

GETTING STARTED IN IOS DEVELOPMENT IN SWIFT



XIAOANG ZHENG

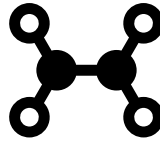
TABLE OF CONTENTS

Preface	6
CHAPTER ONE : INTRODUCTION TO SWIFT AND XCODE.....	7
I. Swift and iOS	7
II. what should you learn to build an app?.....	8
CHAPTER TWO : prepare for HELLO WORLD APP	9
I. what hardware do i need?.....	9
II. Get an Apple id	10
III. Install & configure xcode.....	11
IV. Start your first Xcode project.....	13
V. Get To Know The Working Interface	16
VI. Run Your App For The First Time	17
Exercise #1	18
Exercise #2	19
SUMMARY	19
CHAPTER THREE : XCODE PROJECT AND FILES.....	20
I. PROJECT STRUCTURE	20
II. STORYBOARDS	21
III. Swift Files.....	23
IV. OTHER SUPPORTING Files	23
Exercise #1	25
Exercise #2.....	26
CHAPTER FOUR : USER INTERFACE OF THE HELLO WORLD APP	27
I. Overview of the function	27
II. Modify Main.storyboard	28

III. Link The Button To ViewController	32
IV. Test The Modifications	35
Exercise #1	36
Exercise #2.....	36
SUMMARY	37
CHAPTER Five : documentation and basic swift	38
I.Overview of ViewController.swift	38
II.A Swift Tour And Playground.....	40
III. Simple Values	41
IV. Control Flow	44
V. Functions and Closures	47
Exercise #1	50
Exercise #2.....	50
SUMMARY	51
CHAPTER SIX : INTRODUCTION TO DESIGN PATTERN MVC	52
I. Overview of The MVC pattern	52
II. Apply MVC To Hello World App: VIEW	53
III. Apply MVC To Hello World App: MODEL	57
VI. Apply MVC To Hello World App: CONTROLLER.....	57
V. Further understanding on MVC	60
SUMMARY.....	63
CHAPTER SEVEN : OOP AND DATA STRUCTURES IN SWIFT	64
I.INTRODUCTION	64
II. OBJECTS AND CLASSES.....	64
III. ENUMERATIONS AND STRUCTURES	68
IV. PROTOCOLS AND EXTENSIONS.....	70
V. ERROR HANDLING	72
vi. generics	74
SUMMARY.....	75
CHAPTER EIGHT : POWER FOCUS PROJECT overview.....	76
I. Introduction	76
II. Function Overview.....	76

III. Interface and prototyping	77
IV. Data Structure	78
SUMMARY	79
CHAPTER NINE : CREATE MAIN STORYBOARD.....	81
I. CREATE A NEW PROJECT	81
II. Set Up the Storyboard	83
III. Add Widgets to the Storyboard	84
IV. Image View And Use Assets in Storyboards.....	89
V. Image View and Image in Storyboards.....	90
Exercise #1	94
Exercise #2	94
SUMMARY	96
CHAPTER TEN : CREATE TIMER MODEL.....	97
I. Introduction	97
II. SPECIFICATION.....	98
III. Timer Class Diagram	98
IV. Enumeration ActivityType	99
V. Create Class FocusTimer and Attributes.....	101
V. FocusTimer's_INITIALIZER.....	104
VI. Static Attributes.....	104
VI. Exercise.....	106
Exercise #1	107
Exercise #2.....	108
SUMMARY.....	110
CHAPTER ELEVEN : UNIT TEST OF THE MODEL	111
I. Introduction to SOFTWARE TESTING	111
II. CREATE Unit testS in Xcode	112
III. Basics of Unit Testing	114
IV. Test FocusTimer.....	118
V. UI Tests	121
Exercise	122
SUMMARY	123

CHAPTER Twelve : Communications between Controller and view	124
I. Role of Controller in MVC.....	124
II. Simple binding	125
Exercise	130
SUMMARY	132
CHAPTER Thirteen : Controller Behaviors.....	133
I. Controller behavior and model update	133
II. PowerFocous Timer	135
III. Timer handler and validation.....	137
IV. Update the model	139
Exercise	141
SUMMARY	142
CHAPTER Fourteen : Manage MODEL and Interaction.....	143
I. Introduction	143



PREFACE

Do you enjoy using iPhone and all the amazing apps in the Apple AppStore? If your answer is yes and you want to build your own app, this book is perfect for you to get your programming journey started!

This book will cover everything you need to learn to build your own app and for iOS app development. If you are an absolute beginner with no programming background or experience, don't worry! In my opinion, one of the beauties about learning mobile app developments compared to the hard core computer science is that you don't really learn until you get your hands dirty and really build something of your own. So if you are a beginner, just follow the instructions step by step, and I'm sure that your app is going to be awesome little by little.

In this book, I will show you step by step how to build an app using Xcode and other tools that Apple provides to its developers. If you want to build your own apps with your app ideas, you don't need to finish the whole book to get started. The first half of this book is about some basic concepts and approaches, which you are going to need no matter what you're trying to build. However, the second half is about the cool features that are introduced to iOS 11 and 12 which can make your app look and feel better. So if you want to add a certain feature to your own app, just jump right to the corresponding chapter and follow the template.

Can't wait to get started? Let's build your first app!

CHAPTER ONE : INTRODUCTION TO SWIFT AND XCODE



I. SWIFT AND IOS

If you have experience in C++, Java, or Python programming, you probably have noticed that your program can run in many platforms: Windows, MacOS, Linux... A powerful high level programming language of all purpose has a lot of powerful tools provided by many different parties over the years.

In the iOS platform, however, you can only use Swift and objective C to write your own app. And your code can only run on a iOS device: iPhone, iPad, Apple Watch... So if Swift is so limited, why are we still using Swift to build our own app?

First introduced by Apple in June 2014, Swift has gone through several major updates. It is designed for iOS, watchOS, and TVOS. Swift is advertised as a "fast, modern, safe,

interactive" programming language. The language is easier to learn and comes with features to make programming more productive.

Before Swift, most iOS apps were written in Objective-C. This language was chosen by Apple as the primary programming language for Mac and iOS development. Compared to Objective-C, the syntax is much cleaner and easier to read and write, and more importantly it definitely speeds up the development process. And if you're a total beginner with no prior programming experience, you'll also find the language friendlier and feel more comfortable to develop apps in Swift.

II. WHAT SHOULD YOU LEARN TO BUILD AN APP?

Now you have already known that iOS, Xcode, and Swift work together as a whole to make app development easier for the Apple developers, you may wonder what exactly is the outline of the things that you need learn. In general, it comes down to these topics:

- **Swift:** This is the programming language that you are going to use throughout this tutorial, so you'll learn the basic syntax, the features, and the common practices of this language.
- **Xcode:** This is the environment that you'll have to use to development your own app. You can be really efficient and your workflow can be very pleasant if you are skillful with Xcode.
- **Understand the iOS development kit:** Like other programming languages, the language provider provides many useful tools and features to assist its developers so that the development in that language can be user-friendly, robust, and efficient. This kit comes with a set of software tools and APIs that empowers you to develop iOS apps. For example, if you want to display a web page in your app, the SDK provides a built-in browser that lets you embed right in your application.