

第 1 章 iOS 简介与环境搭建

【教学目标】

- ❖ 了解 iOS 开发的相关知识
- ❖ 掌握 iOS 开发环境搭建的相关知识
- ❖ 运行第一个 iOS 程序

1.1 初识iOS

iOS 是由美国苹果公司开发的移动操作系统。苹果公司早在 2007 年 1 月 9 日的 MacWorld 大会上发布了 iOS 系统，最初是设计给 iPhone 使用的，后来陆续套用到 iPod touch、iPad、Apple TV 等产品上。iOS 与苹果的 Mac OS X 操作系统一样，属于类 UNIX 的商业操作系统，原名为 iPhone OS，因为 iPad、iPhone、iPod touch 都使用 iPhone OS，所以在 2010 年苹果全球开发者大会上宣布将其改名为 iOS。

1.1.1 iOS的发展历程

iOS 经过多年的不断发展和完善，现成为最受用户欢迎的主流手机操作系统之一。2007 年 6 月，苹果公司发布第一版 iOS 操作系统，名为“iPhone Runs OS X”。同年 10 月，苹果公司发布了第一个本地化 iPhone 应用程序开发包（SDK），此版本支持多点触控、虚拟键盘输入、邮件发送等功能。

2008 年 3 月，苹果公司发布了第一个测试版开发包，并且将“iPhone runs OS X”改名为“iPhone OS”。同年 9 月，苹果公司将 iPod touch 系统也换成了 iPhone OS。该版本添加了 APP Store、截图功能，支持手写输入、中文、Office 文档和计时器等功能。

2010 年 2 月，苹果公司发布 iPad，同样搭载了“iPhone OS”。同年，苹果公司重新设计了“iPhone OS”的系统结构和自带程序。

2010 年 6 月，苹果公司将“iPhone OS”改名为“iOS”，同时获得了思科 iOS 的名称授权。到 2010 年第四季度，苹果公司的 iOS 占据了全球智能手机操作系统 26% 的市场份额。

2011 年 6 月，苹果公司发布了 iOS 5。该版本增加了联系人黑名单，把 Twitter 和 Siri 也整合到该系统中。自此，iOS 开始增加第三方应用。

2012 年 6 月，苹果公司在 WWDC 2012 大会上发布了 iOS 6，提供了超过 200 项新功能。

2013 年 6 月，苹果公司在 WWDC 2013 大会上发布了 iOS 7，几乎重绘了所有的系统

App, 去掉了所有的仿实物化, 整体设计风格转为扁平化设计, 界面整体透露出简洁、动感、时尚之感。

2014年9月, 苹果公司在 WWDC 2014 大会上发布了 iOS 8, 在 iOS 7 的界面基础上对一些常用功能进行了大幅的改进, 增强了交互功能, 如通知中心、短信功能等, 同时添加了健康类应用、开放了输入法 API。图 1-1 为 iOS 8 的产品界面。



图 1-1 iOS 8 的产品界面

1.1.2 iOS的功能特性

iOS 是一款优秀、先进的移动操作系统, 具有简单易用的界面、令人惊叹的功能、超强的稳定性, 已经成为 iPhone、iPad 和 iPod touch 的强大基础。尽管其他竞争对手一直努力追赶, 但 iOS 内置的众多技术和功能让 Apple 设备始终保持着遥遥领先的地位。

iOS 的功能特性主要表现在以下 7 个方面。

1. 界面直观优雅

苹果公司的产品 (如 iPhone、iPad 和 iPod touch) 容易操作, 得益于 iOS 极具创新的 Multi-Touch 界面专为手指而设计。主屏幕简洁美观, 从内置 App 到 App Store 提供的近百万款 App 和游戏, 从进行 FaceTime 视频通话到用 iMovie 剪辑视频, 用户所触及的一切, 无不简单、直观、充满乐趣。

2. 功能丰富

iOS 不断丰富功能, 内置 App 越来越多, 让 iPhone、iPad 和 iPod touch 比以往更强大、更具创新精神, 用户使用起来其乐无穷。

3. 软件、硬件配置完美、高效

iPad、iPhone 和 iPod touch 的硬件和操作系统都是由苹果公司制造的, 不需要考虑兼容性问题, 可以让软件、硬件完美配置和高度整合。App 也能充分利用 Retina 显示屏、

Multi-Touch 界面、加速感应器、三轴陀螺仪、加速图形功能和更多的硬件加速功能。

4. 数量庞大的移动 App

iOS 平台拥有数量庞大的移动 App，几乎每类 App 都有数千种，而且每种 App 都很出色。苹果公司为第三方开发者提供了丰富的工具和 API，使得第三方开发者设计的 App 能充分利用每部 iOS 设备蕴含的先进技术。苹果公司将所有 App 都集中在服务器中，使用 Apple ID 即可轻松访问、搜索和购买这些 App。用户需要做的只是在设备上访问 App Store，然后下载。

5. 更新方便

iOS 免费更新，可以将其下载到 iPhone、iPad 或 iPod touch 上，更新非常方便。

6. 高安全性

iOS 提供内置的安全性、专门设计的低层级的硬件和固件功能，用来防止恶意软件和病毒，同时提供高层级的 OS 功能，在访问个人信息和企业数据时确保安全性。

为了保护用户的隐私，从日历、通讯录、提醒事项和照片获取位置信息的 App 必须先获得用户的许可。用户可以设置密码锁，以防有人未经授权访问自己的设备，还可以进行相关设置，允许设备在多次尝试输入密码失败后删除所有数据。该密码会为用户存储的邮件自动加密和提供保护，并能允许第三方 App 为其存储的数据加密。iOS 支持加密网络通信，用于保护 App 传输过程中的敏感信息。如果设备丢失或失窃，可以利用“查找我的 iPhone”功能在地图上定位设备，并远程擦除所有数据。一旦 iPhone 失而复得，还能恢复上一次备份的全部数据。

7. 内置众多辅助功能

引导式访问、VoiceOver 和 AssistiveTouch 功能，让更多的人可以体验 iOS 设备的迷人之处。例如，凭借内置的 VoiceOver 屏幕阅读技术，视力不佳的人可以听到其手指在屏幕上触摸到的项目说明。iOS 开箱即可支持 30 多种无线盲文显示屏，还能提供许多备受赞誉的辅助功能，如动态屏幕放大、隐藏式字幕视频播放、单声道音频、黑底白字显示等。

1.1.3 iOS 8 的新特性

iOS 8 是苹果公司推出的最新一代的 iOS 操作系统，其新特性主要表现在以下 5 方面。

① 扁平化。iOS 8 在 iOS 7 的扁平化外观的基础上进行了一些改动，增加了许多炫目的效果，让手机使用起来更加炫酷。

② 通知/控制中心。iOS 8 中，苹果公司修改了控制中心的外观以及一些细节部分。通知中心还添加了插件编辑功能，用户可以根据自己的需求增减通知中心里插件显示的内容。

③ 拍照。iOS 8 对拍照功能进行了大幅优化，在拍照的选项中提供了自动补光、智能曝光，还在照片处理中增加了照片着色、主题风格、光效、颜色以及图片裁剪、旋转等功能。

④ 开放输入法 API。在用户强烈的要求下，苹果公司终于开放了输入法 API，用户现在可以选择自己喜欢的输入法了。

⑤ 健康应用。在人们越来越关注自身健康的趋势下，iOS 系统通过健康应用的数据平

台与移动应用相结合，用户可以方便地查看自身健康状况，获取相应的医疗服务。

1.2 搭建iOS开发环境

若要在 iOS 系统中开发应用程序，就需要下载并安装 iOS SDK 和开发工具 Xcode。Xcode 是苹果公司的开发工具，可用于管理工程、编辑代码、构建可执行文件，它是一个集成开发环境（IDE），也可进行源代码调试、管理、性能调节等。iOS SDK 一开始是与 Xcode 独立发布的，从 Xcode 3.1 开始，Xcode 已经集成了 SDK，也就是说，用户下载 Xcode 后就不需要再下载 SDK。每个 SDK 会对应当前最新版本的 iOS 系统，因此在开发的时候尽可能选择新的开发环境，使得 App 可以适应新的系统。

Xcode 是一个集成开发环境（IDE），与 Eclipse 一样，可用于创建和管理 iOS 的项目、源文件，可将源代码编译为可执行文件，可在设备（或模拟器）运行代码或调试代码。

打开 App Store，在搜索栏中输入 Xcode，即可找到当前最新版本的 Xcode。下载并安装即可，图 1-2 是已经下载安装好的界面，直接打开即可进入 Xcode。

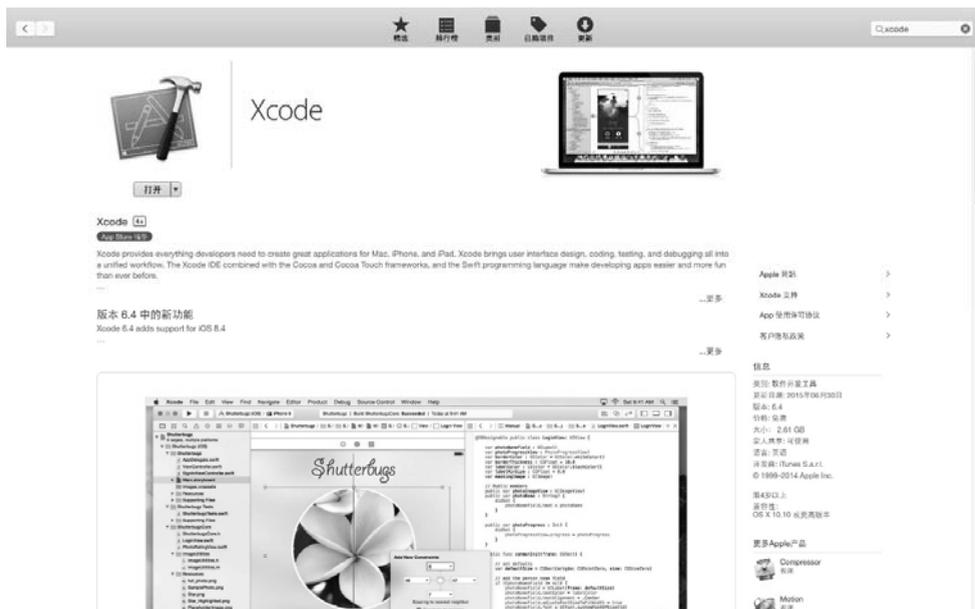


图 1-2 下载 Xcode

1.3 开发第一个iOS应用程序

1.3.1 Xcode工作区窗口

Xcode 的工作区窗口（如图 1-3 所示）分为上、下两部分，上部分为工具栏，下部分从左到右分别为导航器区域、编辑器区域和实用工具区域，不同区域担负着不同职责与功能。

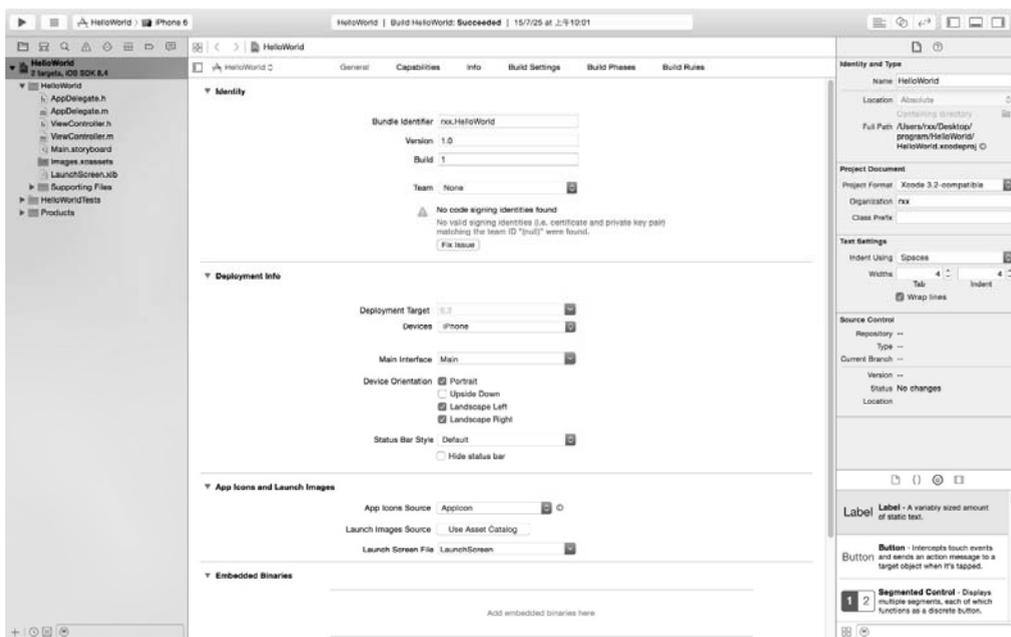


图 1-3 Xcode 的工作区窗口

工具栏左侧依次是用于启动和停止项目运行的控制按钮、用于选择运行方案的下拉菜单以及用于启动和禁用断点的按钮；工具栏中间的大方框是活动视图，用来显示当前正在进行的操作或处理；工具栏右侧是标准/辅助视图切换按钮、控制导航面板和实用工具面板的显示与隐藏按钮、打开 Organizer 窗口按钮，如图 1-4 所示。



图 1-4 工具栏

导航器区域有项目导航面板、符号导航面板、搜索导航面板、问题导航面板、调试导航面板、断点导航面板、日志导航面板。不同面板提供不同配置，供开发者从不同的视角查看项目，单击导航器区域顶部的图标可以在不同导航面板中进行切换，如图 1-5 所示。

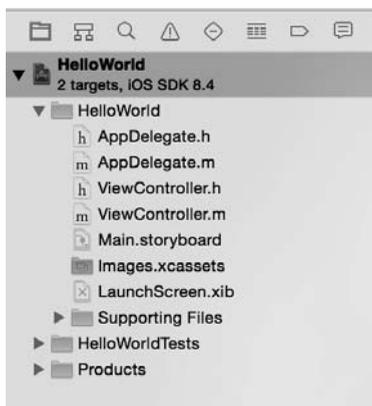


图 1-5 导航器区域

编辑器区域主要进行编辑源代码文件，如图 1-6 所示。

```
//  
// main.m  
// HelloWorld  
//  
// Created by sirwan on 15-2-28.  
// Copyright (c) 2015年 __FULLUSERNAME__. All rights reserved.  
//  
  
#import <UIKit/UIKit.h>  
  
#import "HelloWorldAppDelegate.h"  
  
int main(int argc, char * argv[])  
{  
    @autoreleasepool {  
        return UIApplicationMain(argc, argv, nil, NSStringFromClass([HelloWorldAppDelegate class]));  
    }  
}
```

图 1-6 编辑器区域

实用工具区域主要用于打开、关闭实用工具面板，与检查器面板类似，它也是上下文相关的，其内容随着编辑器面板的显示内容而变化，如图 1-7 所示。

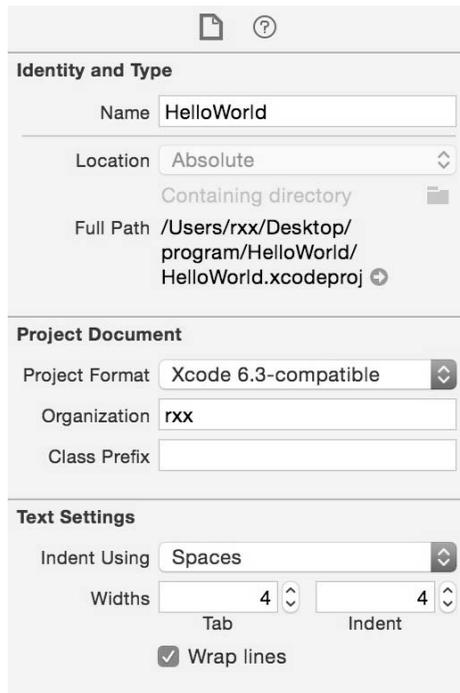


图 1-7 实用工具区域

1.3.2 新建Xcode项目

【步骤 1】打开 Xcode 应用程序。第一次创建或打开 Xcode 项目会出现一个“Welcome to Xcode”欢迎窗口，如图 1-8 所示。以后创建或打开 Xcode 项目时会出现一个项目窗口。



图 1-8 “Welcome to Xcode” 欢迎窗口

【步骤 2】新建应用程序窗口。在“Welcome to Xcode”欢迎窗口中单击“Create a new Xcode project”行，或选择“File”→“New”→“New project”选项（或按快捷键 Shift+Command+N），Xcode 将打开一个新窗口并显示对话框（如图 1-9 所示），在左侧选择“Application”选项，在右侧选择“Single View Application”选项，然后单击“Next”按钮，弹出一个新对话框。

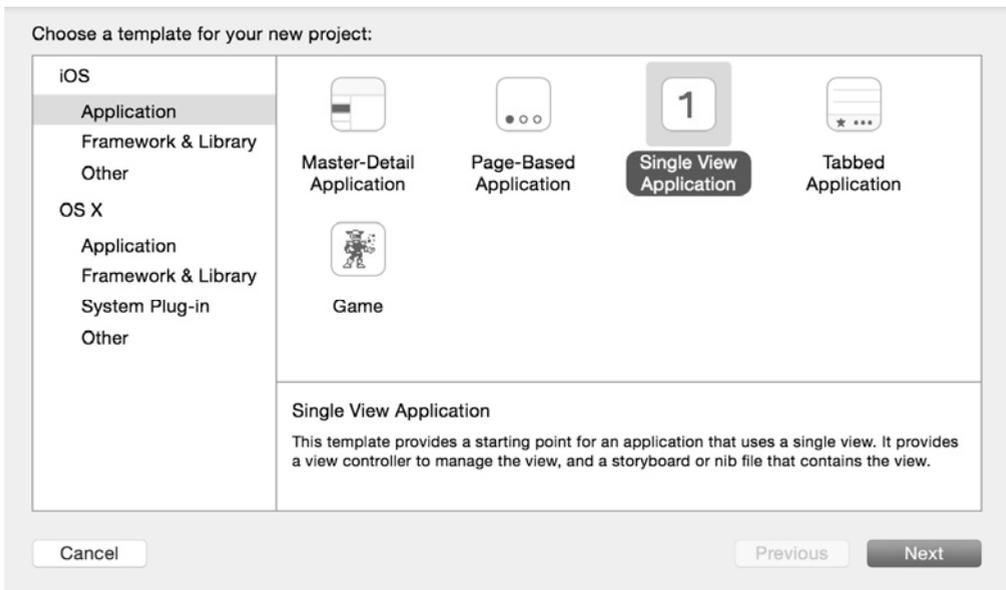


图 1-9 新建应用程序对话框

Xcode 中内置了一些应用程序模板，可以使用这些模板开发常见类型的 iOS 应用程序。如“Tabbed Application”模板可以创建类似 iTunes 的应用程序，“Master-Detail Application”

模板可以创建类似邮件的应用程序。

【步骤 3】填写项目信息。在“Product Name”栏中输入项目名（如 Hello World），在“Organization Name”栏中输入机构名称（如用户的英文名称或公司机构的简写），在“Organization Identifier”栏中输入机构标识符，一般形式为 com.Company Name，在“Language”栏中选择“Objective-C”，在“Devices”栏中选择“iPhone”，如图 1-10 所示。

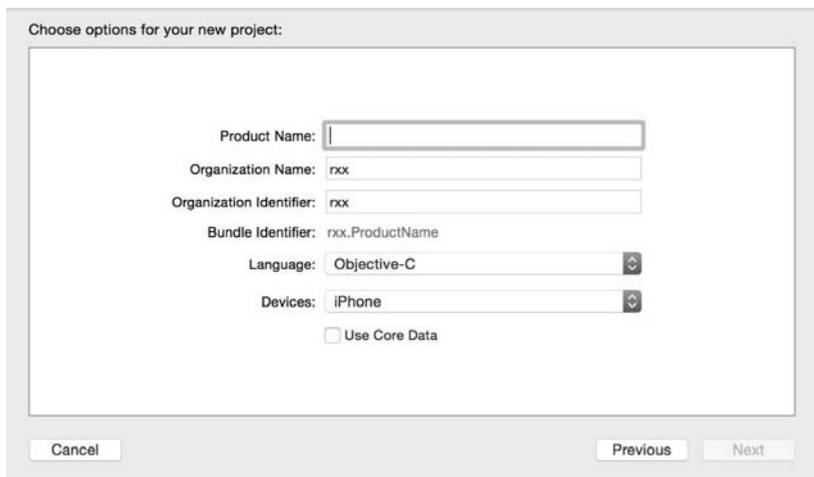


图 1-10 填写项目信息对话框

【步骤 4】单击“Next”按钮，弹出一个新的对话框，用来指定项目存储的位置，如图 1-11 所示。不选择“Source Control”选项，然后单击“Create”按钮，将弹出新的项目窗口，如图 1-12 所示。

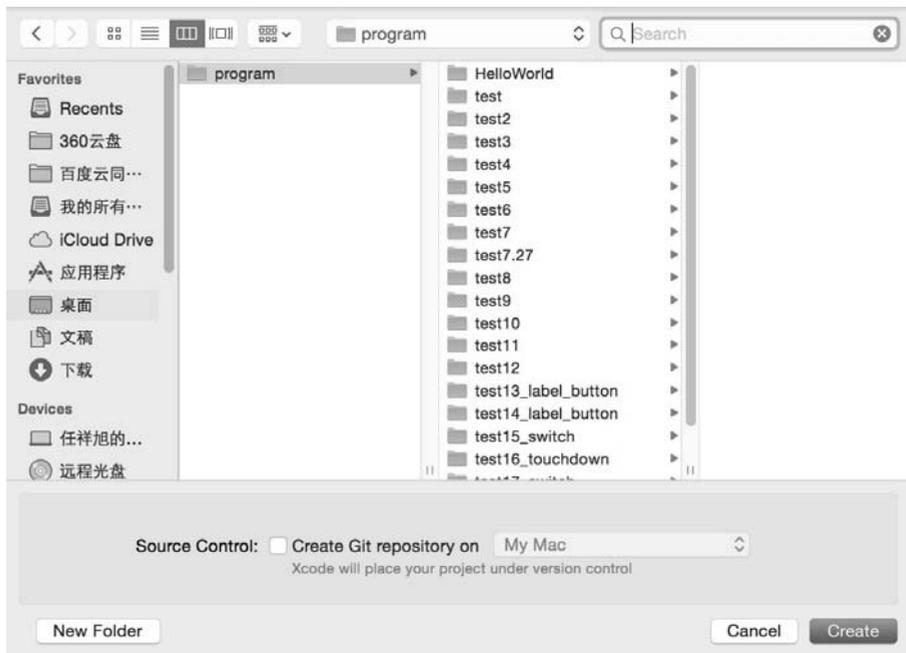


图 1-11 指定项目存储的位置窗口

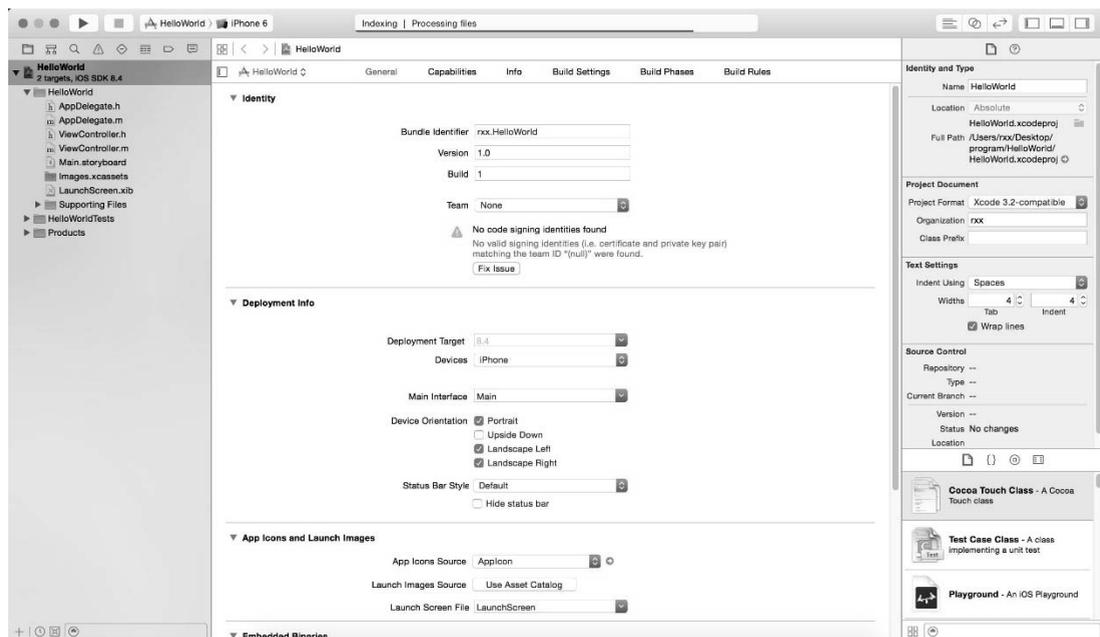


图 1-12 打开新项目窗口

1.3.3 运行应用程序

即使不编写任何代码，也可以构建应用程序。运行应用程序有两种方式：**iPhone Simulator SDK** 和 **iPhone Device SDK**。**iPhone Simulator SDK** 通过此 SDK 来构建的应用程序运行在 **iPhone Simulator**（模拟器）下，**iPhone Device SDK** 通过此 SDK 来构建的应用程序运行在 **iPhone Device**（iPhone 设备）下。如果没有 **iPhone Device**，建议使用 **iPhone Simulator**，模拟器可模拟应用程序在 **iPhone** 设备上运行，让开发者了解应用程序的外观和行为。操作如下。

【步骤 1】在 Xcode 工具栏的“Scheme”弹出式菜单中选择“HelloWorld”→“iPhone Retina(4-inch)”选项。如果弹出式菜单中该选项未被选定，可以将它打开，然后从菜单中选择“iPhone Retina(4-inch)”选项，如图 1-13 所示。

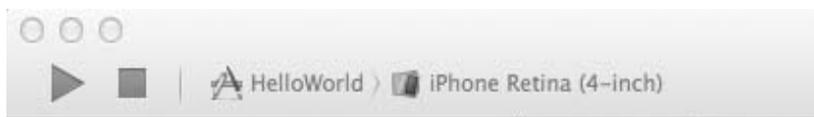


图 1-13 设置运行模拟设备

【步骤 2】单击 Xcode 工具栏中的“Run”按钮，或选择“Product”→“Run”选项。在 Xcode 生成项目后，模拟器会自动启动。因为指定的是 **iPhone** 产品而非 **iPad** 产品，所以模拟器会显示一个类似 **iPhone** 的窗口。在模拟 **iPhone** 屏幕上，用模拟器打开应用程序，运行结果如图 1-14 所示。

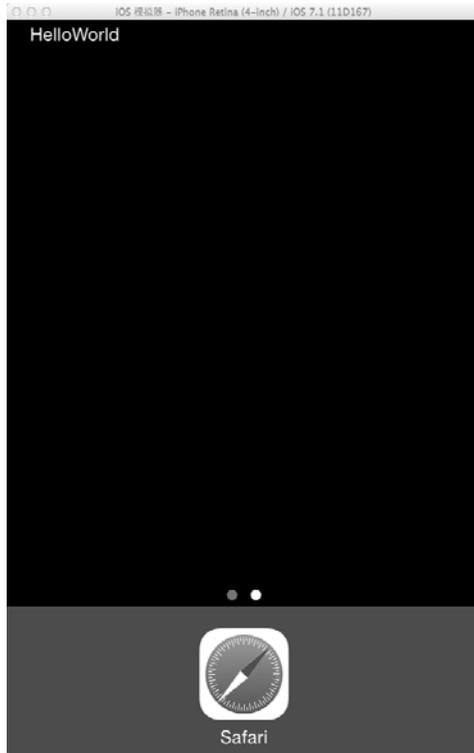


图 1-14 程序运行结果

1.4 iOS应用程序结构分析

一个简单的 Xcode 项目 HelloWorld 包含 HelloWorld、Supporting Files、HelloWorldTests、Products 等文件夹，如图 1-15 所示。



图 1-15 项目文件结构

HelloWorld 文件夹是第一个文件夹，以项目名来命名，包含了应用程序的大部分代码以及用户界面文件，可以在该文件夹下创建子文件夹。

Xcode 中扩展名为 .m 的文件是源程序文件，相当于 .java、.c、.cpp 等。

Xcode 中扩展名为 .h 的文件是头文件，文件 HelloWorldAppDelegate.h 和 HelloWorldAppDelegate.m 是实现应用程序委托。委托是负责为其他对象处理特定任务的对象，能够在某些预定时间点为 UIApplication 类调用。这两个文件是由 Single View Application 模板提供的。

HelloWorldViewController 是视图控制器，由 Single View Application 模板提供，应用程序启动时首先载入视图控制器，视图控制器由 HelloWorldViewController 对象来管理。

Main.storyboard 文件主要用于描述应用程序中有哪些界面、界面有哪些控件和事件以及界面之间的导航关系。

Images.xcassets 主要用于存放图片资源，用户的图片资源可以存放在这里，使用的时候进行调用即可。

Supporting Files 文件夹包含项目中必需的非 Objective-C 类的源代码文件和资源文件。

HelloWorld-Info.plist 文件是一个属性列表，包含应用程序相关的各种信息。

Info.plist 文件是文本文件，包含可能被信息属性列表引用到的可读字符串。

main.m 文件为整个应用程序执行的入口文件。main.m 文件中有 main() 方法，通常不需要编辑或修改该文件。

HelloWorldTests 是测试文件夹，对程序测试时使用。本书中基本不涉及这方面的知识。

Products 文件夹包含构建项目时生成的应用程序。展开该文件夹，可以看到一个名为 HelloWorld.app 的文件，它就是项目创建出来的应用程序。

本章小结

本章讲述了 iOS 的发展历程、功能特性和 iOS7 的新特性，重点介绍了 iOS 开发环境的搭建、创建，并运行第一个 iOS 应用程序，分析了 iOS 应用程序的结构。后面的章节将会依据本章重点内容由易到难逐步展开介绍，可以基于 iOS 框架开发出更多、更好的应用程序。

通过本章的讲解，读者可以对 iOS 编程有一个基本的认识，并且掌握 Xcode 开发工具的基本操作界面。

习题 1

1. 在 Apple 官网下载最新 Xcode，搭建 iOS 开发环境。
2. 熟悉 Xcode 工具，掌握各工作区域的功能。
3. 在 Xcode 中新建一个项目，并实现 1.3 节的内容。