



The iOS Tutorial Beginner to Pro

Jonathan Rasmusson

iOS Tutorial Book

From Beginner to Pro

Jonathan Rasmusson

Contents

Preface	v
What's in the book?	vii
Work in progress	ix
1 Setting up Xcode	1
1.1 Checking your OS	1
1.2 Downloading Xcode	4
1.2.1 The App Store	4
1.2.2 From the Internet	5
1.3 Conclusion	10
1.3.1 What we learned in this chapter	10
1.4 Feedback	10
2 Hello World	13
2.1 Creating a new project from scratch	13
2.2 Getting familiar with Xcode	18
2.3 Running on the simulator	20
2.4 Building the Basic UI	23
2.5 Running on the iPhone (Optional)	29
2.5.1 Check iPhone OS version number	30
2.5.2 Add your Apple developer account	33
2.5.3 Sign the app	35
2.5.4 Connect your physical device	37

2.5.5	Trust this app.	38
2.5.6	Trouble Shooting	50
2.6	Conclusion	50
2.6.1	What we learned in this chapter	51
2.7	Feedback	51
3	I Love Coffee	53
3.1	Download the source	55
3.2	Create a new project	57
3.3	Adding Image Assets	60
3.4	Labels and coordinate systems	70
3.5	Auto Layout	82
3.5.1	Pinning	86
3.5.2	Aligning	99
3.6	Fixing our design	104
3.7	Conclusion	120
3.7.1	What we learned this chapter	121
3.8	Feedback	121
4	Magic 8-Ball	123
4.1	The plan	125
4.2	Create a new project	127
4.3	Add the image assets	129
4.4	Add our UI controls	130
4.4.1	Style the controls	132
4.5	Doing the Auto Layout	137
4.5.1	Positioning the label	140
4.5.2	Laying out the image	142
4.5.3	Positioning the Button	152
4.6	Making it interactive	159
4.6.1	Setting up our View Controller	160
4.6.2	Reacting to actions	170
4.6.3	Creating a data structure to holding our images	173
4.6.4	Randomly selecting an image	182

4.6.5	Updating the image view	185
4.7	Trouble Shooting	186
4.7.1	Dangling outlets and actions	186
4.8	Conclusion	189
4.8.1	What we learned in this chapter	190
4.9	Feedback	190
5	Swift basics	191
5.1	Getting started	191
5.2	Swift Playgrounds	193
5.3	Constants	197
5.4	Variables	197
5.5	Properties	197
5.5.1	Stored properties	198
5.5.2	Computed properties	198
5.6	Optionals	199
5.6.1	Unwrapping with if	200
5.6.2	Unwrapping with guard	200
5.6.3	What's the difference	201
5.7	Structures and Classes	201
5.7.1	Structures and Enumerations are value types	202
5.7.2	Classes are reference types	203
5.7.3	Structs vs Classes	204
5.8	Enums	205
5.9	Functions	206
5.10	Conclusion	207
5.11	What we learned	208
6	Weathery	209
6.1	What you are going to build	209
7	No storyboards	213
7.1	Why no storyboards	214
7.2	Creating a new project	215

7.3	Improving productivity with Code Snippets	223
7.4	Conclusion	230
8	Auto Layout	233
8.1	What is Auto Layout	234
8.2	How does it work	236
8.3	Pinning	238
8.4	Aligning	241
8.5	The SafeAreaLayout Guide	244
8.6	Intrinsic content size	254
8.7	Resolving ambiguity with CHCR	256
8.8	Stack Views	264
8.9	Where to go from here?	268
8.10	Conclusion	269
9	Vector assets	271
9.1	The problem with pngs	272
9.2	Enter vector based graphics	274
9.3	Supporting vectors based images in Xcode	275
9.4	Using vector based images in our project	280
9.5	Conclusion	285
9.6	What we learned	285
10	Dark Mode	287
10.1	What is Dark Mode	288
10.2	Dynamic Type	290
10.2.1	Dynamic Type in action	298
10.3	Colors	303
10.3.1	System Colors	303
10.3.2	Semantic Colors	307
10.4	SF Symbols	309
10.4.1	SF Symbols App	313
10.5	Supporting Dark Mode in the app	314
10.5.1	Testing it on the simulator	319

10.6 Conclusion	326
10.7 What we learned	326
11 Layout	329
11.1 Weathery elements	330
11.2 Adding the location button	332
11.3 Adding the search button	340
11.4 Adding the text field	345
11.5 Adding a stack view on the top	350
11.6 Adding a stack view on right	356
11.7 NSAttributedString	362
11.7.1 Formatting our temperature label	364
11.8 Conclusion	371
11.9 What we learned	372
12 Communication Patterns	373
12.1 Setup	374
12.2 The challenge	380
12.3 Notification Center	383
12.3.1 How does it work	383
12.3.2 Seeing it in action	385
12.3.3 Pros & Cons	394
12.4 Closures	395
12.4.1 How do they work	396
12.4.2 Adding closures in our Weathery app	397
12.4.3 Pros & Cons	409
12.5 Protocol-Delegate	409
12.5.1 How do they work	410
12.5.2 Adding UITextField	414
12.5.3 Adding CLLocationManager	423
12.5.4 Adding the WeatherServiceDelegate	433
12.5.5 Pros & Cons	441
12.5.6 Which to use and when	441
12.6 Conclusion	442

12.7	What we learned	443
13	Parsing JSON	445
13.1	The Weather Payload	446
13.2	Decodable (JSON > Swift)	449
13.3	Encodable - Swift > JSON	451
13.4	Coding keys	452
13.5	Dates	453
13.5.1	Milliseconds	453
13.5.2	iso8601	453
13.5.3	Custom DateFormatter	454
13.6	Arrays	455
13.7	Embedded Arrays	457
13.8	Dictionaries	459
13.9	Other types	460
13.10	Your turn	462
13.11	Parsing JSON in the Weathery App	466
13.12	Conclusion	469
13.13	What we learned	470
14	HTTP	471
14.1	The Weathery API	472
14.2	Adding networking	474
14.2.1	Create a WeatherData struct	475
14.2.2	Add networking to the WeatherService	476
14.3	Conclusion	483
14.4	What we learned	483
15	Error handling	485
15.1	Errors as enums	486
15.2	try, try?, try!	487
15.3	Programmer errors	489
15.3.1	assert()	489
15.3.2	assertionFailure()	490

15.3.3 precondition()	490
15.3.4 preconditionFailure()	491
15.3.5 fatalError()	491
15.4 Dealing with errors in our app	492
15.4.1 Space in city name	493
15.4.2 Failing fast	497
15.4.3 HTTP error codes	498
15.4.4 Parsing JSON safely	503
15.4.5 No Wi-Fi	505
15.5 Conclusion	514
15.6 What we learned	515
16 Unit testing	517
16.1 What are unit tests?	518
16.2 Vegas Baby!	521
16.3 How to add tests to an existing project	529
16.4 Unit testing JSON	533
16.5 Loading JSON from a file	544
16.5.1 Creating the JSON file	545
16.6 Mocking the WeatherService	554
16.6.1 How does mocking work	554
16.6.2 Defining the protocol	556
16.6.3 Testing our UI	558
16.6.4 Testing for errors	561
16.7 Conclusion	571
16.7.1 Solutions	572

Preface

The goal of this book is to teach people how to build apps in iOS. If you are a hobbyist, developer, or just someone who owns a Mac and is curious, come on in and see how iOS apps get built.

This book starts off with the hobbyist in mind. In the first couple chapters I assume you know nothing about iOS development, and show you how to set up and install Xcode, as well as how to build your first iOS applications.

Once we get a few apps under our belt, we then shift gears slightly and start looking at iOS development more seriously as a professional. Here you will go deeper into the iOS stack, learn how to use more advanced app development techniques, and focus more on the skills it takes to start a career and work as a professional.

So if any of that is of interest to you come on in. I will be happy to share with you what I know.

Jonathan Rasmusson / Oct 31, 2020 / Calgary, Alberta, Canada

What's in the book?

Part I starts you off by teaching you the basics of how iOS works and app building in general. Here you will build two HelloWorld style apps, get used to working with Xcode, and get a feel for how iOS development works.

In Part II, *Weathery*, we build a weather fetching application and move beyond the basics and get you building apps more akin to how professional iOS developers work at big companies. Here we leave the world of story boards behind, start building apps programmatically, and go deeper into how different iOS communication patterns, how to work with Dark Mode and Vector images, as well as sections or error handling and unit testing.

Work in progress

This book is very much a work in progress and is by no means complete. In the spirit of releasing early it was released in it's imperfect form to get your feedback.

- So tell me what you like.
- Tell me what you don't like.
- And let me know what, if anything, is missing or could be explained more clearly.

At the end of each chapter you should find a small feedback link like this one.

[Feedback form](#)

Click it, let me know what you think, and hopefully together we can build a great resource for teaching and showing others how much fun iOS is and how lucky we are to work with it.

Thank you in advance! Let's begin.

Chapter 1

Setting up Xcode

The goal of this chapter is to make sure you and your machine are setup. In this chapter you are going to learn:

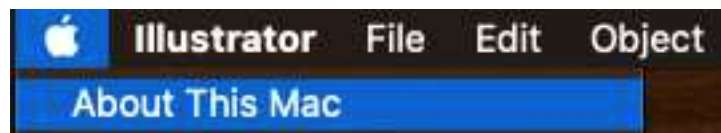
- How to check your Mac's Operating System version
- How to download and install Xcode

1.1 Checking your OS

To work through the contents of this book you will need a Mac capable of running **Catalina 10.15.+**



You can check what version of OS your Mac is currently running by clicking the Apple icon in the upper left hand corner of your screen and then selecting *About This Mac*.



That will pop up a window that looks like this which will then show you your Mac OS's version number.

Table 1.1: Macs capable of running Catalina 10.15.

MacBook (2015 or newer)
MacBook Air (2012 or newer)
MacBook Pro (2012 or newer)
Mac mini (2012 or newer)
iMac (2012 or newer)
iMac Pro (from 2017)
Mac Pro (2013 or newer)



To work through the examples in this book you want that number to be at least **10.15.6**. You may be able to get by with slightly older versions, but we are going to be using **Xcode 12** and it requires a modern version of Catalina.

Apple is pretty good at making sure older Macs are capable of running newer versions of their operating systems. So to see if your Mac is able to run Catalina, check this list for when your Mac was made.

Once you've confirmed your Mac has Catalina installed, you are ready to install Xcode.

1.2 Downloading Xcode

Xcode is the editor we use in iOS to build and ship applications. It's our source code editor. It runs our iPhone simulators. And it's in here that we do all our programming and development.

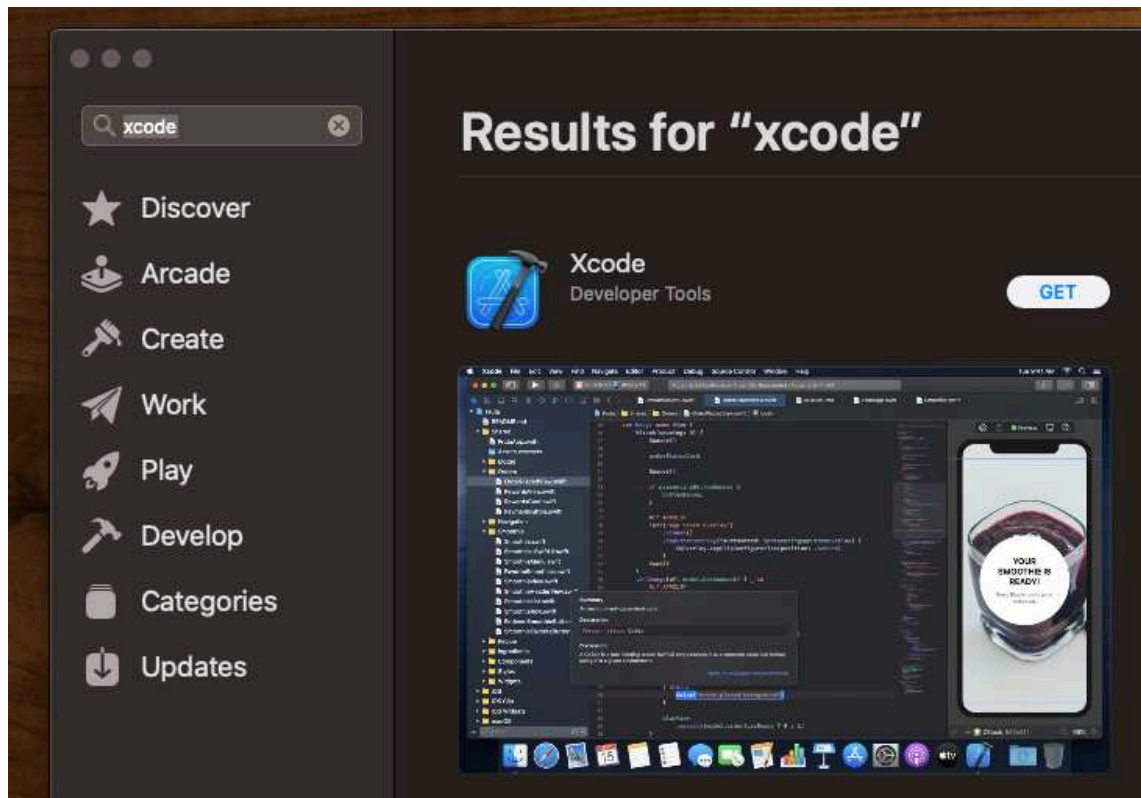
There are two ways to download Xcode:

1. Via the App Store
2. Directly from the internet

1.2.1 The App Store

The easiest (but not necessarily fastest) way to download Xcode is via the App Store. You can download Xcode directly from the Internet (as we will see in the next section). But it requires an Apple Developer ID. So if you don't have an ID, and you don't plan on creating one, the App Store is for you.

To download Xcode via the App Store, fire up the App Store on your Mac, search for *Xcode*, click the *Get* button, and eventually the *Get* button will change to an *Open* button once Xcode finishes downloading. This will install Xcode in the Applications folder on your Mac, at which point you will then be able to open and run.



1.2.2 From the Internet

While downloading Xcode through the App Store works, it can take a while and you don't get much feedback telling you when you can expect the download to be done.

An alternative approach is to Google "download xcode xib" and find a direct link to the Xcode file like [this](#) one here on Stack Overflow.

stackoverflow About Products For Teams Search...

Home

PUBLIC

Stack Overflow

Tags

Users

FIND A JOB

Jobs

Companies

TEAMS What's this?

Free 30 Day Trial

1 Answer Active

▲ You can find the DMGs or XIPs for Xcode and other development tools on <https://developer.apple.com/download/more/> (requires Apple ID to login).

4915 ▼ You must login to have a valid session before downloading anything below.

✓ *(Newest on top. For each minor version (6.3, 5.1, etc.) **only the latest revision is** list.)

🕒

Xcode 12

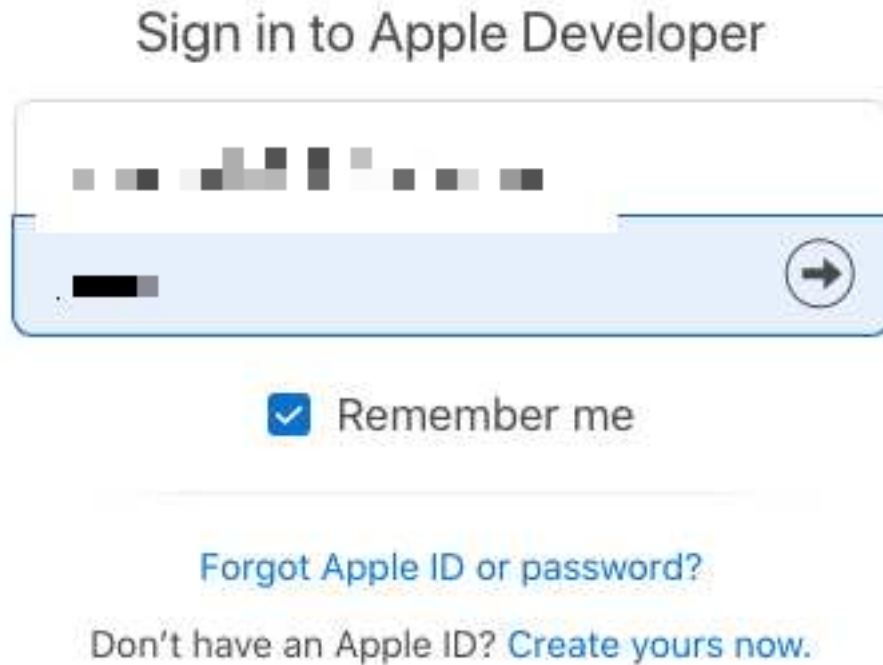
- [12.2 beta](#)
- [12](#) (Requires macOS 10.15.4 or later) (Latest as of 17-Sept-2020)

Xcode 11

- [11.7](#) (Latest as of Sept 02 2020)
- [11.6](#)
- [11.5](#)
- [11.4.1](#) (Requires macOS 10.15.2 or later)
- [11.3.1](#)
- [11.2.1](#)
- [11.1](#)
- [11](#) (Requires macOS 10.14.4 or later)

You will need a valid Apple Developer account to download Xcode via a link as clicking on these links will redirect you to the Apple Developer site. But if you already have an account, or you are planning to create one, you can follow these steps to download the Xcode xib file directly.

Scroll down to the “Xcode 12” download section, click the latest non-beta version of Xcode 12. And this will redirect you to the Apple Developer site. Sign-in with your Apple Developer ID.



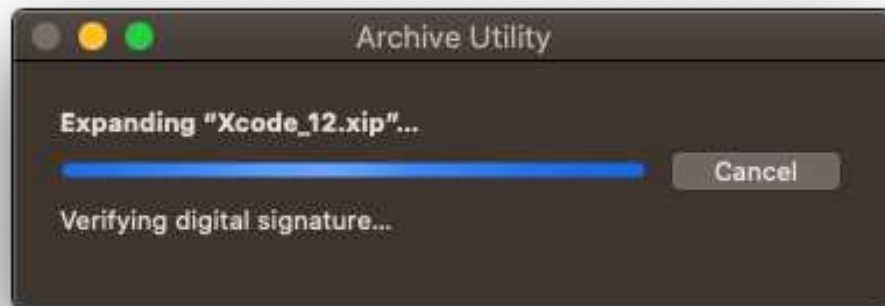
Once signed in, Xcode will then begin downloading automatically.



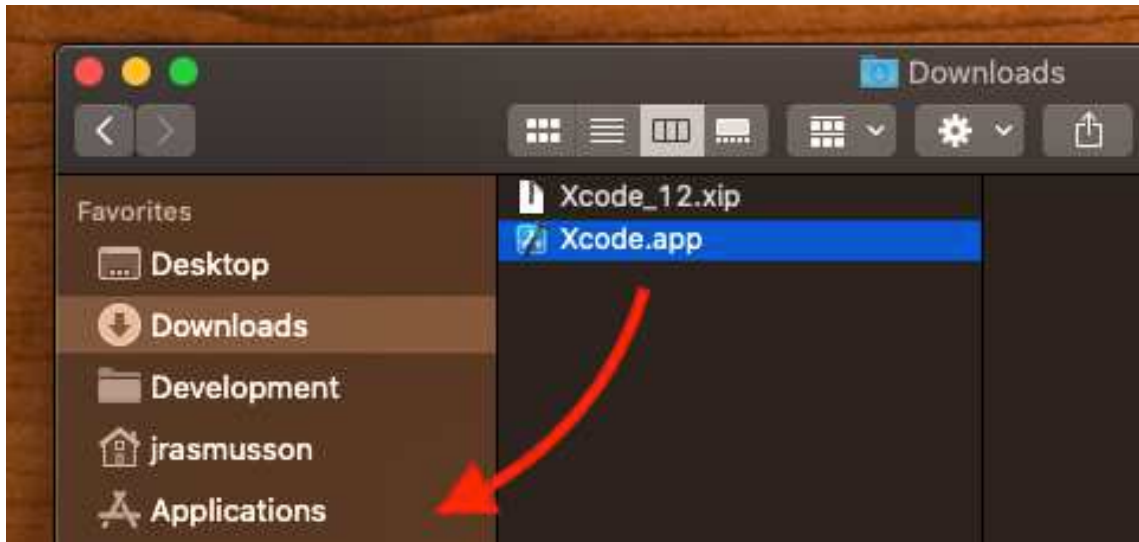
Once your download is complete go to your download directory.



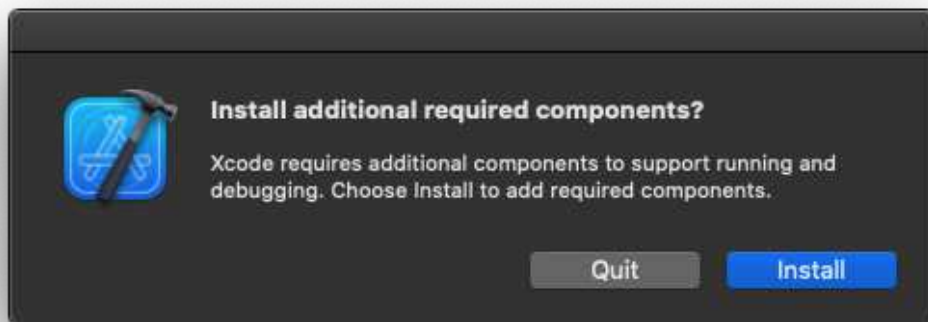
Double click the *xip* file to expand it (this may take several minutes).



And then drag it into your Applications directory and double click to open.



It will then ask for permission to install some additional components. Say yes to *Install*.



Then voila. Welcome to Xcode!



1.3 Conclusion

Congrats! You've downloaded and setup Xcode.

Now that we've got Xcode setup, we can start building applications. In the next chapter, Hello World, you are going to build your first iOS app and get it running in a simulator.

1.3.1 What we learned in this chapter

- How to check your Mac's Operating System version
- How to download and install Xcode

1.4 Feedback

Any suggestions on how to make this chapter better?

[Feedback form](#)

