

Swift 3 Functional Programming

The professional programmer's Deitel® guide to iPhone® and iPad® app development using iOS® 8, Swift™, Xcode® 6, and Cocoa Touch® This book presents leading-edge computing technologies for professional software developers. At the heart of the book is the Deitel “app-driven approach”—a variant of Deitel’s live-code approach—concepts are presented in the context of complete working iOS apps, rather than using code snippets. The introduction and app test drives at the beginning of each chapter show one or more sample executions. The book’s source code is available at:

www.deitel.com/books/iOS8FP1. You’ll quickly learn everything you need to start building iOS 8 apps—beginning with a test-drive of the Tip Calculator app in Chapter 1, then building your first apps in Chapter 2 with visual programming and in Chapter 3 with Swift. By the time you reach Chapter 9, you’ll be ready to create your own apps for submission to the App Store. We’ll overview the submission process, including uploading your apps, deciding whether to sell your apps or offer them for free, and marketing them using in-app advertising, social media, Internet public relations and more. This volume contains the papers presented at the 4th Fuji International Symposium on Functional and Logic Programming (FLOPS’99) held in Tsukuba, Japan, November 11–13, 1999, and hosted by the Electrotechnical Laboratory (ETL). FLOPS is a forum for presenting and discussing all issues concerning functional programming, logic programming, and their integration. The symposium takes place about every 1.5 years in Japan. Previous FLOPS meetings were held in Fuji Susuno (1995), Shonan Village (1996), and Kyoto (1998). There were 51 submissions from Austria (), Belgium (2), Brazil(3), China 3 3 1 7 (1), Denmark (2), France (3), Germany (8), Ireland (1), Israel (), Italy (1), 4 3 12 1 Japan (9), Korea (1), Morocco (1), The Netherlands (1), New Zealand (1), 3 1 1 3 5 Portugal (), Singapore (), Slovakia (1), Spain (4), Sweden (1), UK (4), 2 3 4 6 1 and USA (2), of which the program committee selected 21 for presentation. In 4 addition, this volume contains full papers by the two invited speakers, Atsushi Ohori and Mario Rodr??iguez-Artalejo.

This work, a tribute to renowned researcher Robert Paige, is a collection of revised papers published in his honor in the Higher-Order and Symbolic Computation Journal in 2003 and 2005. Among them there are two key papers: a retrospective view of his research lines, and a proposal for future studies in the area of the automatic program derivation. The book also includes some papers by members of the IFIP Working Group 2.1 of which Bob was an active member.

Core Data best practices by example: from simple persistency to multithreading and syncing This book strives to give you clear guidelines for how to get the most out of Core Data while avoiding the pitfalls of this flexible and powerful framework. We start with a simple example app and extend it step by step as we talk about relationships, advanced data types, concurrency, syncing, and many other topics. Later on, we go well beyond what's needed for the basic example app. We'll discuss in depth how Core Data works behind the scenes, how to get great performance, the trade-offs between different Core Data setups, and how to debug and profile your Core Data code. All code samples in this book are written in Swift. We show how you can leverage Swift's language features to write elegant and safe Core Data code. We expect that you're already familiar with Swift and iOS, but both newcomers and experienced Core Data developers will find a trove of applicable information and useful patterns.

Write and run Swift language programs in the Cloud Written by the team of developers that has helped bring the Swift language to Cloud computing, this is the definitive guide to writing and running Swift language programs for cloud environment. In Swift in the Cloud, you'll find full coverage of all aspects of creating and running Swift language applications in Cloud computing environments, complete with examples of real code that you can start running and experimenting with today. Since Apple introduced the Swift language in 2014, it has become one of the most rapidly adopted computer programming languages in history—and now you too can start benefitting from using the same programming language for all components of a scalable, robust business software solution. Create server applications using Swift and run them on pay-as-you-go cloud infrastructure Quickly write and test Swift code snippets in your own cloud sandbox Use Docker containers to deploy Swift applications into multiple cloud environments without having to change code Grasp the elements and structure of the Swift.org open technology project Find out how to avoid the complexities of runtime configuration by using Cloud Foundry buildpacks for Swift Build high performing web applications and REST APIs with an open source Swift based web server framework Scale up your cloud services by running Swift modules in an asynchronous, open source, 'serverless' cloud environment Whether you are already using Swift to build mobile applications or a seasoned web developer, Swift in the Cloud will help you leverage server-side Swift to power your next generation of applications.

This book constitutes the refereed proceedings of the 12th International Symposium on Practical Aspects of Declarative Languages, PADL 2010, held in Madrid, Spain, in January 2010, colocated with POPL 2010, the Symposium on Principles of Programming Languages. The 22 revised full papers presented together with 2 invited talks were carefully reviewed and selected from 58 submissions. The volume features original work emphasizing novel applications and implementation techniques for all forms of declarative concepts, including functions, relations, logic, and constraints. The papers address all current aspects of declarative programming; they are organized in topical sections on non-monotonic reasoning - answer set programming, types, parallelism and distribution, code quality assurance, domain specific languages, programming aids, constraints, and tabling - agents.

Learn how to build playgrounds so you can test your code, syntax, and ideas quickly. You can even learn from playgrounds built by others or build playgrounds to teach. And the playgrounds you build and use on your Mac and on your iPad are automatically shared using your Apple ID. Exploring Swift Playgrounds shows you how to use playgrounds to try out your basic app design ideas to see what they look like and how they behave. It doesn't matter if you can't remember a pesky little bit of syntax. Rather than look it up, you can try it out in a playground. More and more of the APIs are now available through playgrounds, so that you can do more than ever before. Going beyond print, the Swift Playgrounds book tool allows for immersive experiences for users learning code, organization processes, and anything else that can be described in the rich (and free) authoring tools provided by Apple. All of which you can learn how to use in Exploring Swift Playgrounds. What You'll Learn: Build Swift playgrounds for others to use Teach yourself and others with Swift playgrounds Use Swift playgrounds in your development process Who This Book Is For: Students with no prior coding knowledge and experienced developers. Discover a contemporary overview of today's computer science with Schneider/Gersting's best-selling INVITATION TO COMPUTER SCIENCE, 8E. This flexible, non-language-specific approach provides a solid foundation in computer science using an algorithm-centered approach that's ideal for the reader's first introduction to the field. Measurable learning objectives and an easy-to-follow hierarchy guide readers through algorithms, hardware, virtual machines, software development, applications of computing, and social issues. Readers connect the dots as the book emphasizes real-life context throughout each chapter. Updates introduce the latest developments concerning privacy, drones, cloud computing, and net neutrality. Visual and hands-on activities let readers experience the fundamentals of today's computer science. Important Notice: Media content referenced within the product description or the product text may not be available in the ebook version.

Develop the skills required to create compelling, maintainable, and robust iOS and OS X apps with Swift About This Book Write expressive, understandable, and maintainable Swift 2 code with this hands-on tutorial Unveil the complex underpinnings of Swift to turn your app ideas into reality This book is packed with real-life examples to help you

implement concepts as you learn Who This Book Is For If you are looking to build iOS or OS X apps using the most modern technology, this book is ideal for you. You will find this book especially useful if you are new to programming or if you are yet to develop for iOS or OS X. No prior programming exposure is required. What You Will Learn Form a solid understanding of the Swift 2 language Get to know the practical aspects of how a computer program actually works Understand the paradigms used by Apple's frameworks so you are not intimidated by them Utilize the vast resources written in Objective-C to better inform your Swift programming Develop a basic portfolio of Swift code by learning the critical concepts Experience both object-oriented and functional programming Get to know the new coding techniques made available by Swift 2 Discover resources to ensure you never stop becoming a better developer In Detail Swift is Apple's new programming language and the future of iOS and OS X app development. It is a high-performance language that feels like a modern scripting language. On the surface, Swift is easy to jump into, but it has complex underpinnings that are critical to becoming proficient at turning an idea into reality. This book is an approachable, step-by-step introduction into programming with Swift for everyone. It begins by giving you an overview of the key features through practical examples and progresses to more advanced topics that help differentiate the proficient developers from the mediocre ones. It covers important concepts such as Variables, Optionals, Closures, Generics, and Memory Management. Mixed in with those concepts, it also helps you learn the art of programming such as maintainability, useful design patterns, and resources to further your knowledge. This all culminates in writing a basic iOS app that will get you well on your way to turning your own app ideas into reality. Style and approach This is an approachable, step-by-step guide to programming in Swift 2. Each topic is separated into compressible sections that are full of practical examples and easy-to-understand explanations. Each section builds on the previous topics so you can develop a proficient and comprehensive understanding of app development in Swift 2.

And ConclusionChapter 2. Functions; Function Parameters and Return Value; Void Return Type and Parameters; Function Signature; External Parameter Names; Overloading; Default Parameter Values; Variadic Parameters; Ignored Parameters; Modifiable Parameters; Function In Function; Recursion; Function As Value; Anonymous Functions; Define-and-Call; Closures; How Closures Improve Code; Function Returning Function; Closure Setting a Captured Variable; Closure Preserving Its Captured Environment; Curried Functions; Chapter 3. Variables and Simple Types; Variable Scope and Lifetime.

Bring the power of functional programming to Swift to develop clean, smart, scalable and reliable applications. About This Book Written for the latest version of Swift, this is a comprehensive guide that introduces iOS, Web and macOS developers to the all-new world of functional programming that has so far been alien to them Get familiar with using functional programming alongside existing OOP techniques so you can get the best of both worlds and develop clean, robust, and scalable code Develop a case study on example backend API with Swift and Vapor Framework and an iOS application with Functional Programming, Protocol-Oriented Programming, Functional Reactive Programming, and Object-Oriented Programming techniques Who This Book Is For Meant for a reader who knows object-oriented programming, has some experience with Objective-C/Swift programming languages and wants to further enhance his skills with functional programming techniques with Swift 3.x. What You Will Learn Understand what functional programming is and why it matters Understand custom operators, function composition, currying, recursion, and memoization Explore algebraic data types, pattern matching, generics, associated type protocols, and type erasure Get acquainted with higher-kinded types and higher-order functions using practical examples Get familiar with functional and non-functional ways to deal with optionals Make use of functional data structures such as semigroup, monoid, binary search tree, linked list, stack, and lazy list Understand the importance of immutability, copy constructors, and lenses Develop a backend API with Vapor Create an iOS app by combining FP, OOP, FRP, and POP paradigms In Detail Swift is a multi-paradigm programming language enabling you to tackle different problems in various ways. Understanding each paradigm and knowing when and how to utilize and combine them can lead to a better code base. Functional programming (FP) is an important paradigm that empowers us with declarative development and makes applications more suitable for testing, as well as performant and elegant. This book aims to simplify the FP paradigms, making them easily understandable and usable, by showing you how to solve many of your day-to-day development problems using Swift FP. It starts with the basics of FP, and you will go through all the core concepts of Swift and the building blocks of FP. You will also go through important aspects, such as function composition and currying, custom operator definition, monads, functors, applicative functors, memoization, lenses, algebraic data types, type erasure, functional data structures, functional reactive programming (FRP), and protocol-oriented programming (POP). You will then learn to combine those techniques to develop a fully functional iOS application from scratch Style and approach An easy-to-follow guide that is full of hands-on coding examples of real-world applications. Each topic is explained sequentially and placed in context, and for the more inquisitive, there are more details of the concepts used. It introduces the Swift language basics and functional programming techniques in simple, non-mathematical vocabulary with examples in Swift.

Swift for the Really Impatient is the perfect jumpstart to the Swift language for every experienced Objective-C developer. Leading iOS and OS X developers Matt Henderson and Dave Wood help you take full advantage of Swift's powerful innovations, without wasting time on basics you already know. Through expert code examples and crystal-clear explanations, Henderson and Wood reveal how Swift features improve on Objective-C and show how to make the most of them. You'll quickly master "Swift-er" techniques for using objects, classes, optionals, generics, functions, closures, and more. Each chapter includes hands-on exercises designed to reinforce and deepen your skills. You'll learn how to re-implement common Objective-C programming solutions in Swift, improving code clarity, power, and performance on both desktop and mobile devices. And, because you probably won't (or can't) abandon Objective-C anytime soon, you'll discover how to use both languages together, smoothly integrate Swift into existing projects and

workflows, and gradually move your current code base into the future. Coverage includes Using Swift's new patterns and concise, expressive syntax to produce safer, more reliable code Leveraging the surprising power of Swift's data structures Utilizing Swift's type system to help prevent errors common in other languages Using optionals to correctly handle invalid, missing, or unknown values Implementing generics to reduce duplication, improve power, and simplify maintenance Interacting with C and Objective-C APIs Abstracting functionality for reusable code while maintaining type safety Isolating code in clean, flexible, low-overhead closures Combining Swift with Objective-C in the same project Avoiding gotchas when importing Objective-C classes in Swift projects Nesting types to improve code clarity Creating shared objects with singletons Managing threading and concurrency with Swift's advanced Grand Central Dispatch (GCD) patterns

This book presents the fundamentals of mobile visual computing in iOS development and provides directions for developers and researchers interested in developing iOS applications with image processing and computer vision capabilities. Presenting a technical overview of some of the tools, languages, libraries, frameworks, and APIs currently available for developing iOS applications Image Processing and Computer Vision in iOS reveals the rich capabilities in image processing and computer vision. Its main goal is to provide a road map to what is currently available, and a path to successfully tackle this rather complex but highly rewarding task. .

Bring the power of Swift functional programming to iOS, Web, macOS, watchOS and tvOS application development and build clean, smart, scalable and reliable applications About This Book- Written for Swift 3 -Developers Preview version, this is a comprehensive guide that introduces iOS and OS X developers to the all-new world of functional programming that has so far been alien to them- Learn about first-class functions and how imperative-style patterns can be converted into functional code using some simple techniques- The book will get you familiar with using functional programming alongside existing OOP techniques so you can get the best of both worlds and develop clean, robust code Who This Book Is For The book is for developers with a basic knowledge of Swift programming aiming to incorporate functional programming paradigms in their day-to-day application development What You Will Learn- First-class, higher-order, and pure functions- Closures and capturing values- Custom operators, recursion, and memoization- Value and reference types in Swift- Enumerations, algebraic data types, patterns, and pattern matching- Generics and associated type protocols- Higher-order functions such as map, flatMap filter, and reduce- Dealing with optionals, fmap, and apply for multiple functional mapping- Functional data structures such as Semigroup, Monoid, Binary Search Tree, Linked List, Stack, and Lazy List- Immutability, copy constructors, and lenses- Combining FP paradigms with OOP, FRP, and POP in your day-to-day development activities- Developing a backend application with Swift- Developing an iOS application with FP, OOP, FRP, and POP paradigms In Detail This book is based on Swift 3 Developer preview version and aims at simplifying the functional programming (FP) paradigms making it easily usable, by showing you how to solve many of your day-to-day development problems. Whether you are new to functional programming and Swift or experienced, this book will strengthen the skills you need to design and develop high-quality, scalable, and efficient applications. The book starts with functional programming concepts, the basics of Swift 3, and essential concepts such as functions, closures, optionals, enumerations, immutability, and generics in detail with coding examples. Furthermore, this book introduces more advanced topics such as function composition, monads, functors, applicative functors, memoization, lenses, algebraic data types, functional data structures, functional reactive programming (FRP), protocol-oriented programming (POP) and mixing object-oriented programming (OOP) with functional programming (FP) paradigms. Finally, this book provides a working code example of a front-end application developed with these techniques and its corresponding back-end application developed with Swift. Style and approach This is an easy-to-follow guide full of hands-on coding examples of real-world applications. Each topic is explained sequentially and placed in context, and for the more inquisitive, there are more details of the concepts used. It introduces the Swift language basics and functional programming techniques in simple, non-mathematical vocabulary with examples in Swift.

Kotlin is a powerful and pragmatic language, but it's not enough to know about its features. We also need to know when they should be used and in what way. This book is a guide for Kotlin developers on how to become excellent Kotlin developers. It presents and explains in-depth the best practices for Kotlin development. Each item is presented as a clear rule of thumb, supported by detailed explanations and practical examples.

The three-volume set LNCS 10918, 10919, and 10290 constitutes the proceedings of the 7th International Conference on Design, User Experience, and Usability, DUXU 2018, held as part of the 20th International Conference on Human-Computer Interaction, HCII 2018, in Las Vegas, NV, USA in July 2018. The total of 1171 papers presented at the HCII 2018 conferences were carefully reviewed and selected from 4346 submissions. The papers cover the entire field of human-computer interaction, addressing major advances in knowledge and effective use of computers in a variety of applications areas. The total of 165 contributions included in the DUXU proceedings were carefully reviewed and selected for inclusion in this three-volume set. The 60 papers included in this volume are organized in topical sections on emotion, motivation, and persuasion design, DUXU and children, DUXU in automotive and transport, and DUXU in culture and art.

This book constitutes the thoroughly refereed post-conference proceedings of the 23rd International Symposium on Implementation and Application of Functional Languages, IFL 2011, held in Lawrence, Kansas, USA, in October 2011. The 11 revised full papers presented were carefully reviewed and selected from 33 submissions. The papers by researchers and practitioners who are actively engaged in the implementation and the use of functional and function based programming languages describe practical and theoretical work as well as applications and tools. They discuss new ideas and concepts, as well as work in progress and results.

This book contains the refereed proceedings of the 23rd International Conference on Logic Programming, ICLP 2007, held in Porto, Portugal. The 22 revised full papers together with two invited talks, 15 poster presentations, and the abstracts of five doctoral consortium articles cover all issues of current research in logic programming, including theory, functional and constraint logic programming, program

analysis, answer-set programming, semantics, and applications.

This book presents a selection of papers from the 2017 World Conference on Information Systems and Technologies (WorldCIST'17), held between the 11st and 13th of April 2017 at Porto Santo Island, Madeira, Portugal. WorldCIST is a global forum for researchers and practitioners to present and discuss recent results and innovations, current trends, professional experiences and challenges involved in modern Information Systems and Technologies research, together with technological developments and applications. The main topics covered are: Information and Knowledge Management; Organizational Models and Information Systems; Software and Systems Modeling; Software Systems, Architectures, Applications and Tools; Multimedia Systems and Applications; Computer Networks, Mobility and Pervasive Systems; Intelligent and Decision Support Systems; Big Data Analytics and Applications; Human-Computer Interaction; Ethics, Computers & Security; Health Informatics; Information Technologies in Education; and Information Technologies in Radiocommunications.

Create robust and extensible iOS apps using the advanced features of Swift 2 About This Book Get up to speed with the new features of Swift 2 by following the exhaustive examples in this book Specialize in developing real iOS apps, and 2D and 3D videogames using Swift and Cocos2d Learn how to build server API apps to feed your iOS client apps Who This Book Is For This book is ideal for those who want to learn to develop app in Swift, starting the right way. Whether you are an expert Objective-C programmer or are new to this platform, you'll quickly grasp the code of real world apps, and discover how to use Swift effectively. Prior experience in development for Apple devices would be helpful, but is not mandatory. What You Will Learn Create a server in Swift to deliver JSON data to an iOS app Take advantage of Cocos2d to use third-party libraries Use a clean and effective architecture to decrease complexity and speed up development Take advantage of the most useful parts of the iOS SDK Build games with SpriteKit and SceneKit Develop an app running on the cloud to act as an API server for your client's apps In Detail Swift is no longer the unripe language it was when launched by Apple at WWDC14, now it's a powerful and ready-for-production programming language that has empowered most new released apps. Swift is a user-friendly language with a smooth learning curve; it is safe, robust, and really flexible. Swift 2 is more powerful than ever; it introduces new ways to solve old problems, more robust error handling, and a new programming paradigm that favours composition over inheritance. Swift 2 by Example is a fast-paced, practical guide to help you learn how to develop iOS apps using Swift. Through the development of seven different iOS apps and one server app, you'll find out how to use either the right feature of the language or the right tool to solve a given problem. We begin by introducing you to the latest features of Swift 2, further kick-starting your app development journey by building a guessing game app, followed by a memory game. It doesn't end there, with a few more apps in store for you: a to-do list, a beautiful weather app, two games: Flappy Swift and Cube Runner, and finally an ecommerce app to top everything off. By the end of the book, you'll be able to build well-designed apps, effectively use AutoLayout, develop videogames, and build server apps. Style and approach These easy-to-follow tutorials show you how to build real-world apps. The difficulty and complexity level increases chapter by chapter. Each chapter is dedicated to build a new app, beginning from a basic and unstyled app through to a full 3D game. The last two chapters show you how to build a complete client-server ecommerce app right from scratch.

This book constitutes the refereed proceedings of the 11th International Symposium on Practical Aspects of Declarative Languages, PADL 2009, held in Savannah, GA, USA, in January 2009, colocated with POPL 2009, the Symposium on Principles of Programming Languages. The 18 revised full papers presented together with 1 invited talk were carefully reviewed and selected from 48 submissions. The volume features original work emphasizing novel applications and implementation techniques for all forms of declarative concepts, including functions, relations, logic, and constraints. The papers address all current aspects of declarative programming; they are organized in topical sections on user interfaces and environments, networks and data, multi-threading and parallelism, databases and large data sets, tabling and optimization, as well as language extensions and implementation.

Conquer complex and interesting programming challenges by building robust and concurrent applications with caches, cryptography, and parallel programming. Key Features Understand how to use .NET frameworks like the Task Parallel Library (TPL) and CryptoAPI Develop a containerized application based on microservices architecture Gain insights into memory management techniques in .NET Core Book Description This Learning Path shows you how to create high performing applications and solve programming challenges using a wide range of C# features. You'll begin by learning how to identify the bottlenecks in writing programs, highlight common performance pitfalls, and apply strategies to detect and resolve these issues early. You'll also study the importance of micro-services architecture for building fast applications and implementing resiliency and security in .NET Core. Then, you'll study the importance of defining and testing boundaries, abstracting away third-party code, and working with different types of test double, such as spies, mocks, and fakes. In addition to describing programming trade-offs, this Learning Path will also help you build a useful toolkit of techniques, including value caching, statistical analysis, and geometric algorithms. This Learning Path includes content from the following Packt products: C# 7 and .NET Core 2.0 High Performance by Ovais Mehboob Ahmed Khan Practical Test-Driven Development using C# 7 by John Callaway, Clayton Hunt The Modern C# Challenge by Rod Stephens What you will learn Measure application performance using BenchmarkDotNet Leverage the Task Parallel Library (TPL) and Parallel Language Integrated Query (PLINQ) library to perform asynchronous operations Modify a legacy application to make it testable Use LINQ and PLINQ to search directories for files matching patterns Find areas of polygons using geometric operations Randomize arrays and lists with extension methods Use cryptographic techniques to encrypt and decrypt strings and files Who this book is for If you want to improve the speed of your code and optimize the performance of your applications, or are simply looking for a practical resource on test driven development, this is the ideal Learning Path for you. Some familiarity with C# and .NET will be beneficial.

This book will teach you how to use Swift to apply functional programming techniques to your iOS or OS X projects. These techniques complement object-oriented programming that most Objective-C developers will already be familiar with, providing you with a valuable new tool in your developer's toolbox. We will start by taking a look at Swift's new language features, such as higher-order functions, generics, optionals, enumerations, and pattern matching. Mastering these new features will enable you to write functional code effectively. After that, we will provide several examples of how to use functional programming patterns to solve real-world problems. These examples include a compositional and type-safe API around Core Image, a library for diagrams built on Core Graphics, and a small spreadsheet application built from scratch.

Handbook of Automated Reasoning.

This book constitutes the thoroughly refereed post-conference proceedings of the 11th International Symposium on Trends in Functional Programming, TFP 2010, held in Norman, OK, USA, in May 2010. The 13 revised full papers presented were carefully reviewed and selected from 26 submissions during two rounds of reviewing and improvement. The papers cover new ideas for refactoring, managing source-code complexity, functional language implementation, graphical languages, applications of functional programming in pure mathematics, type theory, multitasking and parallel processing, distributed systems, scientific modeling, domain specific languages, hardware design, education, and testing.

Swift Functional ProgrammingPackt Publishing Ltd

Get ahead of everyone else and learn the latest technologies introduced by Apple. This is the first book to teach you how to work with Swift 3, Xcode 8, iOS 10 and the new APIs. iOS Apps for Masterminds leads the reader step by step to master the complex subjects required to create applications for iPhones and iPads. After reading this book, you will know how to program in Swift, how to design user interfaces, and how to work with the most powerful frameworks available for the construction of modern applications. This book is a complete course that will teach you how to build insanely great applications from scratch. Every chapter explores both basic and complicated concepts of computer programming, the Swift language, and app development. The information is supported by fully functional examples to guide beginners and experts through every single framework included in the iOS SDK. The examples are distributed throughout the book in a specific order to gradually introduce complex topics and make them accessible to everyone. The goal of iOS Apps for Masterminds is to make you familiar with the most advanced technologies for app development. It was designed to prepare you for the future and was written for the genius inside you, for Masterminds. This book includes: Introduction to Swift 3 Swift Paradigm Foundation Framework UIKit Framework Auto Layout Size Classes Navigation Controllers Scroll Views Table Views Collection Views Split View Controller Alert Views Notifications Files Archiving Core Data iCloud Core Graphics and Quartz 2D Core Animation AVFoundation Camera and Photos Library Web Views Contacts Sensors MapKit Gesture Recognizers Timers Operation Queues Error Handling Image and Video Internationalization ...and more! iOS app development with iOS 10, Xcode 8 and Swift 3 App development, Swift programming, Create apps, Create app, iPhone apps, Build app, Swift language, develop application, Objective-C, Apple development, iOS development, iOS Apps, Program apps.

This reader-friendly textbook presents a concise and easy to follow introduction to Scala. Scala is an ideal first programming language, which permits programming in multiple paradigms, and enables developers to be more productive with modern computing infrastructures such as distributed environments. Topics and features: provides review questions and problem-solving exercises (with solutions) in each chapter, inspired by real-world applications; addresses each topic in a self-contained manner, highlighting how Scala can be evolved and grown according to the developer's needs; presents examples from a broad range of different application domains, including consumer electronics, online payment, retail, vehicle manufacturing, and healthcare; encourages an innovation-oriented mind-set, and the development of practical, saleable skills; draws from the author's extensive experience in industrial software development, academic research, and university teaching. This accessible and hands-on guide will embolden professional software engineers to make the switch to Scala. Instructors teaching introductory programming courses will also find this textbook popular among their students.

Move into iOS development by getting a firm grasp of its fundamentals, including the Xcode IDE, the Cocoa Touch framework, and Swift 3—the latest version of Apple's acclaimed programming language. With this thoroughly updated guide, you'll learn Swift's object-oriented concepts, understand how to use Apple's development tools, and discover how Cocoa provides the underlying functionality iOS apps need to have. Explore Swift's object-oriented concepts: variables and functions, scopes and namespaces, object types and instances Become familiar with built-in Swift types such as numbers, strings, ranges, tuples, Optionals, arrays, dictionaries, and sets Learn how to declare, instantiate, and customize Swift object types: enums, structs, and classes Discover powerful Swift features such as protocols and generics Catch up on Swift 3 innovations: revised APIs, new Foundation bridged types, and more Tour the lifecycle of an Xcode project from inception to App Store—including Xcode's new automatic code signing and debugging features Construct app interfaces with the nib editor, Interface Builder Understand Cocoa's event-driven model and its major design patterns and features Find out how Swift communicates with Cocoa's C and Objective-C APIs Once you master the fundamentals, you'll be ready to tackle the details of iOS app development with author Matt Neuburg's companion guide, Programming iOS 10.

The two volume set LNCS 6415 and LNCS 6416 constitutes the refereed proceedings of the 4th International Symposium on Leveraging Applications of Formal Methods, ISoLA 2010, held in Heraklion, Crete, Greece, in October 2010. The 100 revised full papers presented were carefully revised and selected from numerous submissions and discuss issues related to the adoption and use of rigorous tools and methods for the specification, analysis, verification, certification, construction, test, and maintenance of systems. The 46 papers of the first volume are organized in topical sections on new challenges in the development of critical embedded systems, formal languages and methods for designing and verifying complex embedded systems, worst-case traversal time (WCTT), tools in scientific workflow composition, emerging services and technologies for a converging telecommunications / Web world in smart environments of the internet of things, Web science, model transformation and analysis for industrial scale validation, and learning techniques for software verification and validation. The second volume presents 54 papers addressing the following topics: EternalS: mission and roadmap, formal methods in model-driven development for service-oriented and cloud computing, quantitative verification in practice, CONNECT: status and plans, certification of software-driven medical devices, modeling and formalizing industrial software for verification, validation and certification, and resource and timing analysis.

Learn How To Program with Swift 3! Swift is the easiest way to get started developing on Apple's platforms: iOS, OS X, watchOS and tvOS. With the release of Swift 3 in 2016, the Swift language is packed with even more features and enhancements. In this book, you'll learn the basics of Swift from getting started with playgrounds to simple operations to building your own types. Everything you'll learn is platform-neutral; you'll have a firm understanding of Swift by the end of this book, and you'll be ready to move on to whichever app platform you're interested in. Who This Book Is For: This book is for complete beginners to Swift 3. No prior programming experience is necessary! Topics Covered in The Swift Apprentice Playground basics: Learn about the coding environment where you can quickly and easily try out your code as you learn. Numbers and strings:

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

Take your macOS Sierra to the next level using the latest tools, designs, and best coding practices while developing with Swift 3.0 About This Book Learn to harness the power of macOS with the elegance of the Swift programming language Become highly competent in building apps on the macOS platform Get the most in-depth guide with a hands-on approach on the latest version of macOS Who This Book Is For This book is for developers who have some experience with macOS and want to take their skills to next level by unlocking the full potential of latest version of macOS with Swift 3 to build impressive applications. Basic knowledge of Swift will be beneficial but is not required. What You Will Learn Combine beautiful design with robust code for the very best user experience Bring the best coding practices to the new macOS Sierra See what's new in Swift 3.0 and how best to leverage the Swift language Master Apple's tools, including Xcode, Interface Builder, and Instruments Use Unix and other common command-line tools to increase productivity Explore the essential Cocoa frameworks, including networking, animation, audio, and video In Detail macOS continues to lead the way in desktop operating systems, with its tight integration across the Apple ecosystem of platforms and devices. With this book, you will get an in-depth knowledge of working on macOS, enabling you to unleash the full potential of the latest version using Swift 3 to build applications. This book will help you broaden your horizons by taking your programming skills to next level. The initial chapters will show you all about the environment that surrounds a developer at the start of a project. It introduces you to the new features that Swift 3 and Xcode 8 offers and also covers the common design patterns that you need to know for planning anything more than trivial projects. You will then learn the advanced Swift programming concepts, including memory management, generics, protocol orientated and functional programming and with this knowledge you will be able to tackle the next several chapters that deal with Apple's own Cocoa frameworks. It also covers AppKit, Foundation, and Core Data in detail which is a part of the Cocoa umbrella framework. The rest of the book will cover the challenges posed by asynchronous programming, error handling, debugging, and many other areas that are an indispensable part of producing software in a professional environment. By the end of this book, you will be well acquainted with Swift, Cocoa, and AppKit, as well as a plethora of other essential tools, and you will be ready to tackle much more complex and advanced software projects. Style and approach This comprehensive guide takes a hands-on practical approach incorporating a visually-rich format rather than a text heavy format. The focus is on teaching the core concepts through a series of small projects and standalone examples so you gain expertise with various aspects of macOS application development.

Get valuable hands-on experience with Swift 3, the latest version of Apple's programming language. With this practical guide, skilled programmers with little or no knowledge of Apple development will learn how to code with Swift 3 by developing three complete, tightly linked versions of the Notes application for the OS X, iOS, and watchOS platforms. In the process, you'll learn Swift's fundamentals, including its syntax and features, along with the basics of the Cocoa, CocoaTouch, and WatchKit frameworks. This book teaches you how to use common design patterns for Swift, how to structure an application for Apple's platforms, and how to submit working apps to the App Store. Divided into four distinct parts, this book includes: Swift 2 basics: Learn Swift's basic building blocks and features for object-oriented development OS X app development: Set up the document model, build out features, and sync data with iCloud iOS app development: Use multimedia, contacts, location, notifications, and iCloud files to build a fully featured iOS Notes app Advanced app extensions: Build an Apple Watch app, and learn how to debug, monitor, and test all three of your Swift apps

[Copyright: 70e8a27673901fd6fca946bf51ce72ce](#)