you need to get up to speed, well, swiftly. You'll start with the Swift Playground and an introduction to object-oriented programming so you can immediately see Swift in action. You then learn about all of the key language features like functions and closures, classes, methods, extensions, and how Swift works just as well as Objective-C when it comes to easy memory management with ARC. Finally you'll learn how to use Swift alongside Objective-C as well as with Core Data, and you'll learn how to put all of the pieces together with a health app using Apple's new HealthKit framework.

Explore functional programming and discover new ways of thinking about code. You know you need to master functional programming, but learning one functional language is only the start. In this book, through articles drawn from PragPub magazine and articles written specifically for this book, you'll explore functional thinking and functional style and idioms across languages. Led by expert guides, you'll discover the distinct strengths and approaches of Clojure, Elixir, Haskell, Scala, and Swift and learn which best suits your needs. Contributing authors: Rich Hickey, Stuart Halloway, Aaron Bedra, Michael Bevilacqua-Linn, Venkat Subramaniam, Paul Callaghan, Jose Valim, Dave Thomas, Natasha Murashev, Tony Hillerson, Josh Chisholm, and Bruce Tate. Functional programming is on the rise because it lets you write simpler, cleaner code, and its emphasis on immutability makes it ideal for maximizing the benefits of multiple cores and distributed solutions. So far nobody's invented the perfect functional language - each has its unique strengths. In Functional Programming: A PragPub Anthology, you'll investigate the philosophies, tools, and idioms of five different functional programming languages. See how Swift, the development language for iOS, encourages you to build highly scalable apps using functional techniques like map and reduce. Discover how Scala allows you to transition gently but deeply into functional programming without losing the benefits of the JVM, while with Lisp-based Clojure, you can plunge fully into the functional style. Learn about advanced functional concepts in Haskell, a pure functional language making powerful use of the type system with type inference and type classes. And see how functional programming is becoming more elegant and friendly with Elixir, a new functional language built on the powerful Erlang base.The industry has been embracing functional programming more and more, driven by the need for concurrency and parallelism. This collection of articles will lead you to mastering the functional approach to problem solving. So put on your explorer's hat and prepare to be surprised. The goal of exploration is always discovery. What You Need: Familiarity with one or more programming languages.

Build fast and powerful applications by harnessing the power of protocol-oriented programming in Swift 4 About This Book Leverage the power of protocol-oriented programming in your applications and learn from real-world use cases Create a flexible code base with protocols and protocol extensions Leverage the power of generics in Swift 4 to create very flexible frameworks Who This Book Is For This book is for Swift developers who want to learn and implement protocol-oriented programming in their real-world applications.. What You Will Learn Understand the differences between object-oriented programming and protocol-oriented programming Explore the different types that Swift offers and what pitfalls to avoid Delve into generics and generic programming Learn how to implement Copy-On-Write within your custom types Implement several design patterns in a protocol-oriented way Design applications by prioritizing the protocol first and the implementation types second In Detail Swift has become the number one language used in iOS and macOS development. The Swift standard library is developed using protocol-oriented programming techniques, generics, and first-class value semantics; therefore, every Swift developer should understand these powerful concepts and how to take advantage of them in their application design. This book will help you understand the differences between object-oriented programming and protocol-oriented programming. It will demonstrate how to work with protocol-oriented programming using real-world use cases. You will gain a solid knowledge of the various types that can be used in Swift and the differences between value and reference types. You will be taught how protocol-oriented programming techniques can be used to develop very flexible and easy-to-maintain code. By the end of the book, you will have a thorough understanding of protocol-oriented programming and how to utilize it to build powerful and practical applications. Style and approach This book is written for developers who learn best by working with code, so every concept discussed in this book is reinforced with real code examples.

Discover the untapped features of object-oriented programming and use it with other software tools to code fast, efficient applications. Key Features Explore the complexities of object-oriented programming (OOP) Discover what OOP can do for you Learn to use the key tools and software engineering practices to support your own programming needs Book Description Your experience and knowledge always influence the approach you take and the tools you use to write your programs. With a sound understanding of how to approach your goal and what software paradigms to use, you can create high-performing applications quickly and efficiently. In this two-part book, you'll discover the untapped features of object-oriented programming and use it with other software tools to code fast and efficient applications. The first part of the book begins with a discussion on how OOP is used today and moves on to analyze the ideas and problems that OOP doesn't address. It continues by deconstructing the complexity of OOP, showing you its fundamentally simple core. You'll see that, by using the distinctive elements of OOP, you can learn to build your applications more easily. The next part of this book talks about acquiring the skills to become a better programmer. You'll get an overview of how various tools, such as version control and build management, help make your life easier. This book also discusses the pros and cons of other programming paradigms, such as aspect-oriented programming and functional programming, and helps to select the correct approach for your projects. It ends by talking about the philosophy behind designing software and what it means to be a "good" developer. By the end of this two-part book, you will have learned that OOP is not always complex, and you will know how you can evolve into a better programmer by learning about ethics, teamwork, and documentation. What you will learn Untangle the complexity of object-oriented programming by breaking it down to its essential building blocks Realize the full potential of OOP to design efficient, maintainable programs Utilize coding best practices, including TDD, pair programming and code reviews, to improve your work Use tools, such as source control and IDEs, to work more efficiently Learn how to most productively work with other developers Build your own software development philosophy Who this book is for This book is ideal for programmers who want to understand the philosophy behind creating software and what it means to be "good" at designing software. Programmers who want to deconstruct the OOP paradigm and see how it can be reconstructed in a clear, straightforward way will also find this book useful. To understand the ideas expressed in this book, you must be an experienced programmer who wants to evolve their practice.

If you plan on designing and implementing software using Swift, you need to understand the protocol-oriented programming (POP) paradigm. In this course, take a closer look at POP, and learn how to work with it to more efficiently approach app development. First, review what POP is, exactly, and how it differs from the classical object-oriented programming approach. Next, learn about the pillars of this new paradigm: protocol extensions, protocol inheritance, and protocol composition. Protocol extensions let you define default behavior for conforming types without defining a base class. Protocol inheritance is a powerful feature that lets you create more granular designs. Swift does not allow multiple inheritance for classes-but with protocol composition, Swift types can adopt multiple protocols. Plus, explore generics, and see how to implement a fully functional app using a protocol-oriented approach.

Programming in Sw ift is a concise, carefully written tutorial on the basics of the Swift language and its use in developing iOS and OS X applications. The book makes no assumptions about prior experience

with programming languages, or with Swift's precursor, Objective-C. Because of this, both beginners and experienced programmers alike can use this book to quickly and effectively learn the fundamentals of Swift programming. Readers can also learn the concepts of contemporary object-oriented programming without having to first learn all of the intricacies of a procedural language like C. This approach, combined with many small program examples and exercises at the end of each chapter, makes it ideally suited for either classroom use or self-study.

Build fast and powerful applications by harnessing the power of protocol-oriented programming in Swift 4About This Book* Leverage the power of protocol-oriented programming in your applications and learn from real-world use cases* Create a flexible code base with protocols and protocol extensions* Leverage the power of generics in Swift 4 to create very flexible frameworksWho This Book Is ForThis book is for Swift developers who want to learn and implement protocol-oriented programming in their real-world applications..What You Will Learn* Understand the differences between object-oriented programming and protocol-oriented programming* Explore the different types that Swift offers and what pitfalls to avoid* Delve into generics and generic programming* Learn how to implement Copy-On-Write within your custom types* Implement several design patterns in a protocol-oriented way* Design applications by prioritizing the protocol first and the implementation types secondIn DetailSwift has become the number one language used in iOS and macOS development. The Swift standard library is developed using protocol-oriented programming techniques, generics, and first-class value semantics; therefore, every Swift developer should understand these powerful concepts and how to take advantage of them in their application design.This book will help you understand the differences between object-oriented programming and protocol-oriented programming. It will demonstrate how to work with protocol-oriented programming using real-world use cases. You will gain a solid knowledge of the various types that can be used in Swift and the differences between value and reference types. You will be taught how protocol-oriented programming techniques can be used to develop very flexible and easy-to-maintain code.By the end of the book, you will have a thorough understanding of protocol-oriented programming and how to utilize it to build powerful and practical applications.Style and approachThis book is written for developers who learn best by working with code, so every concept discussed in this book is reinforced with real code examples.

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. 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. ¿ 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 ¿ IMPORTANT NOTE ABOUT XCODE AND SWIFT: With Xcode 6.3 and Swift 1.2, Apple introduced several changes in Swift that affect the book'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 (you’ll have to log in with your Apple developer account to see the list of downloads). ¿ 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 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 ¿

Stay motivated and overcome obstacles while learning to use Swift Playgrounds and Xcode 10.2 to become a great iOS developer. This book, fully updated for Swift 5, is perfect for those with no programming background, those with some programming experience but no object-oriented experience, or those that have a great idea for an app but haven't programmed since school. Many people have a difficult time believing they can learn to write iOS apps. Swift 5 for Absolute Beginners will show you how to do so. You'll learn Object-Oriented Programming (OOP) and be introduced to User Interface (UI) design following Apple's Human Interface Guidelines (HIG) using storyboards and the Model-View-Controller (MVC) pattern before moving on to write your own iPhone and Apple Watch apps from scratch.

Swift OS X Programming for Absolute Beginners is your step-by-step guide to learning how to code using Swift, Apple's hottest new programming language. This book will not only teach complete programming novices how to write OS X programs, but it can also help experienced programmers moving to the Macintosh for the first time. You will learn to understand the principles of programming, how to use Swift and Xcode, and how to combine your knowledge into writing OS X programs. If you've always wanted to learn coding but felt stymied by the limitation of simplistic programming languages or intimidated by professional but complicated programming languages, then you'll want to learn Swift. Swift is your gateway to both Macintosh and iOS app development while being powerful and easy to learn at the same time, and Swift OS X Programming for Absolute Beginners is the perfect place to start - add it to your library today.

Jump into the app development world with confidence! iOS Swift 24-Hour Trainer combines book and video lessons in Apple's Swift programming language to prepare you to build iPhone and iPad apps—and distribute them through the Appstore. First, this approachable text covers the fundamentals of Swift by introducing you to iOS development in this language, and presenting best practices for setting up a development environment and using variables, statements, expressions, operators, functions, and closures. Next, you explore common tasks, such as alert views, table views, and collection views. You then deepen your knowledge of Swift by considering network programming and local data storage. Finally, this engaging resource dives into slightly more advanced concepts, such as tab bars, web views, the

accelerometer, camera, photo library, Google maps, and core location. Swift was designed by Apple to incorporate modern scripting features while offering simpler, cleaner syntax than Objective-C to maintain a minimal and easy to read style. This more expressive code offers numerous key features, such as closures unified with function pointers, tuples and multiple value returns, generics, and functional programming patterns. Learn how to obtain a device UDID Test your applications on an actual device, so you can see your work in action Distribute your applications outside of the App store, allowing you to test your work with real users Review common reasons why apps are rejected by Apple to strengthen your case when submitting your apps for distribution iOS Swift 24-Hour Trainer is an essential guide to Apple's Swift programming language for beginning programmers.

Build fast and powerful applications by exploiting the power of protocol-oriented programming in SwiftAbout This Book• The only book that shows how to harness the power of Protocol-Oriented Programming in Swift to build real-world applications, • Get familiar with the protocol focused approach of application development, • Increase the overall productivity and performance of applications with Protocol Oriented Programming.Who This Book Is ForThis book is for Swift developers who want to learn and implement protocol oriented programming in their real world applications.What You Will Learn• The difference between Object-Oriented programming and Protocol-Oriented programming• The difference between reference and value types and when to use each• How we can leverage tuples to reduce the complexity of our code• What are protocols and how to use them• How to implement protocol extensions to create a very flexible code base• How to implement several design patterns in a Protocol-Oriented approach• How to solve real world design issue with protocol oriented programmingIn DetailAt the heart of Swift's design is an incredibly powerful idea: protocol-oriented programming. Its many benefits include better code maintainability, increased developer productivity and superior application performance. The book will teach the reader how to apply the ideas behind the protocol oriented programing paradigm to improve the code they write.This book will introduce the readers to the world of protocol-oriented programming in Swift and will demonstrate the ideas behind this new programming paradigm with real world examples. In addition to learning the concepts of Protocol Oriented programming, it also shows the reader how to reduce the complexity of their codebase using protocol extensions. Beginning with how to create simple protocols, readers will learn how to extend protocols and also to assign behaviors to them.By the end of this book readers will be able to harness the power of protocol-oriented programming to build real world applications.Style and approachIn its latest release of Swift, Apple has introduced Protocol Extensions as a new feature at the heart of Swifts design making Swift 2 a protocol-oriented language. Protocol oriented programming being a less explored OOP paradigm, there is little guidance on hot to take advantage of protocol extensions in real-world applications. In addition to offering an in-depth coverage of protocol oriented programming and its concepts, this book also explains how a developer can leverage these features to build powerful, real-world applications

Zero to iOS Hero is an easy-to-read, fully comprehensive book aimed at helping students become iOS app developers, without any prior knowledge. With this book, anyone can go from having zero experience in computer science to programming noteworthy applications over the course of four simple sections. Along the way, you'll also get to build 6 brand-new apps, from the ground up. The simple and straightforward lessons in this book use Xcode 9, Swift 4, and iOS 12 to help you transform your idea to a fully-functional app. What you'll learn in Zero to iOS Hero: Explore the Xcode environment paired with the Swift language, Apply your knowledge in using some of Swift's intermediate and advanced features, Learn about fundamental computer science concepts, Employ data structures within Swift, Delve into object-oriented programming for iOS, and Create SIX New Apps! The mission of No Stoppin' is to empower students by promoting and enabling peer-to-peer education by authoring educational content students wouldn't have access to otherwise.

Build fast and powerful applications with the power of protocol-oriented programmingAbout This Book- Leverage the power of protocol-oriented programming in your applications and learn from real world use cases- Create a flexible codebase with protocols and protocol extensions- Increase the overall productivity and performance of applications with protocol-oriented programmingWho This Book Is ForThis book is for Swift developers who want to learn and implement protocol-oriented programming in practical applications.What You Will Learn- Understand the difference between object-oriented programming and protocol-oriented programming- Explore the different types that Swift offers and what pitfalls to avoid- Error handling with do-try-catch block- Delve into Generics and Generic programming- Implement several design patterns in a protocol-oriented way- How to design applications by prioritizing the protocol first and the actual type secondIn DetailOne of the most important additions to the new features and capabilities of the Swift programming language was an overhaul of Protocols. Protocol-oriented programming and first class value semantics have now become two incredibly powerful concepts at the heart of Swift's design.This book will help you understand the difference between object-oriented programming and protocol-oriented programming. It will demonstrate how to work with protocol-oriented programming using real world use cases. You will gain solid knowledge of the different types that can be used in Swift and the differences between value and reference types. You will be taught how to utilize the advanced features of protocol-oriented programming to boost the performance of your applications.By the end of the book, you will have a thorough understanding of protocol-oriented programming and how to utilize it to build powerful, practical applications.Style and approachThis book is written for developers who learn best by working with code, so every concept discussed in this book is reinforced with real code examples.

Summary Classic Computer Science Problems in Swift invites readers to invest their energy in some foundational techniques that have been proven to stand the test of time. Along the way they'll learn intermediate and advanced features of the Swift programming language, a worthwhile skill in its own right. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the Technology Don't just learn another language. Become a better programmer instead. Today's awesome iOS apps stand on the shoulders of classic algorithms, coding techniques, and engineering principles. Master these core skills in Swift, and you'll be ready for AI, data-centric programming, machine learning, and the other development challenges that will define the next decade. About the Book Classic Computer Science Problems in Swift deepens your Swift language skills by exploring foundational coding techniques and algorithms. As you work through examples in search, clustering, graphs, and more, you'll remember important things you've forgotten and discover classic solutions to your "new" problems. You'll appreciate author David Kopec's amazing ability to connect the core disciplines of computer science to the real-world concerns of apps, data, performance, and even nailing your next job interview! What's Inside Breadth-first, depth-first, and A* search algorithms Constraint-satisfaction problems Solving problems with graph algorithms Neural networks, genetic algorithms, and more All examples written in Swift 4.1 About the Reader For readers comfortable with the basics of Swift. About the Author David Kopec is an assistant professor of computer science and innovation at Champlain College in Burlington, Vermont. He is an experienced iOS developer and the author of Dart for Absolute Beginners. Table of Contents Small problems Search problems Constraint-satisfaction problems Graph problems Genetic algorithms K-means clustering Fairly simple neural networks Miscellaneous problems

Understanding the Protocol-Oriented Programming (POP) paradigm is imperative if you plan on designing and implementing software using Swift 5. In this book, you'll learn how to work with POP to approach app development more efficiently. First, we review what POP is and how it differs from the classical object-oriented programming approach. Next, we discuss the pillars of this new paradigm: protocol extensions, protocol inheritance, and protocol composition. In the last part of this book, we're going to implement a fully functional app using the protocol-oriented approach. Topics include: What's protocol-oriented programming? The pillars of POP Defining method requirements Class-bound protocols Adopting a protocol Generics and protocols Implementing an app from scratch using POP Throughout the

book, you'll acquire coding skills that can be applied in real-world situations. About the Author Karoly Nyisztor is a veteran software engineer and instructor. He has worked with large companies such as Apple, Siemens, and SAP. Karoly has designed and built several enterprise frameworks, and he holds twelve patents related to inventions in the field of mobile computing. After 18 years, he left the corporate world to start his own business. Since 2016, he's fully committed to teaching. As an instructor, he aims to share his 20+ years of software development expertise. Karoly teaches Software Architecture, Object-Oriented Programming and Design, Python, Swift and iOS Programming, and other, programming-related topics. You can find his courses and books on all major platforms including Amazon, LinkedIn Learning, Pluralsight, Udemy, and iTunes.

In just 24 sessions of one hour each, learn how to build powerful applications for today's hottest handheld devices: the iPhone and iPad! Using this book's straightforward, step-by-step approach, you'll master every skill and technology you need, from setting up your iOS development environment to building great user interfaces, sensing motion to writing multitasking applications. Each lesson builds on what you've already learned, giving you a rock-solid foundation for real-world success! Step-by-step instructions carefully walk you through the most common iOS development tasks. Quizzes and Exercises help you test your knowledge. By the Way notes present interesting information related to the discussion. Did You Know? tips show you easier ways to perform tasks. Watch Out! cautions alert you to possible problems and give you advice on how to avoid them. John Ray is currently serving as the Director of the Office of Research Information Systems at the Ohio State University. His many books include Using TCP/IP: Special Edition, Maximum Mac OS X Security, Mac OS X Unleashed, Teach Yourself Dreamweaver MX in 21 Days, and Sams Teach Yourself iOS 7 Application Development in 24 Hours. Printed in full color–figures and code appear as they do in Xcode Covers iOS 8 and up Learn to navigate the Xcode 6.x development environment Prepare your system and iDevice for efficient development Get started quickly with Apple's new language: Swift Test code using the new iOS Playground Understand the Model-View-Controller (MVC) development pattern Visually design and code interfaces using Xcode Storyboards, Segues, Exits, Image Slicing, and the iOS Object Library Use Auto Layout and Size Classes to adapt to different screen sizes and orientations Build advanced UIs with Tables, Split Views, Navigation Controllers, and more Read and write preferences and data, and create System Settings plug-ins Use the iOS media playback and recording capabilities Take photos and manipulate graphics with Core Image Sense motion, orientation, and location with the accelerometer, gyroscope, and GPS Integrate online services using Twitter, Facebook, Email, Web Views, and Apple Maps Create universal applications that run on both the iPhone and iPad Write background-aware multitasking applications Trace, debug, and monitor your applications as they run

iOS 12 App Development Essentials, the latest edition of this popular book series, has now been fully updated for the iOS 12 SDK, Xcode 10 and the Swift 4 programming language. Beginning with the basics, this book provides an outline of the steps necessary to set up an iOS development environment. An introduction to the architecture of iOS 12 and programming in Swift 4 is provided, followed by an in-depth look at the design of iOS applications and user interfaces. More advanced topics such as file handling, database management, graphics drawing and animation are also covered, as are touch screen handling, gesture recognition, multitasking, location management, local notifications, camera access and video playback support. Other features are also covered including Auto Layout, local map search, user interface animation using UIKit dynamics, Siri integration, iMessage app development, CloudKit sharing and biometric authentication. Additional features of iOS development using Xcode are also covered, including Swift playgrounds, universal user interface design using size classes, app extensions, Interface Builder Live Views, embedded frameworks, collection and stack layouts and CloudKit data storage in addition to drag and drop integration and the document browser. The key new features of iOS 12 and Xcode 10 are also covered in detail, including Siri shortcuts and the new iOS machine learning features. The aim of this book, therefore, is to teach you the skills necessary to build your own apps for iOS 12. Assuming you are ready to download the iOS 12 SDK and Xcode 10, have an Intel-based Mac and ideas for some apps to develop, you are ready to get started.

iOS 8 App Development Essentials is latest edition of this popular book series and has now been fully updated for the Swift 1.2 programming language, the iOS 8 SDK and Xcode 6.3. Beginning with the basics, this book provides an outline of the steps necessary to set up an iOS development environment. An introduction to the architecture of iOS 8 and programming in Swift is provided, followed by an in-depth look at the design of iOS applications and user interfaces. More advanced topics such as file handling, database management, in-app purchases, graphics drawing and animation are also covered, as are touch screen handling, gesture recognition, multitasking, iAds integration, location management, local notifications, camera access and video and audio playback support. Other features are also covered including Auto Layout, Twitter and Facebook integration, App Store hosted in-app purchase content, collection views, Sprite Kit-based game development, local map search and user interface animation using UIKit dynamics. The key new features of the iOS 8 SDK and Xcode 6 are also covered, including Swift playgrounds, universal user interface design using size classes, app extensions, Interface Builder Live Views, embedded frameworks, CloudKit data storage and TouchID authentication. The aim of this book is to teach the range of skills necessary to build apps for iOS 8. iOS 8 App Development Essentials takes a modular approach to the subject of iOS 8 application development for both the iPhone and iPad, with each chapter covering a self contained topic area consisting of detailed explanations, examples and step-by-step tutorials. This makes the book both an easy to follow learning aid and an excellent reference resource.

The Swift standard library is developed using protocol-oriented programming techniques, generics, and first-class value semantics; therefore it is important that every Swift developer understand these powerful concepts and how to take advantage of them.This book will demonstrate how to use protocol-oriented programming techniques to build ...

Implement object-oriented programming paradigms with Swift 3.0 and mix them with modern functional programming techniques to build powerful real-world applications About This Book Leverage the most efficient object-oriented design patterns in your Swift applications Write robust, safer, and better code using the blueprints that generate objects Build a platform with object-oriented code using real-world elements and represent them in your apps Who This Book Is For This book is for iOS and macOS developers who want to get a detailed practical understanding of object-oriented programming with the latest version of Swift: 3.0. What You Will Learn Write high-quality and easy-to-maintain reusable object-oriented code to build applications for iOS, macOS, and Linux Work with encapsulation, abstraction, and polymorphism using Swift 3.0 Work with classes, instances, properties, and methods in Swift 3.0 Take advantage of inheritance, specialization, and the possibility to overload or override members Implement encapsulation, abstraction, and polymorphism Explore functional programming techniques mixed with object-oriented code in Swift 3.0 Understand the differences between Swift 3.0, previous Swift versions, and Objective-C code In Detail Swift has quickly become one of the most-liked languages and

developers' de-facto choice when building applications that target iOS and macOS. In the new version, the Swift team wants to take its adoption to the next level by making it available for new platforms and audiences. This book introduces the object-oriented paradigm and its implementation in the Swift 3 programming language to help you understand how real-world objects can become part of fundamental reusable elements in the code. This book is developed with XCode 8.x and covers all the enhancements included in Swift 3.0. In addition, we teach you to run most of the examples with the Swift REPL available on macOS and Linux, and with a Web-based Swift sandbox developed by IBM capable of running on any web browser, including Windows and mobile devices. You will organize data in blueprints that generate instances. You'll work with examples so you understand how to encapsulate and hide data by working with properties and access control. Then, you'll get to grips with complex scenarios where you use instances that belong to more than one blueprint. You'll discover the power of contract programming and parametric polymorphism. You'll combine generic code with inheritance and multiple inheritance. Later, you'll see how to combine functional programming with object-oriented programming and find out how to refactor your existing code for easy maintenance. Style and approach This simple guide is packed with practical examples of solutions to common problems. Each chapter includes exercises and the possibility for you to test your progress by answering a quiz

Software developers need to solve various problems. Many times, these problems are the same or similar to the ones they've already encountered in other projects. Wouldn't it be great to apply the solution you've found instead of reinventing the wheel over and over again?That's precisely the reason why software design patterns exist. A design pattern is a standardized way to address a recurring problem. Relying on a proven strategy will not only save you time, but you can rest assured that it's indeed the right choice. Design patterns are the result of a long evolution process. It all started with a book published in 1994 - yes, it's that old! - called "Design Patterns - Elements of Reusable Object-Oriented Software." That's a quite tedious title, so we usually refer to it as "the book by the gang of four." The gang consists of four renowned software engineers: Erich Gamma, Ralph Johnson, Richard Helm, and John Vlissides. They identified the most significant common issues that occurred in multiple projects and developed best practices to solve them. The best part: these solutions are (programming) language-agnostic. You can use the design patterns with any object-oriented programming language. Many modern programming languages and frameworks have integrated the GoF patterns. You don't have to write additional code to support say the Iterator or the Observer.Swift is no exception. Actually, it provides many advanced language features and constructs --such as type extensions, lazy initialization, and predefined protocols -- that let us adopt and integrate the design patterns into our projects easily. This book covers all these topics and provides best practices you can apply in your upcoming projects.

You are one step away from programming ios 14 apps using Swift 5.3 and Xcode if you take the decision to purchase this book.Have you ever wondered how iOS apps are built and designed? If your answer is yes, then you are in for a long and exciting journey with this guide. Apps in the Apple Playstore are built with Swift, which is a general-purpose, multi-paradigm, compiled programming language. Swift is used with Xcode which is an Apple's integrated development environment (IDE) used for building software for devices using iOS. Swift is similar to Python as it is also an object-oriented programming language. Apps built on this platform can be uploaded on the Apple Playstore by the developer. Programming iOS 14 using Swift and Xcode: A step by step beginners to pro guide to programming iOS 14 using Swift 5.2 and Xcode 12.2 provides both new and existing app developers a theoretical and practical approach to learning Apple's Swift programming language. Several theories explained in this guide has a tutorial chapter attached to it for practical learning. Topics covered in this guide include and are not limited to: Swift Playgrounds Swift Data types Swift Operators and Expressions SwiftUI and UIKit Error Handling SwiftUI Stacks and Frames Uploading the App to the Apple Playstore These represent a few of the chapters covered in this simple and straightforward guide. Start your journey of becoming an iOS app developer today. Scroll up and click the BUY NOW WITH 1-CLICK BUTTON

"Swift is a general-purpose, multi-paradigm programming language created mainly for iOS and OSX. Designed to replace Objective C, Swift has quickly gained popularity in the developer community and is now the 14th most popular programming language in the TIOBE index. This course will be your companion to get familiarized with the features of Swift 3.0, and get you up and running with writing your own Swift code in no time.You will start with understanding the basics of object-oriented programming, and get introduced to Swift. You will learn about its features, code syntax and write your first application. Work with variables, strings, arrays, and the newly introduced Xcode playground. If you already have worked with Swift 2, this course will also show you how you can port your Swift 2 code to Swift 3 without any hassle. An exercise is provided at the end of each section of this tutorial to test the skills you have learned, and you will also develop several Swift applications throughout the tutorial to put your knowledge to practical use.With regularly updated information and code samples, this course will ensure your success in setting up a solid foundation of programming with Swift 3."--Resource description page.

Covers iOS 9.1 and up, Xcode 7.x, iPhone, iPad, and More! In just 24 sessions of one hour each, learn how to build powerful applications for today’s hottest handheld devices: the iPhone and iPad! Using this book’s straightforward, step-by-step approach, you’ll master every skill and technology you need, from setting up your iOS development environment to building great user interfaces, sensing motion to writing multitasking applications. Each lesson builds on what you’ve already learned, giving you a rock-solid foundation for real-world success! Step-by-step instructions carefully walk you through the most common iOS development tasks. Quizzes and Exercises help you test your knowledge. Notes present interesting information related to the discussion. Tips show you easier ways to perform tasks. Cautions alert you to possible problems and give you advice on how to avoid them. Printed in full color—figures and code appear as they do in Xcode 7.x • Learn to navigate the Xcode 7.x development environment and install apps on your iDevice • Get started quickly with Apple’s Open Source language: Swift 2.0 • Test code and application logic using the iOS Playground • Understand the Model-View-Controller (MVC) development pattern • Visually design and code interfaces using Xcode Storyboards, Segues, Exits, Image Slicing, and the iOS Object Library • Use Auto Layout and Size Classes to adapt to different screen sizes and orientations • Build advanced UIs with Tables, Split Views, Navigation Controllers, and more • Read and write preferences and data, and create System Settings plug-ins • Use iOS media playback and recording capabilities • Take photos and manipulate graphics with Core Image • Sense motion, orientation, and location with the accelerometer, gyroscope, and GPS • Use 3D touch to add Peek, Pop, and Quick Actions to your apps • Integrate online services using Twitter, Facebook, Email, Web Views, and Apple Maps • Create universal applications that run on both the iPhone and iPad • Write background-aware

multitasking applications • Trace, debug, and monitor applications as they run • Additional files and updates available online

Over 50 recipes to help you quickly and efficiently build applications with Swift 4 and Xcode 9 About This Book Write robust and efficient code and avoid common pitfalls using Swift 4 Get a comprehensive coverage of the tools and techniques needed to create multi-platform apps with Swift 4 Packed with easy-to-follow recipes, this book will help you develop code using the latest version of Swift Who This Book Is For If you are looking for a book to help you learn about the diverse features offered by Swift 4 along with tips and tricks to efficiently code and build applications, then this book is for you. Basic knowledge of Swift or general programming concepts will be beneficial. What You Will Learn Explore basic to advanced concepts in Swift 4 Programming Unleash advanced features of Apple's Xcode 9 IDE and Swift Playgrounds Learn about the conditional statements, loops, and how to handle errors in Swift Define flexible classes and structs using Generics, and learn about the advanced operators, and create custom operators Explore functionalities outside of the standard libraries of Swift Import your own custom functionality into Swift Playgrounds Run Swift on Linux and investigate server-side programming with the server side framework Vapor In Detail Swift 4 is an exciting, multi-platform, general-purpose programming language. Being open source, modern and easy to use has made Swift one of the fastest growing programming languages. If you interested in exploring it, then this book is what you need. The book begins with an introduction to the basic building blocks of Swift 4, its syntax and the functionalities of Swift constructs. Then, introduces you to Apple's Xcode 9 IDE and Swift Playgrounds, which provide an ideal platform to write, execute, and debug the codes thus initiating your development process. Next, you'll learn to bundle variables into tuples, set order to your data with an array, store key-value pairs with dictionaries and you'll learn how to use the property observers. Later, explore the decision-making and control structures in Swift and learn how to handle errors in Swift 4. Then you'll, examine the advanced features of Swift, generics and operators, and then explore the functionalities outside of the standard library, provided by frameworks such as Foundation and UIKit. Also, you'll explore advanced features of Swift Playgrounds. At the end of the book, you'll learn server-side programming aspect of Swift 4 and see how to run Swift on Linux and then investigate Vapor, one of the most popular server-side frameworks for Swift. Style and approach Each recipe addresses a specific problem, with a detailed discussion that explains the solution and offers insight into how it works.

Stay motivated and overcome obstacles while learning to use Swift Playgrounds and Xcode 10.2 to become a great iOS developer. This book, fully updated for Swift 5, is perfect for those with no programming background, those with some programming experience but no object-oriented experience, or those that have a great idea for an app but haven’t programmed since school. Many people have a difficult time believing they can learn to write iOS apps. Swift 5 for Absolute Beginners will show you how to do so. You'll learn Object-Oriented Programming (OOP) and be introduced to User Interface (UI) design following Apple’s Human Interface Guidelines (HIG) using storyboards and the Model-View-Controller (MVC) pattern before moving on to write your own iPhone and Apple Watch apps from scratch. What You’ll Learn Work with Swift classes, properties, and functions Examine proper User Interface (UI) and User Experience (UX) design Understand Swift data types: integers, floats, strings, and booleans Use Swift data collections: arrays and dictionaries Review Boolean logic, comparing data, and flow control Use the Xcode debugger to troubleshoot problems with your apps Store data in local app preferences and Core Data databases Who This Book Is For Anyone who wants to learn to develop apps for the Mac, iPhone, iPad, and Apple Watch using the Swift programming language. No previous programming experience is necessary.

Get to grips with object-oriented programming in Swift to efficiently build powerful real-world applications About This Book Leverage the most efficient object-oriented design patterns in your Swift applications Write robust, safer, and better code using the blueprints that generate objects Build a platform with object-oriented code by using real-world elements and represent them in your app Who This Book Is For If you are an iOS developer who has a basic idea of object-oriented programming and want to incorporate its concepts with Swift to optimize your application's code and create reusable and easily to understand building blocks, then this book is for you. This is a very useful resource for developers who want to shift from Objective C, C#, Java, Python, JavaScript, or other object-oriented languages to Swift What You Will Learn Build solid, stable, and reliable applications using Swift Work with encapsulation, abstraction, and polymorphism using Swift 2.0 Customize constructors and destructors based on your needs Develop Swift 2.0 with classes, instances, properties, and methods Take advantage of generic code to maximize code reuse and generalize behaviors Use state of inheritance, specialization, and the possibility to overload members Write high quality object-oriented code to build apps for iOS or Mac OS X In Detail Object-Oriented Programming (OOP) is a programming paradigm based on the concept of objects; these are data structures that contain data in the form of fields, often known as attributes and code. Objects are everywhere, and so it is very important to recognize elements, known as objects, from real-world situations and know how they can easily be translated into object-oriented code. Object-Oriented Programming with Swift is an easy-to-follow guide packed full of hands-on examples of solutions to common problems encountered with object-oriented code in Swift. It starts by helping you to recognize objects using real-life scenarios and demonstrates how working with them makes it simpler to write code that is easy to understand and reuse. You will learn to protect and hide data with the data encapsulation features of Swift. Then, you will explore how to maximize code reuse by writing code capable of working with objects of different types. After that, you'll discover the power of parametric polymorphism and will combine generic code with inheritance and multiple inheritance. Later, you move on to refactoring your existing code and organizing your source for easy maintenance and extensions. By the end of the book, you will be able to create better, stronger, and more reusable code, which will help you build better applications. Style and approach This simple guide is packed with practical examples of solutions to common problems. Each chapter includes exercises and the possibility for you to test your progress by answering questions.

Stay motivated and overcome obstacles while learning to use Swift Playgrounds to be a great iOS developer. This book is perfect for those with no programming background, those with some programming experience but no object-oriented experience, or those that have a great idea for an app but haven’t programmed since school, and it is now updated for Swift 4. Many people have a difficult time believing they can learn to write iOS apps. Swift 4 for Absolute Beginners will show you how to do so. You'll learn Object Oriented Programming and be introduced to HealthKit before moving on to write your own iPhone and Watch apps from scratch. Gary Bennett and Brad Lees are full-time professional iOS developers and have developed a broad spectrum of apps for Fortune 500 companies. The authors have taken their combined 14 years of writing apps, teaching online iOS courses, the experience from their first three iOS books, along with their online instruction and free online forum at XcelMe.com to create an excellent training book. And the material in this book is supplemented by with the free, live online training

sessions. What You'll Learn Work with Swift classes, properties, and functions Examine proper user interface and user experience design Understand Swift data types: integers, floats, strings, and Booleans Use Swift data collections: arrays and dictionaries Review Boolean logic, comparing data, and flow control Who This Book Is For Anyone who wants to learn to develop apps for the Mac, iPhone, and iPad, and Watch using the Swift programming language. No previous programming experience is necessary.

Learn the HomeKit platform structure and how it supports devices—existing and planned—and get a thorough grounding on new and useful apps that deliver a new generation of home automation in a secure and innovative environment. Let your imagination run wild as you design compatible devices with unlimited capabilities. Learn Apple HomeKit on iOS shows you how to move to secure, home automation projects that integrate with your digital world automatically—after you set them up as described in the book. Having your calendar and appointments control your lights, locks, thermostat, and other home devices is the heart of home automation. In homes and small offices, you can banish notes taped to switches and controls that say, "Do not turn off this switch" or "Leave the thermostat alone." The book gets you up to speed on HomeKit, and it also answers some of the pesky questions, such as "What happens when the power goes out?" Along the way there are tips and suggestions for app developers, hardware manufacturers, interior designers, and real estate professionals. For programmers, there's an entire chapter (plus sections in other chapters) dedicated to the core coding issues. For non-programmers, this book is the perfect resource mastering the amazing potential of Apple HomeKit. What You Will Learn: For device developers, understanding the structure of HomeKit—homes, rooms, and accessories—enables you to build devices that are easily managed by a single, simple source and interface. For DIY home networking users, gain a thorough knowledge of how they can adapt HomeKit to their existing spaces. For programmers, there's an entire chapter plus sections in other chapters dedicated to the core coding issues you'll need to learn. For non-programmers, this book is your perfect resource for easily getting your mind around the amazing potential of Apple HomeKit. Author Jesse Feiler develops, consults, and writes about Apple technologies with an emphasis on mobile and location-based apps. Who This Book Is For: Device developers, DIY home networking users, programmers, and those interested in integrating their iOS devices with their IoT devices.

The goal of this book is to teach the skills necessary to build iOS 14 applications using SwiftUI, Xcode 12 and the Swift 5.3 programming language. Beginning with the basics, this book provides an outline of the steps necessary to set up an iOS development environment together with an introduction to the use of Swift Playgrounds to learn and experiment with Swift. The book also includes in-depth chapters introducing the Swift 5.3 programming language including data types, control flow, functions, object-oriented programming, property wrappers and error handling. An introduction to the key concepts of SwiftUI and project architecture is followed by a guided tour of Xcode in SwiftUI development mode. The book also covers the creation of custom SwiftUI views and explains how these views are combined to create user interface layouts including the use of stacks, frames and forms. Other topics covered include data handling using state properties in addition to observable, state and environment objects, as are key user interface design concepts such as modifiers, lists, tabbed views, context menus, user interface navigation, and outline groups. The book also includes chapters covering graphics drawing, user interface animation, view transitions and gesture handling, WidgetKit, document-based apps and SiriKit integration. Chapters are also provided explaining how to integrate SwiftUI views into existing UIKit-based projects and explains the integration of UIKit code into SwiftUI. Finally, the book explains how to package up a completed app and upload it to the App Store for publication. Along the way, the topics covered in the book are put into practice through detailed tutorials, the source code for which is also available for download. The aim of this book, therefore, is to teach you the skills necessary to build your own apps for iOS 14 using SwiftUI. Assuming you are ready to download the iOS 14 SDK and Xcode 12 and have an Apple Mac system you are ready to get started.

Swift (programming language) ... Swift is a general-purpose, multi-paradigm, compiled programming language developed by Apple Inc. for iOS, iPadOS, macOS, watchOS, tvOS, Linux, and z/OS.Language paradigms: Compiled languageParent language: Objective-CLanguage designers: AppleBest Programming Language for iOS App Development Apple iPhone and iPad products have become the standard of mobile smartphones and tablets. Apple Watch is one of the most-sold smartwatches in the world. All of these Apple devices are powered by Apple's operating system, iOS. Best Programming Language for iOS App Development If you want to build iPhone, iPad, or Apple Watch apps, you need to learn iOS development. In this article, I will discuss the most popular iOS development programming languages and which language you should choose for your next iOS app. What programming languages can be used to develop iOS mobile apps? Here is a list of the most popular programming languages used to develop iOS apps. 1. Objective-C2. Swift3. C#4. Python5. C++6. HTML 5 Objective-C Objective-C was developed by Tom Love and Brad Cox in 1984. Prior to Apple launching Swift in 2014, Objective C was the primary language of Apple iOS mobile apps. Objective-C is a general-purpose, object-oriented programming language that brings Smalltalk flavor to C programming language. Message passing among objects is a key feature of Objective-C that became really useful for Apple iOS operating systems. Today, Swift has taken over Objective-C in popularity and usefulness. Objective-C is a superset of the C programming language and provides object-oriented capabilities and a dynamic runtime. Objective-C inherits the syntax, primitive types, and flow control statements of C and adds syntax for defining classes and methods. It also adds language-level support for object graph management and object literals while providing dynamic typing and binding, deferring many responsibilities until runtime. Swift Swift is the primary programming language of the iOS operating system. Swift was developed and launched by Apple in 2014. In Dec 2015, Apple open-sourced Swift under the Apache License 2.0. Besides iOS, Swift is also a programming language of macOS, watchOS, tvOS, Linux and z/OS. Prior to Swift, Objective-C was the primary language for iOS development. Objective C being 30 years old, the language did not support modern needs. Swift is a modern programming language that provides modern language features such as dynamic, safe, late binding, and extensibility. Earlier in 2018, Swift surpassed Objective-C in popularity and became the #1 programming language for iOS and other Apple operating systems. Swift is a highly recommend language for building your new iOS, tvOS, and watchOS platforms. To learn Swift, here is a complete training course on Swift. iOS Development with Swift 4 includes ARKit, CoreML, App Design and much more.Master iOS Programming Using Swift

Covering the latest in Java technologies, Object-Oriented Programming and Java teaches the subject in a systematic, fundamentals-first approach. It begins with the description of real-world object interaction scenarios and explains how they can be translated, represented and executed using object-oriented programming paradigm. By establishing a solid foundation in the understanding of object-oriented programming concepts and their applications, this book provides readers with the pre-requisites for writing proper object-

oriented programs using Java.
The two volume set LNCS 12506 and 12507 constitutes the proceedings of the 19th International Semantic Web Conference, ISWC 2020, which was planned to take place in Athens, Greece, during November 2-6, 2020. The conference changed to a virtual format due to the COVID-19 pandemic. The papers included in this volume deal with the latest advances in fundamental research, innovative technology, and applications of the Semantic Web, linked data, knowledge graphs, and knowledge processing on the Web. They were carefully reviewed and selected for inclusion in the proceedings as follows: Part I: Features 38 papers from the research track which were accepted from 170 submissions; Part II: Includes 22 papers from the resources track which were accepted from 71 submissions; and 21 papers in the in-use track, which had a total of 46 submissions.
Swift Protocol-Oriented ProgrammingIncrease productivity and build faster applications with Swift 5, 4th EditionPackt Publishing Ltd
It is incredible to think that a programming language developed in 1995 in response to the shortcomings of the prevalent language at the time, C, remains one of the world's most popular coding languages more than twenty years later. This is the ongoing legacy of Java, which is hailed as easy to use for a variety of goals and an important part of today's technology. This book traces the evolution of the language and explains how the language works and what it's used for, including Java's role in big data and the internet of things.
Transition from Objective-C to the cleaner, more functionalSwift quickly and easily Professional Swift shows you how to create Mac and iPhoneapplications using Apple's new programming language. Thiscode-intensive, practical guide walks you through Swift bestpractices as you learn the language, build an application, andrefine it using advanced concepts and techniques. Organized foreasy navigation, this book can be read end-to-end for a self-pacedtutorial, or used as an on-demand desk reference as unfamiliarsituations arise. The first section of the book guides you throughthe basics of Swift programming, with clear instruction oneverything from writing code to storing data, and Section II addsadvanced data types, advanced debugging, extending classes, andmore. You'll learn everything you need to know to make thetransition from Objective-C to Swift smooth and painless, so youcan begin building faster, more secure apps than ever before. Get acquainted with the Swift language and syntax Write, deploy, and debug Swift programs Store data and interface with web services Master advanced usage, and bridge Swift and Objective-C Professional Swift is your guide to the future of OS Xand iOS development.