《iOS编程(第四版)》

作者: [美] Christian Keur

牎杉俧恵

- 书名:《ios编程(第4版)》
- 原书名: iOS Progarmming, 4th Edition
- ISBN: 978-7-5609-9790-2
- 作者: [美] Christian Keur / [美] Aaron Hillegass / [美] Joe Conway
- 译者:丁道骏/张召/吴春燕
- 出版社: 华中科技大学出版社
- 版权所有 侵权必究

佢耋箆仑

Christian Keur是Big Nerd Ranch的高级讲师和软件工程师,负责编写Big Nerd Ranch的"OS新手培训课程"教材。该教材广受好评,是本书的原型。Christian毕业于美国佐治亚理工学院计算机科学系,目前居住在亚特兰大。

Aaron Hillegass是Big Nerd Ranch的创始人之一,曾就职于NeXT公司和Apple公司,他拥有近20年的 Objective-C、Cocoa、iOS开发与教学经验。Aaron与他人合著了《Mac OS X编程》和《Objectibe-C编程》。

Joe Conway曾参与编写了"iOS新手培训课程"教材。他最近创办了stable/kernel公司,开发高质量的移动应用。

笄1竦 笄7丰箆卛皊i0S庚甮

本章介绍如何编写一个简单的iOS应用——Quiz,功能为:在视图中显示一个问题,用户点击视图 下方的按钮,可以显示相应的答案,用户点击上方的按钮,则会显示一个新问题(见图1-1)。

图1-1 第一个应用: Quiz

编写iOS应用时,读者必须先处理以下两个基本问题。

·如何创建并设置对象(例如,在某处放置一个按钮并将其标题设置为"显示问题")?

·如何处理用户交互(例如,在用户按下某个按钮时执行某段代码)?

本书会用大量的篇幅回答这两个问题。

在阅读第1章时,请读者尽量走完整个流程,但不用试图搞懂每一个细节。模仿是一种有效的学习 方式。可以通过模仿学会说话,也可以通过模仿学习iOS编程。等读者熟悉开发环境后,可以再尝试自 行开发应用。现在,还请读者跟着本章照做,后续章节会详细介绍细节。

1.1 创建Xcode项目

运行Xcode,在File菜单中选择New→Project...。

Xcode会显示新的工作空间(workspace)窗口,同时工具栏(toolbar)处会弹出下拉窗口 (sheet)。选择位于下拉窗口左侧iOS栏下的Application(见图1-2),右侧有若干应用模板可供选择。 请读者选择Empty Application(空应用)。

图1-2 创建新项目

空应用模板几乎没有多余的代码,而其他模板会生成很多通用代码。这些代码虽然能帮助开发应 用,但是对于初学者,弊大于利。

本书中的项目都是使用Xcode 5.0.2创建的。Apple公司未来发布新版Xcode时,这些模板的名称可能 有改动。读者在选择模板时,如果没有找到空应用模板,则可以选择一种看上去最简单的模板,例如 Single View Application(单视图应用)。还可以访问本书原作者提供的论坛: http://forums.bignerdranch.com,以获取帮助。

单击Next按钮,在新出现的界面中,将Quiz填入Product Name文本框(见图1-3)。Organization Name和Company Identifier文本框也是必填的,读者可以分别填入Big Nerd Ranch和combignerdranch,也可以填入自己的公司名称和公司的反向域名,如compourcompanynamehere。

将BNR填入Class Prefix文本框,在标题为Devices的弹出式菜单中选择iPhone。确保标题为Use Core Data的选择框未被选中。

图1-3 设置新项目

虽然之前将Quiz项目的设备类型设置为了iPhone,但是生成的应用也能在iPad上运行。在iPad 上,Quiz会在iPhone屏幕大小的窗口中运行,但不能充分利用iPad的大屏幕。对于一个用于学习的示例应 用,这不是大问题。本书前半部分的应用都会使用基于iPhone设备的模板,并将重心放在学习iOS SDK 的基础知识上。无论是哪种iOS设备,这些内容都是相同的。后面会介绍一些iPad独有的特性,以及如 何编写在iPhone和iPad这两种设备上都能全屏运行的原生应用。

单击Next按钮后,Xcode会显示最后一个界面,提示读者保存项目。请准备好保存本书所有代码的 目录,然后将Quiz项目保存在该目录下。本书不会介绍选择框Create local git repository for this project的作 用,勾选或取消都可以。单击Create按钮,Quiz项目就创建好了。

项目创建完毕后,Xcode会显示工作空间窗口(见图1-4)。



位于工作空间窗口左侧的是导航面板区域(navigator area),负责显示各种不同的导航面板。这些 导航面板能分别显示项目的某些特定部分。单击导航面板选择条(位于导航面板区域上方)中的某个图标,可以选择相应的导航面板。

在Quiz项目工作空间中,当前选中的导航面板应该是项目导航面板(project navigator),项目导航 面板的作用是显示项目中的文件(见图1-5)。读者可以尝试选中任意一个文件,文件会在导航面板区 域右边的编辑区域(editor area)中打开。 项目导航面板中的文件可以按目录分组,以帮助整理项目。Xcode模板已经为Quiz项目创建了若干 组。读者可以随意修改组名或增加新的组。项目导航面板中的组只用来整理文件,与文件系统无关。

图1-5项目导航面板列出的Quiz项目中的文件

在项目导航面板中,找到名为BNRAppDelegate.h和BNRAppDelegate.m的两个文件。它们被称为 BNRAppDelegate的类(class)文件,是空应用模板自动创建的。

一个类(class)表示一种对象(object)。iOS开发是面向对象的,每个iOS应用都可以看成是由一系列协同工作的对象构成的。Quiz应用启动时,系统将创建一个BNRAppDelegate对象。BNRAppDelegate 对象有时也称为BNRAppDelegate类的一个实例(instance)。

读者将在第2章中学习更多关于类和对象的知识。现在,还请读者关注iOS设计和开发的基本理论,继续完成本章的Quiz应用。

1.2 模型-视图-控制器

模型-视图-控制器(Model-View-Controller),简称MVC,是iOS开发中频繁使用的一种设计模式。 其含义是,应用创建的任何一个对象,其类型必定是模型对象、视图对象或控制器对象三种类型中的一种。

•视图对象是用户可以看见的对象,例如按钮、文本框、滑动条。视图对象用来构建用户界面,在 Quiz应用中,显示问题和答案的标签以及标签下方的按钮都是视图对象。

•模型对象负责存储数据,与用户界面无关。Quiz应用中的模型对象是两个包含字符串对象的数组: questions数组和answers数组。

•通常情况下,模型对象表示真实世界中与用户相关的事物。例如,读者要为一家保险公司开发应用,那么很可能会设计一个InsurancePolicy(保险协议)类的模型对象。

•控制器对象扮演"管家"的角色,它用于控制视图对象为用户呈现的内容,以及负责确保视图对象 和模型对象的数据保持一致。

一般来说,控制器用来回答:然后会发生什么?例如,用户从列表中选择了一项之后,控制器负责 呈现接下来应该看到的内容。

图1-6显示的是应用响应用户操作的流程,例如用户点击了应用界面上的一个按钮。

图1-6 MVC设计模式

请读者注意,模型对象和视图对象之间没有直接产生联系,而是由控制器对象负责彼此间的消息发送和数据传递。

1.3 设计Quiz

读者将使用MVC设计模式开发Quiz应用。以下列出了开发中需要使用的对象:

•四个视图对象: UILabel和UIButton的对象各两个。

•两个控制器对象: BNRAppDelegate和BNRQuizViewController的对象各一个。

•两个模型对象:NSArray的对象两个。

图1-7显示的是Quiz应用的对象图,图中勾勒出了上述对象和相互关系。

图1-7 Quiz应用的对象图

图1-7中展示了完成后的Quiz应用是如何工作的。例如,当用户按下Show Question按钮时,会触发 BNRQuizViewController对象的一个方法(method)。方法与非面向对象语言中的函数(function)类似, 都是一系列需要执行的命令。这个方法会从questions数组里取出一道新题目,然后通过位于视图上方的 标签将题目显示出来。

读者现在可能还无法完全看懂这幅对象图,没关系,到本章结尾再回来看这幅图时,就能深刻理解 Quiz应用的工作原理了。

现在请读者跟着本章一步步开发Quiz应用。第一步是创建控制器对象,应用的核心控制器——BNRQuizViewController。

1.4 创建视图控制器

空应用模板已经创建好了BNRAppDelegate类文件,现在请读者创建BNRQuizViewController类文件。 我们将在第2章和第6章介绍更多关于类和视图控制器的知识。现在,还请读者跟着本章照做。

在File菜单中选择New→File...Xcode会弹出选择文件模板的下拉窗口。选择位于下拉窗口左侧iOS栏下的Cocoa Touch,再选择右侧的Objective-C class(Objective-C 类)并单击Next按钮(见图1-8)。

图1-8 创建Objective-C 类

在新出现的窗口中,将BNRQuizViewController填入Class文本框,然后点击Subclass of文本框的下拉按钮,在弹出式菜单中选择UIViewController。勾选标题为With XIB for user interface的选择框(见图1-9)。

图1-9 创建视图控制器

单击Next按钮,Xcode会提示读者为BNRQuizViewController类文件选择保存位置。为项目创建一个新的类时,通常应该将类文件保存在对应的项目文件夹中,Xcode会默认选中当前项目文件夹,也可以点击标题为Group的弹出式菜单,菜单中显示了与项目导航面板中对应的组,读者可以选择想要保存类文件的组。因为组只用来整理文件,而Quiz项目又很简单,所以全部使用默认设置就可以了。

请读者确保选中了Targets中Quiz前的选择框(见图1-10)。选中后,Xcode在构建项目时会一起编译 BNRQuizViewController类文件。最后单击Create按钮。

图1-10 选中Targets中的Quiz

1.5 创建界面

在项目导航面板中找到BNRQuizViewController类文件。读者在创建这个类时勾选了With XIB for user interface,因此Xcode自动生成了第三个文件:BNRQuizView-Controller.xib。请读者选择BNRQuizViewController.xib。

Xcode会使用Interface Builder(界面创建工具)打开XIB(发音"zb")文件。Interface Builder是一种可 视化编辑器,可以用拖动对象的方式创建图形用户界面。XIB的全称就是XML Interface Builder。

很多其他平台的GUI创建工具,需要先描述应用外观,然后单击某个按钮,生成大量代码。Interface Builder则不同,它是一种对象编辑器:使用时需要先创建并设置对象,例如按钮和标签,然后将对象保 存在固化文件里。这种固化文件就是XIB文件。

Interface Builder将编辑区域分为两部分: 左侧是dock, 右侧是画布。

dock可以使用两种方式展示XIB文件中的对象:图标视图(icon view)和大纲视图(outline view)。 图标视图能为屏幕腾出更多的空间,但是出于学习目的,使用大纲视图查看对象更加方便。

如果读者的dock处于图标视图状态,请点击icon/outline视图切换按钮将dock切换到大纲视图状态,按钮在画布左下角的位置(见图1-11)。

图1-11 使用Interface Builder编辑XIB文件

大纲视图中的BNRQuizViewController.xib包含三个对象: Placeholders模块中的两个占位符对象 (placeholder)和一个View对象。请读者先忽略占位符对象,我们稍后再介绍它。

View对象是一个UIView类的对象,读者可以在画布上看到它。它是用户界面的基石,可以容纳其他 视图对象。而且,读者在画布上看到的和应用正式运行时的外观完全相同。

单击大纲视图中的View对象,它会显示在画布上。拖曳移动视图,移动视图不会修改对象自身,只 会重新组织画布。单击视图左上角的x按钮可以关闭视图。同样,这样做不会删除视图,只是将其从画 布中移除。再次选中大纲视图中的View对象,可将其加回画布。

现在Quiz应用的用户界面只有一个空白的视图对象,还需要添加两个标签和两个按钮(见图1-12)。

图1-12 需要的标签和按钮

创建视图对象

要加入这些视图对象,需要打开工具区域(utilities area)的对象库(object library)。

工具区域位于编辑器区域的右侧,分上下两部分:检视面板(inspector)和库面板(library)。上方的检视面板负责显示编辑器区域当前选中的文件或对象的各种设置。下方的库面板则会列出可以加入文件或项目的物件(例如对象或代码)。拖曳这两个区域间的分隔条,可以改变其相对的尺寸。

这两个面板的顶部都有一选择条,可以用来选择各种不同类型的面板和库(见图1-13)。在库面板 选择条中,单击图标,可以打开对象库面板。

图1-13 Xcode工具区域

对象库中的对象主要用于构建用户界面,它们可以被拖曳到XIB文件中。

请读者在对象库中找到Label对象(应该可以在列表顶部找到Label对象,如果没有,可以往下滚动 列表,或者使用面板底部的查询框),然后选中该对象并拖曳至(画布上的)视图对象上,再将这个 Label对象放在视图正中间靠近顶部的位置。拖曳第二个Label对象,放在视图正中间靠近底部的位置。

接下来请找到Button对象,拖曳两个Button对象至视图对象,并分别放在两个Label对象的下方。

现在读者已经为BNRQuizViewController.xib添加了四个新的视图对象,在编辑区域的大纲视图中可以 看到它们。

设置视图对象

视图对象已经全部创建好了,接下来应该设置对象的属性。部分属性,例如大小、位置和文本可以 在画布中直接编辑。其他属性则必须通过属性检视面板(attributes inspector)来编辑,我们很快就会介 绍它。

请读者修改以上四个对象的大小,使其宽度能够横跨窗口的大部分区域(在画布或大纲视图中选中 对象,然后拖曳它的角或边,就可以修改对象的大小,如图1-14所示)。

图1-14 修改标签和按钮的大小

双击画布中的某个Button对象,就可编辑该对象的标题。将上方Button对象的标题改为Show Question(显示问题),下方的改为Show Answer(显示答案)。使用同样的方法可编辑Label对象的文 字。删除上方Label对象的文字(之后用于显示问题),下方的设置为???。这时的界面如图1-15所 示。

图1-15 设置标签和按钮的文本

如果标签中的文本处于居中对齐状态,则界面会更加美观。设置标签的文本对齐方式必须通过属性 检视面板来编辑。

选中位于视图下方的Label对象,单击检视面板选择条中的图标,打开属性面板。

在属性面板中找到标题为Alignment的分段控件(segmented control)。选择中间的那个选项(居中对 齐),如图1-16所示。

图1-16 设置label的文本对齐方式为居中对齐

修改后请读者观察画布中的变化,视图下方Label的文字???会改为居中对齐。下面将视图上方的Label也设置为居中对齐(该对象目前还没有文字,但是运行时会有)。

为了让用户知道按钮的点击区域,可以改变按钮的背景颜色。首先请在画布中选择Show Question按钮。

在属性检视面板中向下滚动,会看到View标题下方的属性。在Background标签右侧,点击颜色面板

欢迎访问: 电子书学习和下载网站(https://www.shgis.cn)
文档名称: 《《iOS编程(第四版)》》[美] Christian Keur 著.epub
请登录 https://shgis.cn/post/293.html 下载完整文档。
手机端请扫码查看:

