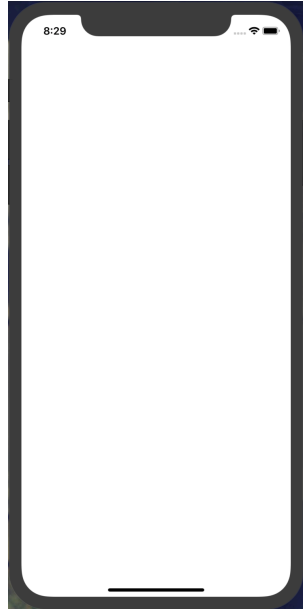


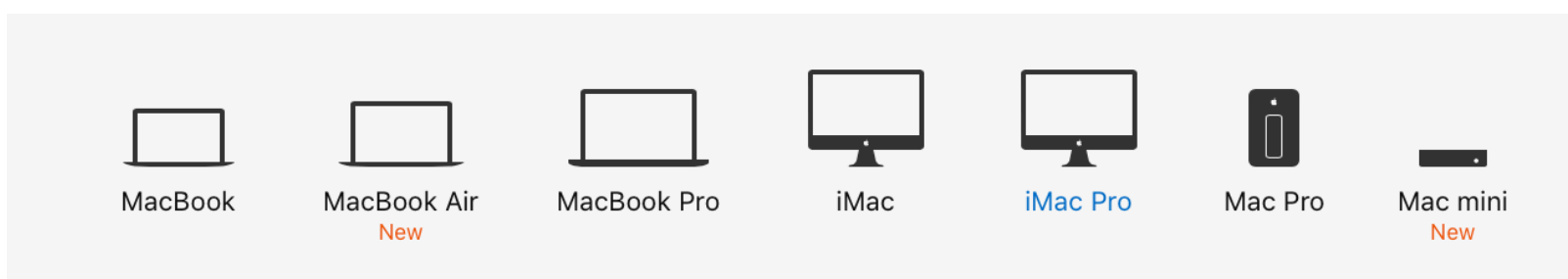
CHAPTER TWO : PREPARE FOR HELLO WORLD APP



I. WHAT HARDWARE DO I NEED?

In order to develop iOS apps, you have to have a Mac. You will need to get a Mac running on MacOS X version 10.12 (or later). However, if you have a PC, some people will use a virtual environment like Hackintosh instead of buying a Mac. This is highly discouraged for beginners (and for all serious Apple developers) because in order to learn and code in a stable environment, you should use an environment (both hardware and software) which is good enough to run all the iOS development tools smoothly.

So if you don't have a Mac yet, you can click the following link and get yourself one, I promise that you won't regret it!



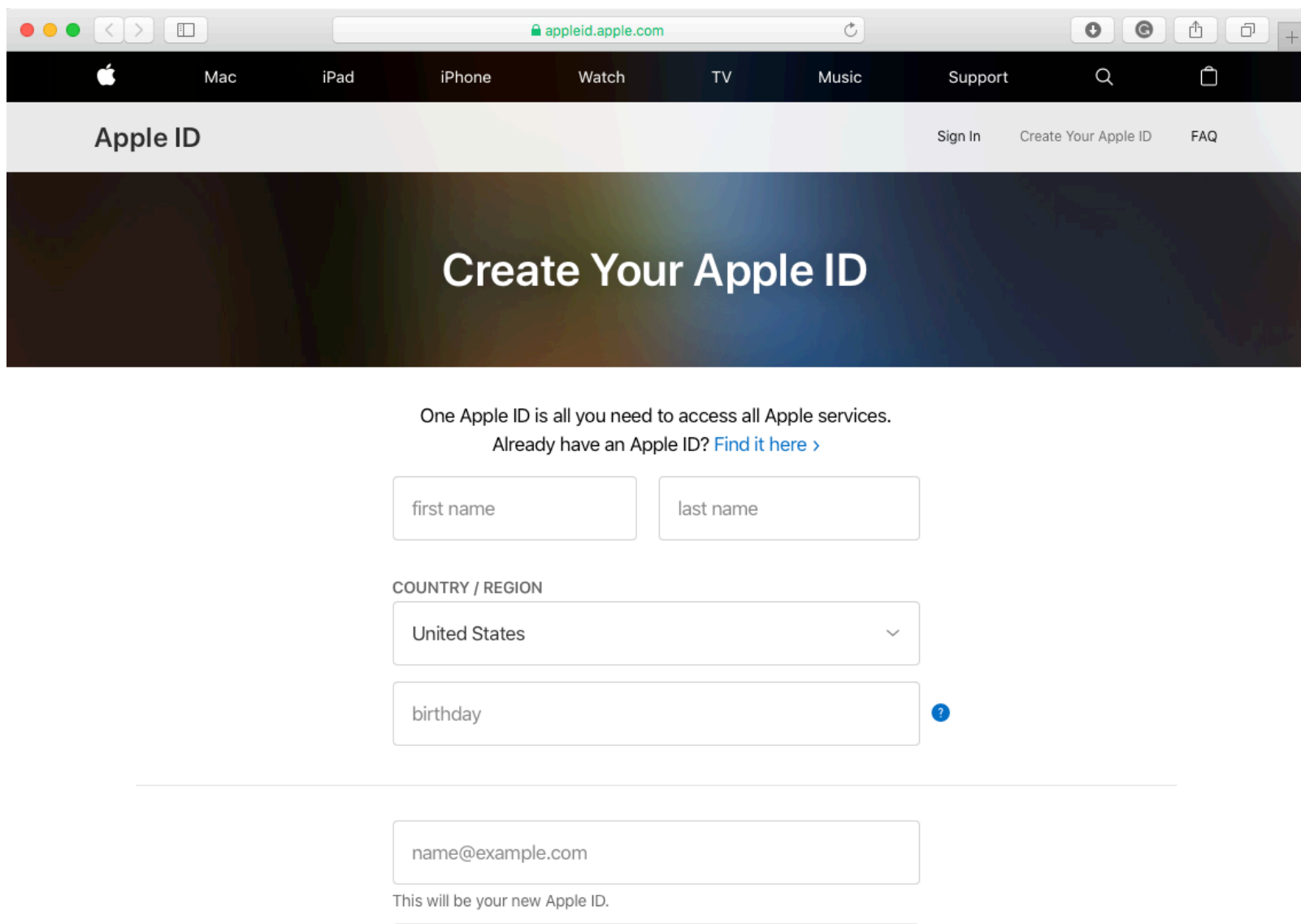
<https://www.apple.com/mac/>

II. GET AN APPLE ID

If you've already have an iPhone, iPad, or Mac, I'm pretty sure that you have an Apple ID.

For those who are new the the Apple ecosystem, you will need an Apple ID to download Xcode, access iOS SDK documentation, and other technical resources. Most importantly, it will allow you to deploy your app to a real iPhone/iPad for testing.

If you have downloaded an app from the App Store, it is quite sure that you already own an Apple ID. In case you haven't created your Apple ID before, you have to get one. Simply go to Apple's website (<https://appleid.apple.com/account>) and follow the procedures for registration.



The screenshot shows the Apple ID creation page in a web browser. The browser's address bar displays "appleid.apple.com". The page has a dark header with the Apple logo and navigation links for Mac, iPad, iPhone, Watch, TV, Music, and Support. Below the header, the "Apple ID" section includes links for "Sign In", "Create Your Apple ID", and "FAQ". The main heading is "Create Your Apple ID". Below this, a message states: "One Apple ID is all you need to access all Apple services. Already have an Apple ID? [Find it here >](#)". The registration form consists of several fields: "first name" and "last name" (text inputs), "COUNTRY / REGION" (a dropdown menu currently showing "United States"), "birthday" (a date picker with a blue question mark icon), and an email address field (pre-filled with "name@example.com"). Below the email field, a note says "This will be your new Apple ID."

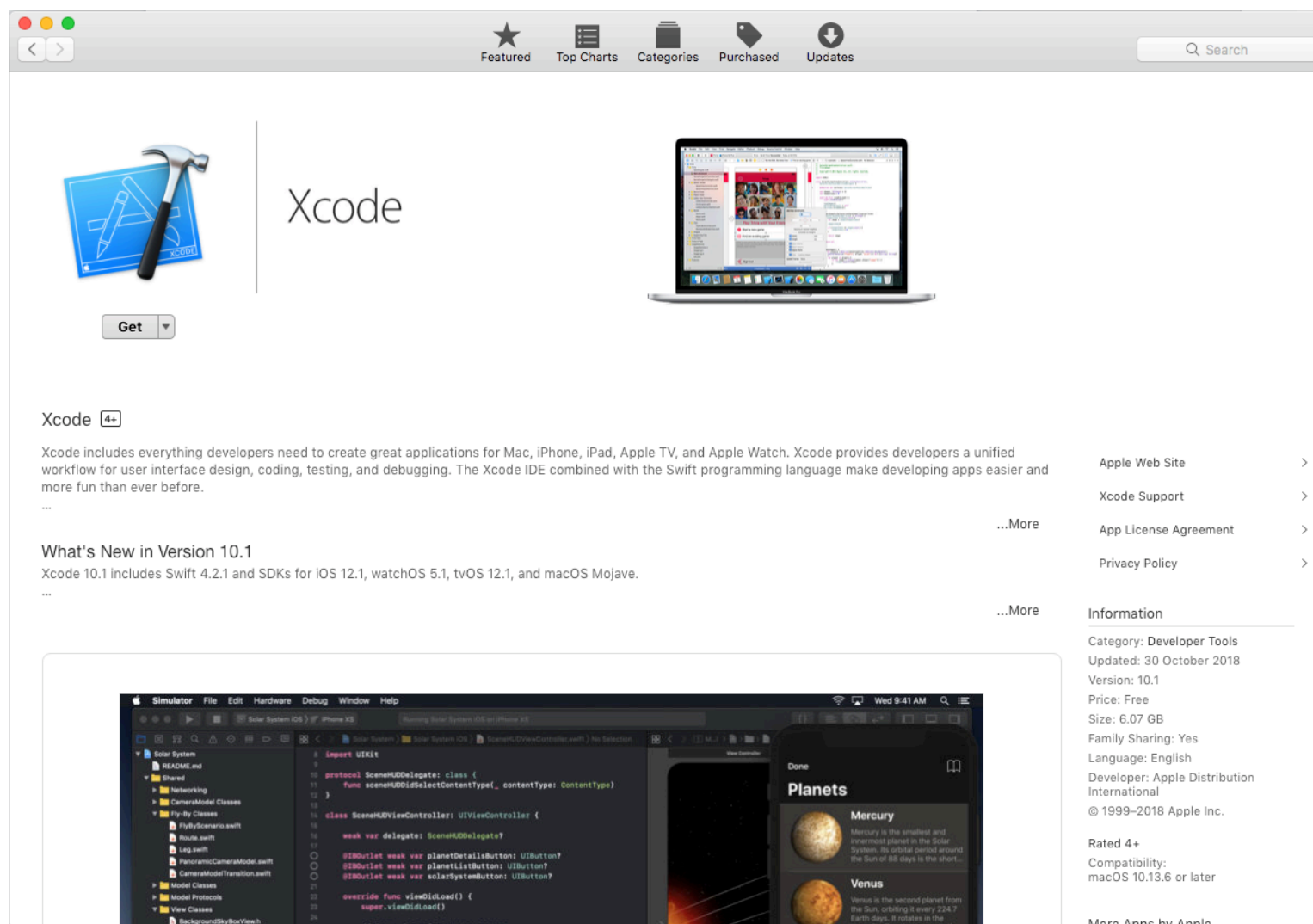
III. INSTALL & CONFIGURE XCODE

To start developing iOS apps, Xcode is the only tool you need to download. Xcode is an integrated development environment (IDE) provided by Apple. Xcode provides everything you need to kick start your app development.

So simply go to your dock and click App Store icon to enter the App Store.



In the Mac App Store, simply search "Xcode" and click the "Get" button to download it.



And in order to check if you have correctly installed Xcode, type in your terminal the following command:

```
/usr/bin/xcodebuild -version
```

If your Xcode is ready to go, you will have the following result:

```
[carlistles-mbp:~ Carlistle$ /usr/bin/xcodebuild -version  
Xcode 9.4  
Build version 9F1027a  
carlistles-mbp:~ Carlistle$
```

Here my Xcode version is 9.4, if your version or build version is different from mine, don't worry. However, if your Xcode version is the version 8 or older, you should update your Xcode to be able to develop in iOS 10 smoothly. In order to do so, go to App Store, choose the application Xcode, click update, and confirm when the prompt appears.

So it is time to fire up your Xcode for the first time! You can double click the app icon in your dash board or the Xcode.app in your application folder. Just to let you know, you can open your Xcode IDE with a terminal command as well:

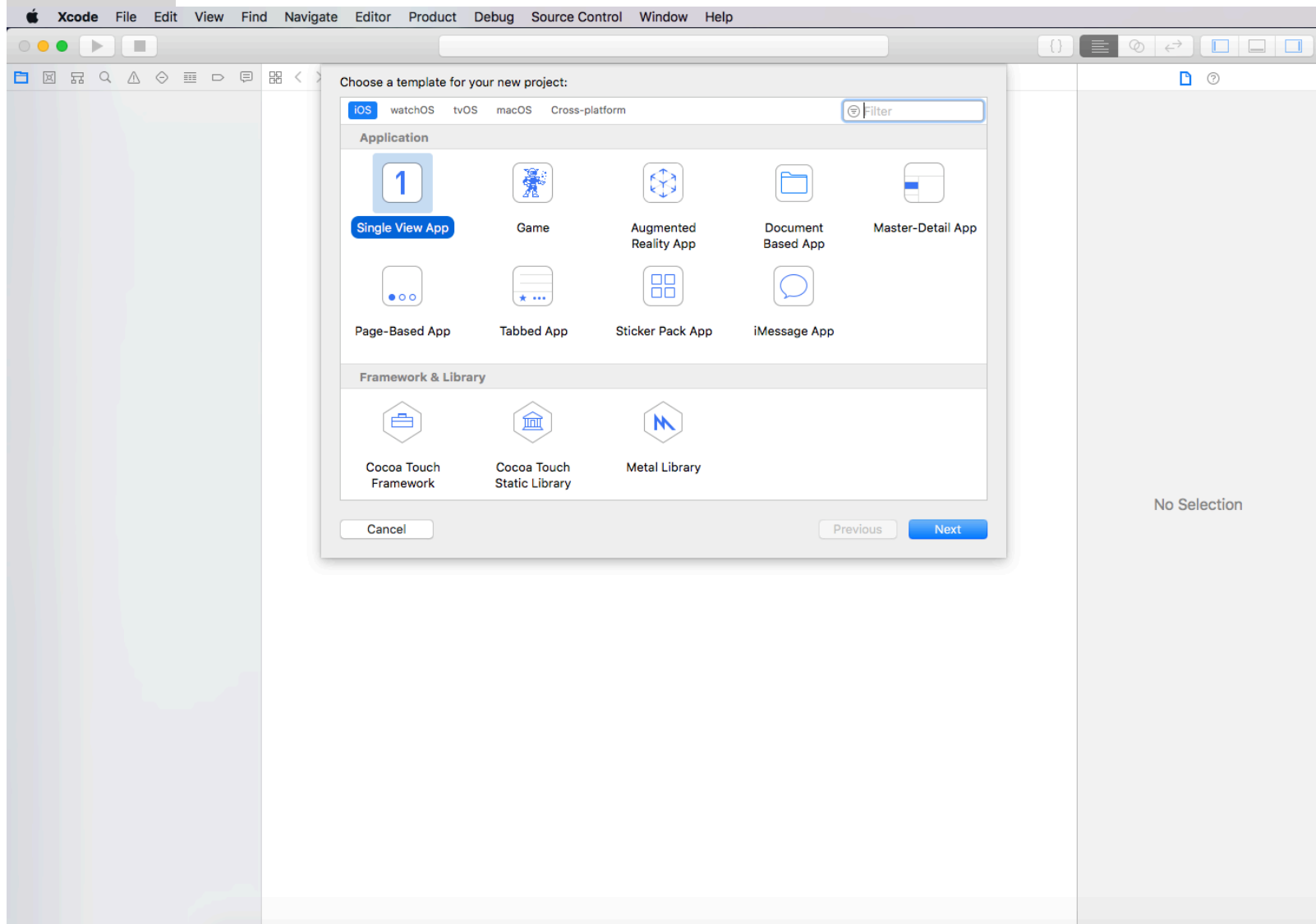
```
Carlistle — -bash — 80x24  
Last login: Wed Feb 27 12:31:30 on console  
[Carlistles-MacBook-Pro:~ Carlistle$ open -a Xcode  
Carlistles-MacBook-Pro:~ Carlistle$
```

The command open with option `-a` allows you to open a certain application. Moreover, this command is more useful when you want to open a certain folder in Xcode. To do so, just add your directory path after your first option:

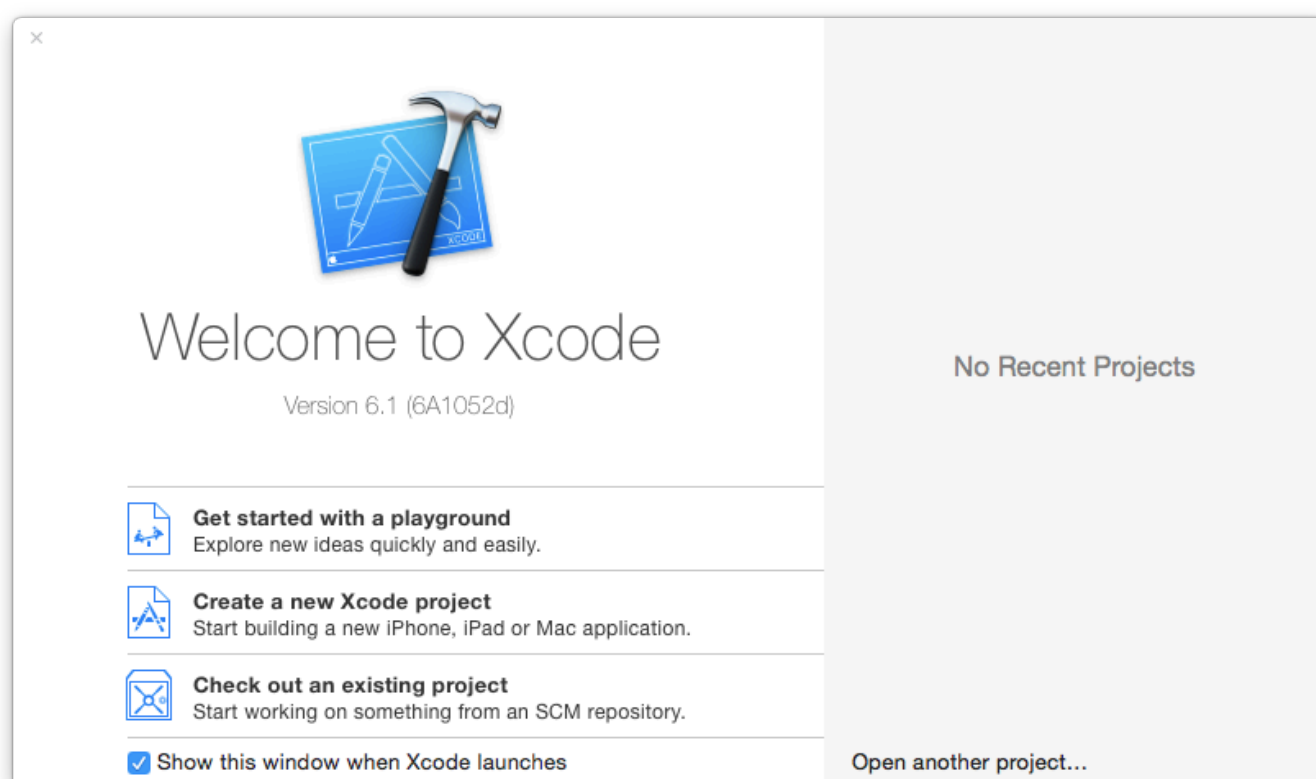
```
Downloads — -bash — 80x24  
Carlistles-MacBook-Pro:Downloads Carlistle$ open -a Xcode ./test\ project/
```

Downloads		
Name	Size	Kind
▶ test project	--	Folder

Here I opened the (relative)directory `./test\ project/` using Xcode. You should pay attention that in the finder the name of the folder is test project, but I used `./test\ project/` in my terminal.



IV. START YOUR FIRST XCODE PROJECT

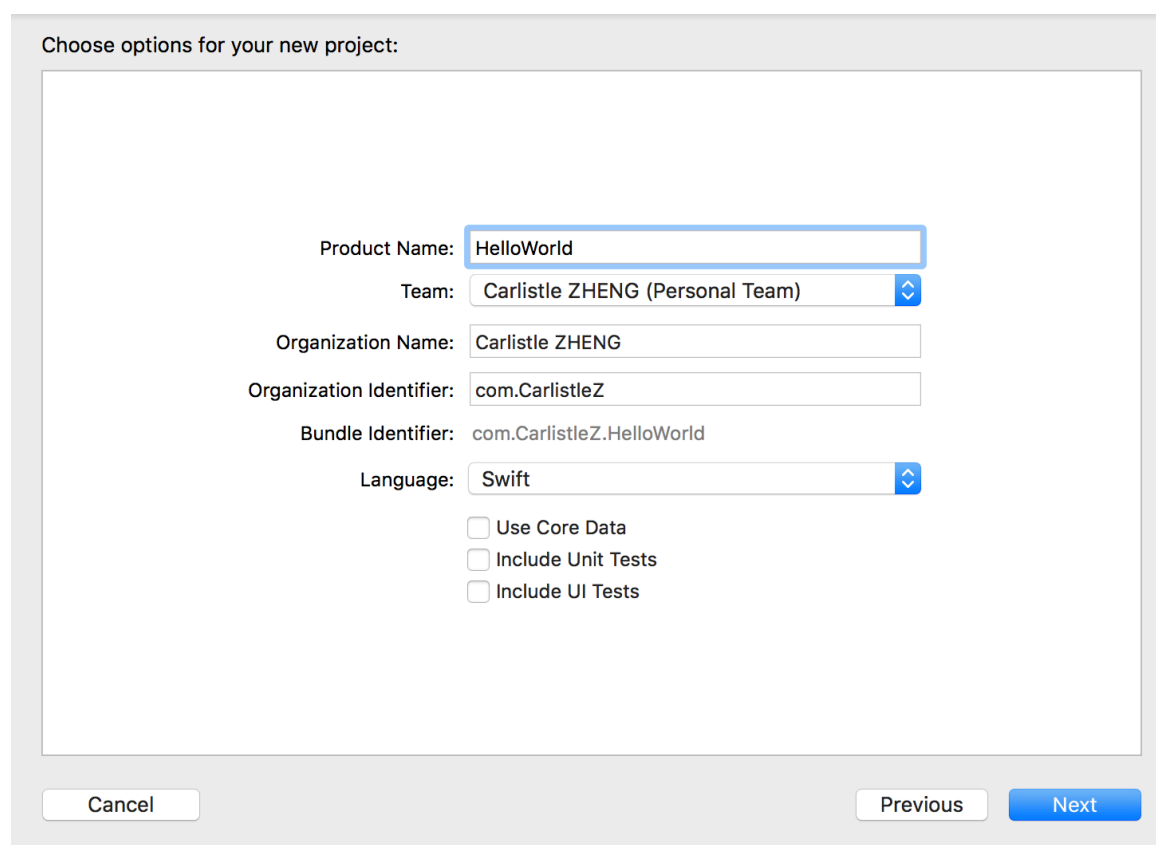


If you have done everything correctly so far, you will see this welcome dialog. Here are your options:

- The playground is a great place for beginners to learn basic swift syntax. However, you can't really build an app with playground.
- For the most of the time, you will choose “Create a new Xcode project”, you can also open recent projects that you created or opened from the Recent Project List on the right.
- If you are working with others and you use a version control like Git or SVN in your team, choose this to pull from the remote repository.

For now, let's just choose the second option “Create a new Xcode project” to move on. For our first app, choose Single View App(which is almost always the one we choose) under the tab iOS. In the second half of this book, we are going to use “Document Based App”, “Master-Detail App”, and “Page Based App”. But for now, let's keep everything nice and simple with the model “**Single View App**”.

Then you will see the following window to let you choose options for your new project:



The screenshot shows a dialog box titled "Choose options for your new project:". It contains several input fields and checkboxes. The "Product Name" field is highlighted with a blue border and contains the text "HelloWorld". The "Team" dropdown menu is set to "Carlistle ZHENG (Personal Team)". The "Organization Name" field contains "Carlistle ZHENG". The "Organization Identifier" field contains "com.CarlistleZ". The "Bundle Identifier" field contains "com.CarlistleZ.HelloWorld". The "Language" dropdown menu is set to "Swift". There are three unchecked checkboxes: "Use Core Data", "Include Unit Tests", and "Include UI Tests". At the bottom, there are three buttons: "Cancel", "Previous", and "Next".

In the field **Product Name**, type the name of your project(in this case HelloWorld). Why do I use `HelloWorld` instead of `Hello World`? If you use the latter, Xcode will automatically add a `-` to make the name like this: `Hello-World`. So for beginners, spaces are discouraged here. This small habit will save you a lot of headaches later on.

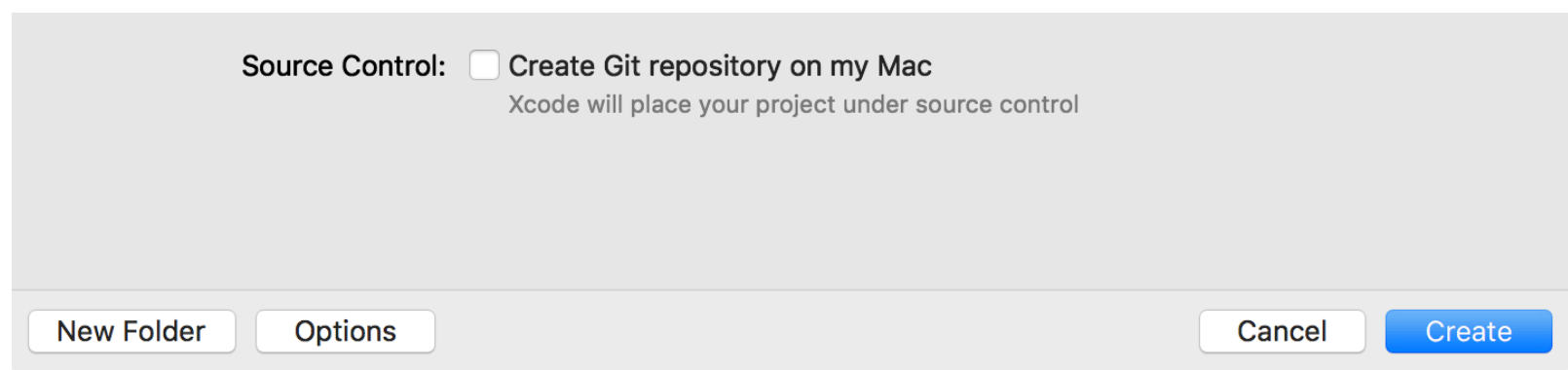
In the field **Organization Identifier**, use `com.` and your `user id` to make a unique identifier. If you leave this field blank, you won't be able to test your app out on a real device. In my Xcode I chose `com.CarlistleZ` as my Organization Identifier, you should(and have to) change it to your own ID.

Now leave everything else intact. If you wonder what “Core Data”, “Unit Tests”, and “UI Tests” are, I highly encourage you to check them out in the Apple official documentation in following links:

- <https://developer.apple.com/documentation/coredata>
- <https://developer.apple.com/documentation/xctest>
- https://developer.apple.com/documentation/xctest/user_interface_tests

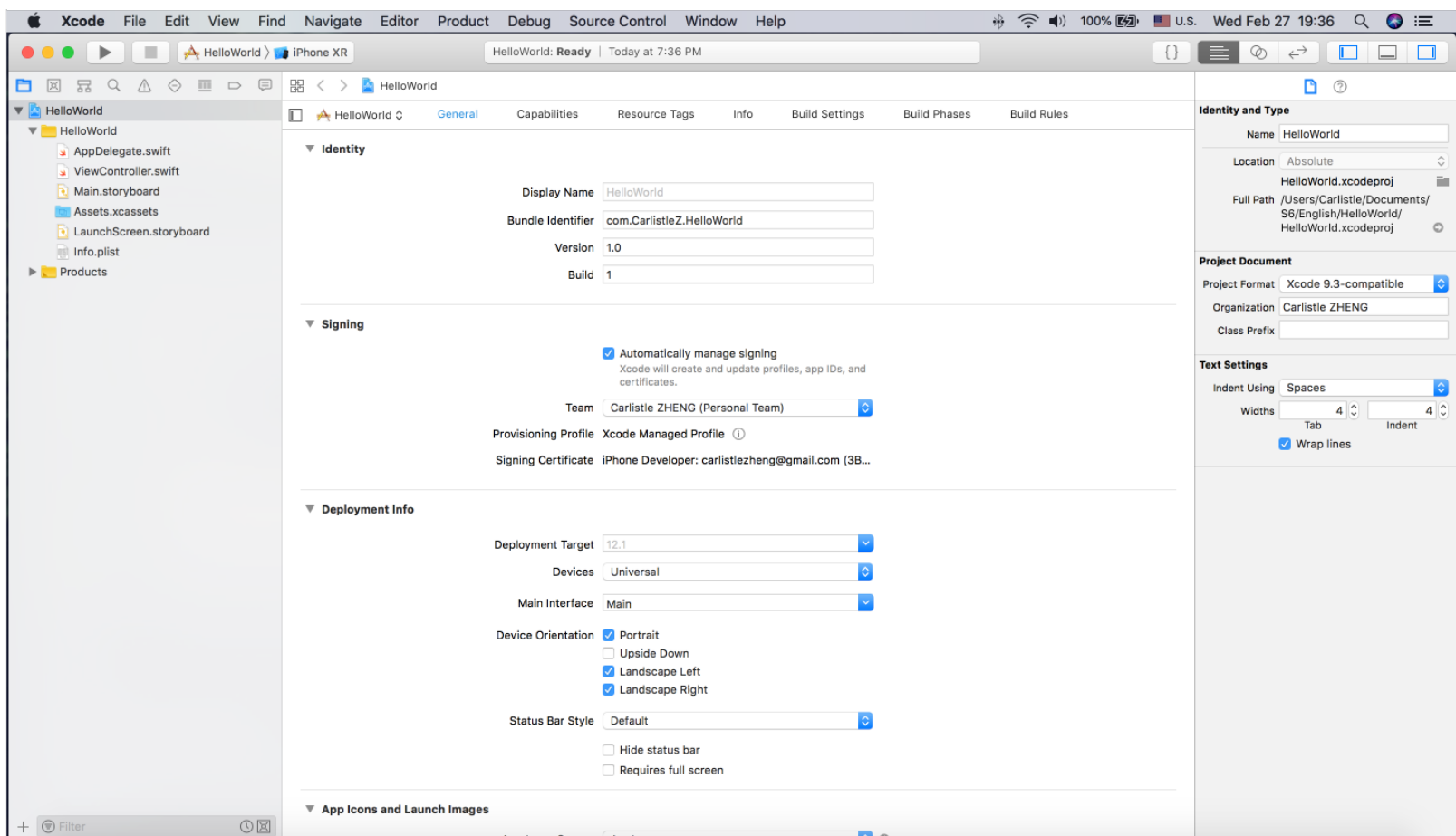
Then choose where you want to put your project root folder on your disk. This directory is the origin point for your entire app. So for any relative path you use, you just need to add this before your relative path to form a full path.

If you have some project experience, you probably have used Git before. So here is where you create a `.git` folder for your project. If you don't know what Git is, don't worry! We don't really need a Git for now, so keep it unchecked and click **Create**.

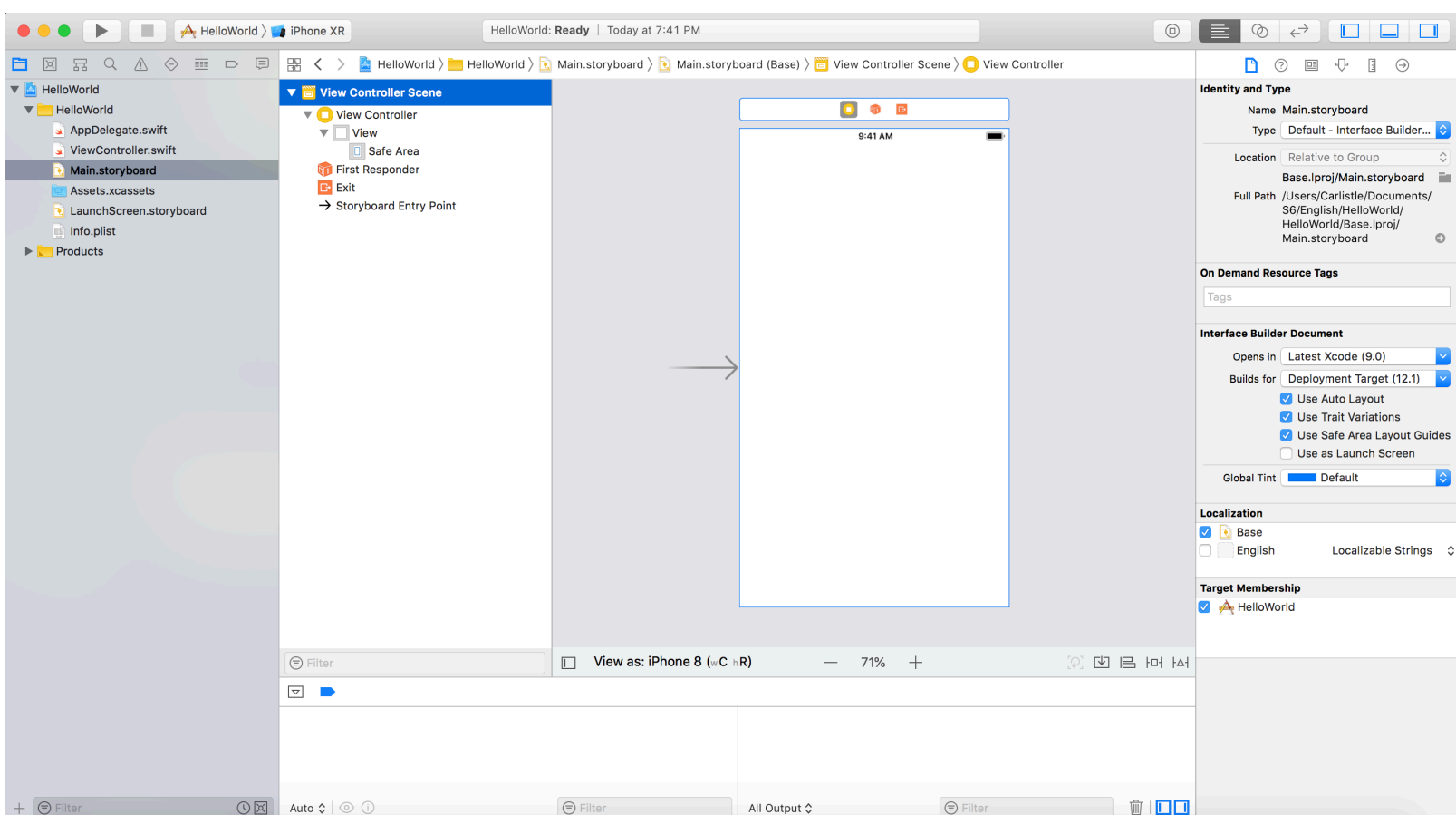


V. GET TO KNOW THE WORKING INTERFACE

If you've done everything properly, you should see our main working window like this:



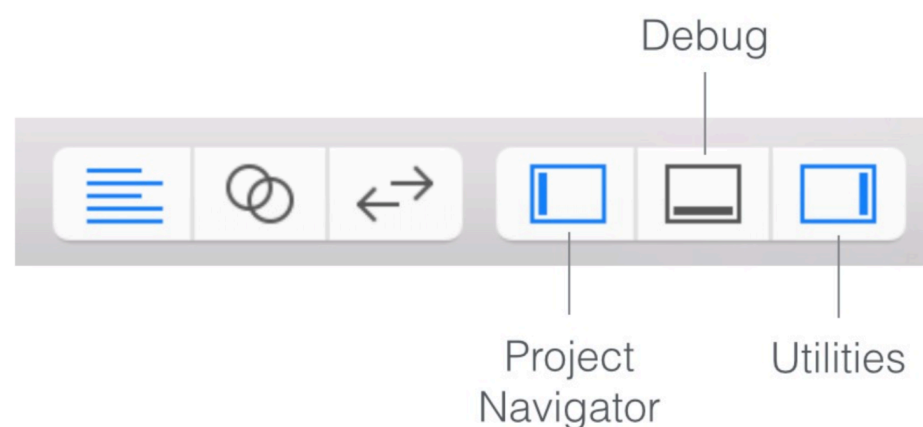
Great! Now the preparation is done and we will code our app in this interface. In this “General” tab, you will see your project information. Select **Main.storyboard** on the left file tree to see your app UI.



You can see the screen of our app(nothing for now...) in the middle area. In the left(**project navigator**) area, you can see all the files in your project, you can also see the navigators for source control, symbol, find, issue, test, debug, break point, and report.

In the right(**Utilities**) area, you can change the settings and parameters of the file in the middle area, this is the area that we use the most often.

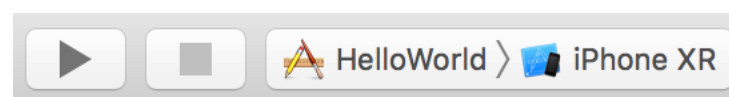
In the bottom(**Debug**) area, you can see the console output, debug info, and a terminal of your project. Just like every IDE(Integrated Development Environment), this is the part where you visualize the result without any GUI(Graphic User Interface).



If you have a MacBook Air, a MacBook, or a 13 inch MacBook Pro, your screen space might not be enough for you to work freely on your main editing area. On the upper right corner, you can toggle very easily these three areas, or use `cmd+0`, `cmd+option+0`, `cmd+shift+y`.

VI. RUN YOUR APP FOR THE FIRST TIME

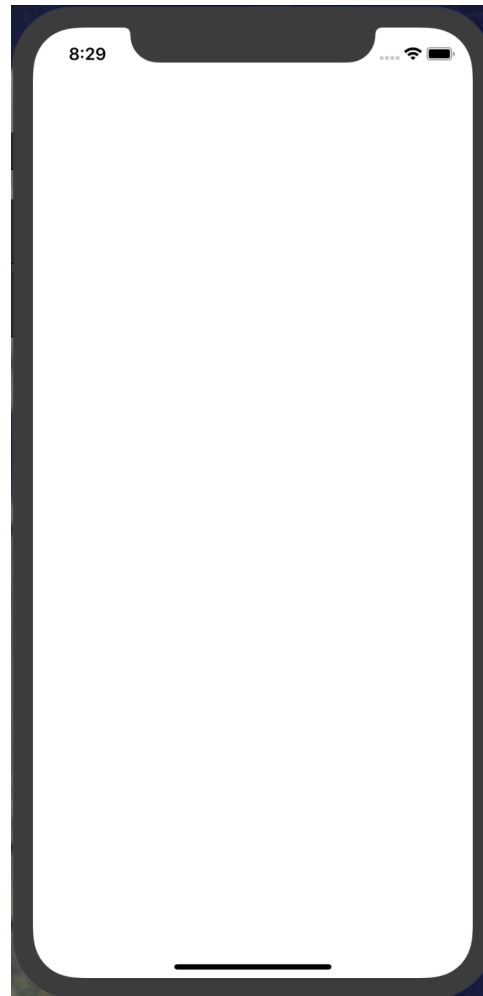
Can't wait to test out your very first app? On the upper left corner, you can choose to run your app on a simulator, a generic iOS device, or on a real device connected to your Mac.



In order to run a simulator, you need a lot of extra memory and system resources. If your Mac has 16Gb or 32Gb memory, your simulator is very likely to run smoothly. However, if your machine is an old model with only has 8Gb memory, you should close some apps that

you don't currently need before going to the simulator. In order to save system resources, you should never keep two simulators running at the same time (even you have 32Gb DDR4 in your Mac)!

Now click run, you will see a simulator like this:



Awesome! Since we haven't designed and coded our UI, we only have a blank screen in our app. In the next chapter, you will build your UI little by little and you will learn more about Xcode and Swift.

EXERCISE #1

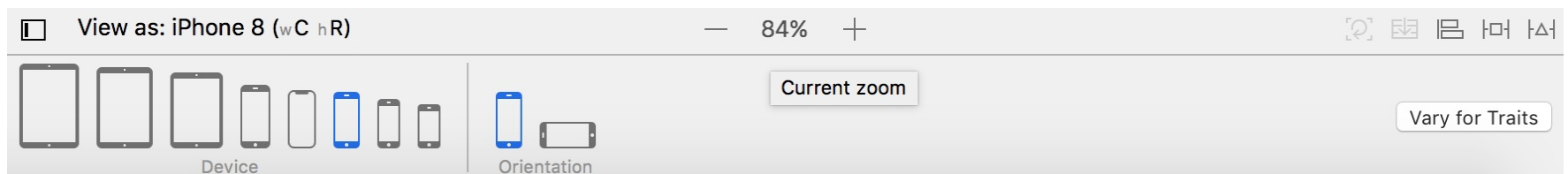
You have started this HelloWorld app in this chapter. Now try to start a new project of your own. You can choose any name you want with the suggested naming convention. If you don't know what that is, go to **Part IV** to give it another look. If you are also comfortable with Git, try to use Git for this app and experiment how to use Git in Xcode

Hint: Navigator/Source Control Navigator

EXERCISE #2

You have seen me running the demo app on an iPhone XR. Now try to run your app on a different device, an "iPad Pro 12.9 inch 2nd generation" for instance. Do you see any difference between them? Think about how these differences can influence our app. Do we need to write many versions for different devices to adapt to different hardware?

Hint: Screen sizes, Home button, Touch ID, Face ID...



SUMMARY

In this chapter, we have worked on these topics:

- set up our development environment Xcode
- learnt how to start a project in Xcode
- got familiar with Xcode UI
- ran an app on a simulator or a real device

If any of these is unclear to you, please make sure to go back and read the related part or parts before moving on to the next chapter.

For reference, you can download the complete Xcode project from

https://github.com/CarlistleZ/MyiOSTutorial/blob/master/ChapterTwo_HelloWorld.zip