



程序员最可信赖的求职帮手

- Amazon榜首畅销书
- 全面覆盖Android开发知识点
- 全真示例、循循善诱、轻松上手

Android 编程权威指南

[美] Bill Phillips / Brian Hardy 著
王明发 译

Android Programming
The Big Nerd Ranch Guide



人民邮电出版社
POSTS & TELECOM PRESS

图灵社区会员 zwjjbb(zwjjbb@qq.com) 专享 尊重版权



程序员最可信赖的求职帮手

数字版权声明

图灵社区的电子书没有采用专有客户端，您可以在任意设备上，用自己喜欢的浏览器和PDF阅读器进行阅读。

但您购买的电子书仅供您个人使用，未经授权，不得进行传播。

我们愿意相信读者具有这样的良知和觉悟，与我们共同保护知识产权。

如果购买者有侵权行为，我们可能对该用户实施包括但不限于关闭该帐号等维权措施，并可能追究法律责任。



程序员最可信赖的求职帮手



Bill Phillips Big Nerd Ranch资深Android讲师、高级软件工程师。他与Brian Hardy合作，为Big Nerd Ranch开发了广受好评的5天Android训练营培训课程。Bill擅长透彻地理解事物的本质，并帮助其他人做到这一点。闲暇时间，Bill喜欢阅读和弹钢琴。



Brian Hardy Big Nerd Ranch首席软件工程师、资深讲师。Big Nerd Ranch的Android、iOS和Ruby培训课程均由Brian设计开发。闲暇时间，Brian喜欢骑自行车和听音乐。



程序员最可信赖的求职帮手

程序设计丛书



Android 编程权威指南

[美] Bill Phillips Brian Hardy 著
王明发 译

Android Programming
The Big Nerd Ranch Guide



人民邮电出版社

北 京

图灵社区会员 zwjbb(zwjbb@qq.com) 专享 尊重版权

图书在版编目 (C I P) 数据

Android编程权威指南 / (美) 菲利普斯
(Phillips, B.), (美) 哈迪 (Hardy, B.) 著 ; 王明发译.
— 北京 : 人民邮电出版社, 2014. 4
(图灵程序设计丛书)
书名原文: Android programming: the big nerd
ranch guide
ISBN 978-7-115-34643-8

I. ①A… II. ①菲… ②哈… ③王… III. ①移动终
端—应用程序—程序设计 IV. ①TN929.53

中国版本图书馆CIP数据核字 (2014) 第027899号

内 容 提 要

Big Nerd Ranch 是美国一家专业的移动开发技术培训机构, 本书主要以训练营的 5 天教学课程为基础, 融合了两位作者多年的心得体会, 是一本完全面向实战的 Android 编程权威指南。全书共 37 章, 详细介绍了 GeoQuiz、HelloMoon、DragAndDraw 等 8 个 Android 应用。这些应用的难易程度不一, 最复杂的 CriminalIntent 应用占用了 13 章的篇幅。通过这些精心设计的应用, 读者可掌握很多重要的理论知识和开发技巧, 获得最前沿的开发经验。

如果你熟悉 Java 语言, 或者了解对面向对象编程, 那就立刻开始 Android 编程之旅吧!

-
- ◆ 著 [美] Bill Phillips Brian Hardy
译 王明发
责任编辑 李 瑛
执行编辑 李 静 邢 妍
责任印制 焦志炜
- ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京 印刷
- ◆ 开本: 800×1000 1/16
印张: 34
字数: 798千字 2014年4月第1版
印数: 1-4 000册 2014年4月北京第1次印刷
- 著作权合同登记号 图字: 01-2013-3661号
-

定价: 99.00元

读者服务热线: (010)51095186转600 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京崇工商广字第 0021 号

图灵社区会员 zwjjbb(zwjbb@qq.com) 专享 尊重版权



程序员最可信赖的求职帮手

版权声明

Authorized translation from the English language edition, entitled *Android Programming: The Big Nerd Ranch Guide, First Edition* by Brian Hardy, Bill Phillips, published by The Big Nerd Ranch(Aaron Hillegass). Copyright © 2012 by The Big Nerd Ranch(Aaron Hillegass).

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Posts and Telecom Press(Turing Book Company). Copyright Posts and Telecom Press (Turing Book Company).

版权所有。未经出版人事先书面许可，对本出版物的任何部分不得以任何方式或途径复制或传播，包括但不限于复印、录制、录音，或通过任何信息存储和检索系统。

本书中文简体字版由The Big Nerd Ranch(Aaron Hillegass)授权人民邮电出版社（北京图灵文化发展有限公司）独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。



程序员最可信赖的求职帮手

献词

献给Donovan。他很忙，但他知道什么时候该用Fragment。

——B.H.



程序员最可信赖的求职帮手

致 谢

我们很不安，因为封面上只印了我们两个的名字。事实上，本书能够出版发行，完全是团队合作的成果。我们满怀感激之情。

- ❑ 感谢Chris Stewart和Owen Matthews为本书部分章节提供了大量基础性内容。
- ❑ 感谢我们的同事Chris Stewart和Christopher Moore老师。他们在使用持续更新的教学材料教学时充满耐心，并针对其中的内容提出了改进建议；在我们准备做出重大修订时，他们为我们提供了宝贵的参考意见。
- ❑ 感谢我们的同事Bolot Kerimbaev和Andrew Lunsford。他们提供的反馈意见，在很大程度上促成了我们多用Fragment的决定。
- ❑ 感谢技术审校团队帮我们找出并修正了多处问题。他们是Frank Robles、Jim Steele、Laura Cassell、Mark Dalrymple和Magnus。
- ❑ 感谢Aaron Hillegass。他的绝对信任给了我们很大的源动力，否则我们也没机会出版这本书。（他还为我们提供了资金支持，好人一个。）
- ❑ 感谢我们的编辑Susan Loper。通过她出色的编辑润色，原来程序注释般不修边幅的文稿，一下子变得简洁通畅了，而且那些冷笑话也变得栩栩如生，富有了启发性；原来不太恰当的幽默也变得相得益彰。正是因为她的努力，本书才如此妙趣横生。她告诉了我们在技术写作中，什么叫清晰明白和通俗易懂。
- ❑ 感谢美国航空航天局。相比伟大的太阳系探索工程，本书显得微不足道。
- ❑ 感谢Ellie Volckhausen为本书设计了封面。
- ❑ 感谢网站的Chris Loper。他设计并制作了本书的纸版、Epub版和Kindle版。他使用的DocBook工具给本书的设计与制作带来了极大便利。
- ❑ 感谢Facebook的员工们。他们在本教程的学习中，为我们提供了很多很好的反馈意见。

最后感谢我们的学员们。限于篇幅，这里无法一一列出他们的名字。在本书的创作过程中，他们帮助我们纠正错误，并提出了宝贵建议。正是他们旺盛的求知欲和不断的困惑，让我们有动力编写这本书，再次表示感谢。



程序员最可信赖的求职帮手

如何学习Android开发

学习Android开发，对每个新手都是一个很大的挑战，就好像在异国他乡学会生存一样。即使会说当地的语言，一开始也绝不会有在家的感觉，因为你不能完全理解周围人理解的东西。原有的知识储备在新环境下可能完全派不上用场。

Android有自己的语言文化，亦即Java语言。但仅掌握Java还远远不够，还需要学习很多新的理论和技术知识来理清头绪，从而指引你穿越陌生的领域。

该由我们登场了。在Big Nerd Ranch，我们相信，要成为一名合格的Android开发人员，必须做到：

- ❑ 着手开发一些Android应用；
- ❑ 彻底理解你的Android应用。

本书将协助你完成以上两件事情，我们已用它成功培训了数百位专业的Android开发人员。本书将指导你完成多个Android应用开发，并根据需要逐步介绍各种理论概念及技术知识。在学习过程中，如果遇到知识疑难点，请勇敢面对，我们也会尽最大努力抽丝剥茧，让你知其然更知其所以然。

我们的教学方法是：在学习理论的同时，就着手运用它们开发实际的应用，而非先学习一大堆理论，再考虑如何将理论应用于实践。

读完本书，你将具备必要的开发经验及知识，成长为一名Android开发者。以此为起点，你就能够进行实际开发并继续深入学习。

本书读者对象

使用本书，你需要熟悉Java语言，包括类、对象、接口、监听器、包、内部类、匿名内部类、泛型类等基本概念。

如果你对这些概念感到陌生，那么你很可能在翻到第二页时就已经无法再读下去了。对此，建议先放下本书，找本Java入门书看一看。市面上有很多优秀的Java入门书，你可以基于自己的编程经验及学习风格去挑选。

如果你熟悉面向对象编程，但Java知识已经忘得差不多了，那么阅读本书可能也不会有太大的问题。对于接口、匿名内部类等重要的Java语言点，我们会做必要的简短回顾。建议在学习过程中手边备上一本Java参考书，方便查阅。

如何使用本书

本书基于Big Nerd Ranch培训基地的5天教学课程编写而成。课程从基础知识讲起，各章节内容以循序渐进的方式编排，建议不要在章节间跳读，以免学习效果大打折扣。因此本书不适合作为参考书。本书旨在让你跨越学习的初始障碍，能够充分利用其他各种参考资料和代码实例类图书来深入学习。

我们的学员在学习期间也受益于良好的培训环境：专门的培训教室、可口的美食、舒适的住宿条件、动力十足的学习伙伴，以及一位随时答疑解惑的指导老师。

本书读者同样需要类似的良好环境。应保证充足的睡眠，找一个安静的地方开始学习。参考以下建议也很有帮助：

- (1) 组织朋友或同事组成兴趣小组学习；
- (2) 集中安排时间逐章学习；
- (3) 参与本书交流论坛（forums.bignerdranch.com）的讨论；
- (4) 寻求Android开发高手的帮助。

本书内容

通过本书，我们将学习开发8个Android应用。有些应用很简单，一章即可讲完。有些相对复杂。最复杂的一个应用跨越了13章。通过这些精心编排的应用，可学到很多重要的理论知识和开发技巧，从中获得最直接的开发经验。

❑ GeoQuiz

本书第一个应用，通过它学习Android应用的基本组成、activity、界面布局（layout）以及显式intent。

❑ CriminalIntent

本书最复杂的应用，用来记录办公室同事的种种陋习。通过本应用学习fragment、master-detail用户界面、list-backed用户界面、菜单选项、相机调用、隐式意图（implicit intent）等内容。

❑ HelloMoon

通过阿波罗登月历史事件资料的媒体播放应用，继续深入学习fragment、媒体文件的播放与控制、应用资源及本地化的配置。

❑ NerdLauncher

通过个性化启动器的开发，深入学习Android的意图（intent）以及任务（task）的概念知识。（task也可称作Activity栈。）

❑ RemoteControl

通过小巧的示例应用，学习使用样式（style），状态列表绘制（state list drawable）以及其

他一些工具，创建更吸引人的用户界面。

❑ PhotoGallery

通过Flickr网站接口下载并显示照片的客户端应用，借此学习Android服务、多线程、网络内容获取服务等知识。

❑ DragAndDraw

简单的画图应用，通过它学习触摸手势事件处理以及创建个性化视图等知识。

❑ RunTracker

定位追踪并在地图上显示环城或环球旅行线路的应用。借此应用学习使用定位服务、SQLite数据库、加载器（loader）以及地图调用。

挑战练习

大部分章末尾都配备有练习题。可借此机会学以致用，查阅官方文档，锻炼独立解决问题的能力。

我们强烈建议大家完成这些挑战练习。在练习过程中，尝试另辟蹊径，探索自己独特的学习之路，有助于巩固所学知识，增强未来开发应用的信心。

遇到一时难以解决的问题，请随时访问论坛<http://forums.bignerdranch.com>寻求在线帮助。

深入学习

本书部分章末尾还包含一块标注为“深入学习”的内容，针对章节内的知识点，提供深入讲解或更多学习信息。本部分内容不属于必须掌握的部分，但我们也希望大家有兴趣阅读并有所收获。

代码风格

有别于其他Android开发学习社区常见的编码风格，我们有着自己的判断与选择，主要体现在以下三个方面。

❑ 我们在监听器代码部分使用匿名内部类

这通常取决于个人选择。我们认为使用匿名内部类可以让代码更简练，让监听器实现方法一目了然。不过在高性能要求的场景下，匿名内部类可能会有一些问题，但大多数情况下它们都工作得很好。

❑ 自第7章引入fragment后，后续所有用户界面都使用它

这一点，我们有充足的坚持理由。很多Android开发者仍然习惯于开发基于activity的代码。我们不打算墨守成规，相信我们，一旦适应了fragment，使用起来将不会太困难。相比activity，fragment在创建和显示用户界面时具有更加灵活的明显优势，因此值得为此付出努力。

❑ 我们开发兼容Gingerbread和Froyo设备的应用

随着Ice Cream Sandwich、Jelly Bean以及随后Key Lime Pie的推出，Android开发平台经历了不断的变化与升级。然而，事实上有半数在用设备依然运行着Froyo或Gingerbread系统。（第6章将介绍以美食命名的各个不同的Android开发版本。）

因而，明知有困难，我们还是特意选择开发向后兼容Froyo或至少是Gingerbread系统版本的应用。尽管在教学以及开发方面，开发最新版本系统应用要更加容易一些，但我们还是希望做好为实际在用设备开发的准备。要知道，目前仍有超过40%的Android设备运行着Gingerbread系统。

版式说明

为了方便读者阅读，本书会对某些特定内容采用专门的字体。变量、常量、类型、类名、接口名和方法名会以代码体显示。

所有代码与XML清单也会以代码体显示。需要输入的代码或XML总是以粗体显示。应该删除的代码或XML打上删除线。例如，在下列实现代码里，我们删除了makeText(...)方法的调用，增加了checkAnswer(true)方法的调用。

```
@Override
public void onClick(View v) {
    Toast.makeText(QuizActivity.this, R.string.incorrect_toast,
    Toast.LENGTH_SHORT).show();
    checkAnswer(true);
}
```

开发必备工具

ADT套件

准备开发前，需要Android开发工具（ADT）套件。

该工具套件包括下面这些。

❑ Eclipse

一套支持Android开发的集成开发环境。Eclipse本身使用Java代码开发，因此可以安装在PC、Mac、Linux等多个平台。Eclipse用户界面遵循“原生应用观感”法则（native look-and-feel），因此，开发工具界面可能会因系统环境的不同而与本书稍有差异。

❑ Android开发工具

Eclipse的一套插件。本书使用的ADT版本号为21.1。请确保使用相同或者更高版本的ADT。

❑ Android SDK

最新版本的Android SDK。

- ❑ Android SDK工具以及平台工具
用来测试与调试应用的一套工具。
- ❑ Android模拟器系统镜像
用来支持在不同虚拟设备上开发与测试应用。

开发套件的下载与安装

以ZIP格式打包的ADT开发套件可从Android开发者网站下载。

- (1) 访问网址<http://developer.android.com/sdk/index.html>下载开发套件。
- (2) 解压文件到指定的安装Eclipse及其他工具的路径。
- (3) 在解压后的文件中，找到并打开eclipse目录，运行该目录下的Eclipse程序。

如果是Windows系统，若遇到Eclipse无法正常启动的情况，请至网站下载安装JDK6。如仍存在问题的话，请访问网址<http://developer.android.com/sdk/index.html>寻找相应的帮助信息。

下载早期版本的SDK

ADT开发套件自带Android最新版本的SDK与系统模拟器镜像。但若想在Android早期版本上测试应用，还需额外下载相关工具组件。

可通过Android SDK管理器来配置安装这些组件。运行Eclipse，在Window菜单项下选择Android SDK Manager，具体请参见图0-1。

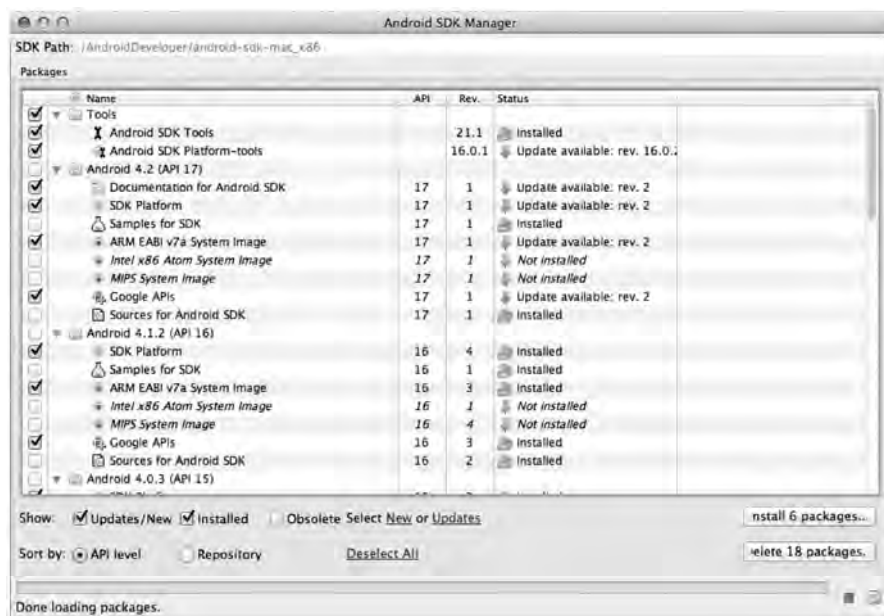


图0-1 Android SDK管理器

对于Android 2.2（Froyo）及其后的每一个系统版本，建议选择安装以下3个组件：

- ❑ SDK平台；
- ❑ 模拟器镜像；
- ❑ Google API。

下载这些组件需要一定时间，请耐心等待。

通过Android SDK管理器，还可以及时获取Android的更新信息，比如新的系统平台发布或工具版本更新等。

硬件设备

尽管模拟器用来测试应用很有用，我还是建议同时准备一台Android设备来运行开发应用。本书最后一个应用需要实体设备支持。



程序员最可信赖的求职帮手

译者序

2007年，苹果公司发布了革命性的iPhone，自此开启了智能手机的新时代。随后，2008年，谷歌公司和开放手机联盟共同开发并推出了Android智能手机。时至今日，移动智能设备不仅深刻影响了智能手机行业，也改变了整个科技产业以及人们生活的方方面面，在全球掀起巨大的移动浪潮。

迎着这股浪潮，国际巨头、创业公司、独立开发者各展身手，奋力搏击，抢登浪潮之巅。苹果一度成了全球市值最高的公司，三星则是卖出了数亿部Android手机。2013年，百度以19亿美元的价格收购了91无线。2014年开春，Facebook更是以190亿美元的天价收购了开发WhatsApp应用仅有50名员工的公司，直接把这股移动浪潮推到了巅峰。

豪不夸张地说，所有这一切都离不开移动软件的开发。目前主流的开发平台是苹果的iOS系统和谷歌的Android系统。凭借精美绝伦的UI、流畅顺滑的交互体验，iOS开发在早期抢占了先机。但如今，谷歌在Android系统UI设计及优化方面的前进步伐已赶超苹果在创新方面的进步。越来越多的软件人开始投身Android应用开发阵营。对于业余爱好者来说，Android还是iOS，喜欢谁就选谁。而对于专业开发人员，果粉也好，Android迷也罢，从职业发展及商业利益角度来说，掌握双平台开发是必须的。

作为荣获2012 Jolt 生产力大奖的iOS编程教程的姊妹篇，Big Nerd Ranch公司再接再厉，推出了这本《Android编程权威指南》。英文版甫一上架立即赢得了Amazon读者的广泛赞誉。本书基于Big Nerd Ranch公司的Bootcamp教程编写而成。Big Nerd Ranch创办于2001年，是美国一家知名IT培训公司，每年为微软、谷歌、Facebook等行业巨头培养众多专业人才。而BootCamp在英文中原意为美国海军陆战队新兵训练营，应用于IT培训行业中，意指通过全真IT项目实战，培训出像美国海军陆战队员那样优秀的IT人才。

本书适合有一定Java编程经验（至少熟悉Java）并对Android开发感兴趣的读者阅读。本书最大的特点是，从Android应用的基本概念及组成开始介绍直至完成一个复杂实用的谷歌地图应用，作者巧妙地把Android开发所需的庞杂知识、行业实践、编程规范等融入本书，并以一种润物无声的导学方式引领读者轻松完成全书的开发学习。第26章在讲解后台任务和线程时，作者寓教于乐，还精心安排了在鞋店工作的闪电侠案例，既能帮助读者形象地理解复杂抽象概念，又让人印象深刻，难以忘却。类似这样的案例、幽默全书俯拾皆是，还是等读者自己去发掘吧。另外，几乎每章都配有深入学习及难度逐步升级的挑战练习版块。深入学习意在让读者进一步掌握本章关键知识点并指明学习更高级主题的方向；挑战练习能够让读者立即获得练手的机会，通过练习巩

固运用所学知识。学完本书，在实际开发中，读者自然而然就会知道该做什么，如何去做以及为什么这样做。

最后，感谢图灵各位编辑老师的辛勤工作，尤其感谢李静老师的细心指导，本书及本人都获益良多。更要感谢的是我的clover和千寻，没有他们的宽容、理解与支持，本书译稿不可能完成。

虽然我已尽力传达原作本意并保证译稿的较高质量，但有时拼写错误、因版本升级而导致某些内容不再适用，甚至是囿于个人水平而犯错的情况在所难免。如果你发现了问题或有好的建议，请批评指正并不吝电邮提交至BNRAndroid@gmail.com或反馈至图灵社区。

2014年2月28日于上海



程序员最可信赖的求职帮手

目 录

第 1 章 Android 应用初体验.....1	2.10 挑战练习三：从按钮到图标按钮42
1.1 应用基础.....2	第 3 章 Activity 的生命周期45
1.2 创建 Android 项目.....2	3.1 日志跟踪理解 Activity 生命周期.....46
1.3 Eclipse 工作区导航.....5	3.1.1 输出日志信息46
1.4 用户界面设计6	3.1.2 使用 LogCat.....48
1.4.1 视图层级结构9	3.2 设备旋转与 Activity 生命周期.....52
1.4.2 组件属性10	3.3 设备旋转前保存数据56
1.4.3 创建字符串资源.....11	3.4 再探 Activity 生命周期57
1.4.4 预览界面布局12	3.5 深入学习：测试 onSaveInstanceState- (Bundle)方法59
1.5 从布局 XML 到视图对象13	3.6 深入学习：日志记录的级别与方法.....60
1.6 组件的实际应用15	第 4 章 Android 应用的调试.....62
1.6.1 类包组织导入16	4.1 DDMS 应用调试透视图63
1.6.2 引用组件16	4.2 异常与栈跟踪64
1.6.3 设置监听器17	4.2.1 诊断应用异常65
1.7 使用模拟器运行应用.....21	4.2.2 记录栈跟踪日志66
1.8 Android 编译过程22	4.2.3 设置断点.....68
第 2 章 Android 与 MVC 设计模式26	4.2.4 使用异常断点71
2.1 创建新类.....26	4.3 文件浏览器.....72
2.2 Android 与 MVC 设计模式.....30	4.4 Android 特有的调试工具73
2.3 更新视图层.....31	4.4.1 使用 Android Lint.....73
2.4 更新控制层.....33	4.4.2 R 类的问题.....74
2.5 在设备上运行应用37	第 5 章 第二个 activity.....75
2.5.1 连接设备37	5.1 创建第二个 activity.....76
2.5.2 配置设备用于应用开发38	5.1.1 创建新布局77
2.6 添加图标资源38	5.1.2 创建新的 activity 子类.....80
2.6.1 向项目中添加资源.....39	5.1.3 在 manifest 配置文件中声明 activity81
2.6.2 在 XML 文件中引用资源.....40	5.1.4 为 QuizActivity 添加 cheat 按钮82
2.7 关于挑战练习41	
2.8 挑战练习一：为 TextView 添加监听 器41	
2.9 挑战练习二：添加后退按钮42	

5.2 启动 activity	83	第 8 章 使用布局与组件创建用户 界面	128
5.3 activity 间的数据传递	85	8.1 升级 Crime 类	128
5.3.1 使用 intent extra	86	8.2 更新布局	129
5.3.2 从子 activity 获取返回结果	88	8.3 生成并使用组件	131
5.4 activity 的使用与管理	92	8.4 深入探讨 XML 布局属性	132
5.5 挑战练习	95	8.4.1 样式、主题及主题属性	132
第 6 章 Android SDK 版本与兼容	96	8.4.2 dp、sp 以及屏幕像素密度	133
6.1 Android SDK 版本	96	8.4.3 Android 开发设计原则	134
6.2 Android 编程与兼容性问题	97	8.4.4 布局参数	135
6.2.1 全新的系统版本—— Honeycomb	97	8.4.5 边距与内边距	135
6.2.2 SDK 最低版本	99	8.5 使用图形布局工具	136
6.2.3 SDK 目标版本	99	8.5.1 添加新组件	138
6.2.4 SDK 编译版本	99	8.5.2 属性视图中编辑组件属性	138
6.2.5 安全添加新版本 API 中的代 码	100	8.5.3 在框架视图中重新组织组件	139
6.3 使用 Android 开发者文档	103	8.5.4 更新子组件的布局参数	140
6.4 挑战练习：报告编译版本	105	8.5.5 android:layout_weight 属性的工作原理	141
第 7 章 UI fragment 与 fragment 管理器	106	8.5.6 图形布局工具使用总结	142
7.1 UI 设计的灵活性需求	107	8.5.7 组件 ID 与多种布局	142
7.2 fragment 的引入	107	8.6 挑战练习：日期格式化	143
7.3 着手开发 CriminalIntent	108	第 9 章 使用 ListFragment 显示列表	144
7.3.1 创建新项目	110	9.1 更新 CriminalIntent 应用的模型层	145
7.3.2 fragment 与支持库	112	9.2 创建 ListFragment	147
7.3.3 创建 Crime 类	113	9.3 使用抽象 activity 托管 fragment	149
7.4 托管 UI fragment	115	9.3.1 通用的 fragment 托管布局	149
7.4.1 fragment 的生命周期	115	9.3.2 抽象 activity 类	150
7.4.2 托管的两种方式	116	9.4 ListFragment、ListView 及 ArrayAdapter	154
7.4.3 定义容器视图	116	9.4.1 创建 ArrayAdapter<T> 类实例	157
7.5 创建 UI fragment	117	9.4.2 响应列表项的点击事件	159
7.5.1 定义 CrimeFragment 的布局	118	9.5 定制列表项	160
7.5.2 创建 CrimeFragment 类	119	9.5.1 创建列表项布局	160
7.6 添加 UI fragment 到 FragmentManager	122	9.5.2 创建 adapter 子类	162
7.6.1 fragment 事务	123	第 10 章 使用 fragment argument	165
7.6.2 FragmentManager 与 fragment 生命周期	125	10.1 从 fragment 中启动 activity	165
7.7 activity 使用 fragment 的理由	127	10.1.1 附加 extra 信息	166
7.8 深入学习：Honeycomb、ICS、 Jelly Bean 以及更高版本系统上 的应用开发	127	10.1.2 获取 extra 信息	167

10.1.3 使用 Crime 数据更新 CrimeFragment 视图	167	13.8 挑战练习：在 HelloMoon 应用中 播放视频	208
10.1.4 直接获取 extra 信息方式的 缺点	169	第 14 章 fragment 的保留	209
10.2 fragment argument	169	14.1 保留 fragment 实例	209
10.2.1 附加 argument 给 fragment	169	14.2 设备旋转与保留的 fragment	210
10.2.2 获取 argument	170	14.3 保留的 fragment：一切都完美了吗	212
10.3 重新加载显示列表项	171	14.4 设备旋转处理与 onSaveInstanceState(Bundle)方法	212
10.4 通过 fragment 获取返回结果	172	14.5 深入学习：fragment 引入前的设备 旋转问题	214
第 11 章 使用 ViewPager	174	第 15 章 应用本地化	215
11.1 创建 CrimePagerActivity	175	15.1 本地化资源	215
11.1.1 以代码的方式定义并产生 布局	176	15.2 配置修饰符	216
11.1.2 ViewPager 与 Pager- Adapter	177	15.2.1 可用资源优先级排定	217
11.1.3 整合配置并使用 CrimePagerActivity	178	15.2.2 多重配置修饰符	218
11.1.4 FragmentStatePager- Adapter 与 Fragment- PagerAdapter	180	15.2.3 寻找最匹配的资源	219
11.2 深入学习：ViewPager 的工作原理	182	15.3 更多资源使用原则及控制	220
第 12 章 对话框	184	15.3.1 资源命名	220
12.1 创建 DialogFragment	186	15.3.2 资源目录结构	220
12.1.1 显示 DialogFragment	187	15.4 测试备选资源	221
12.1.2 设置对话框的显示内容	188	第 16 章 操作栏	223
12.2 fragment 间的数据传递	190	16.1 选项菜单	223
12.2.1 传递数据给 DatePicker- Fragment	191	16.1.1 在 XML 文件中定义选项菜 单	225
12.2.2 返回数据给 CrimeFragment	193	16.1.2 创建选项菜单	227
12.3 挑战练习：更多对话框	198	16.1.3 响应菜单项选择	230
第 13 章 使用 MediaPlayer 播放音频	199	16.2 实现层级式导航	232
13.1 添加资源	200	16.2.1 启用应用图标的导航功能	232
13.2 定义 HelloMoonFragment 布局 文件	202	16.2.2 响应向上按钮	233
13.3 创建 HelloMoonFragment	203	16.3 可选菜单项	236
13.4 使用布局 fragment	204	16.3.1 创建可选菜单 XML 文件	236
13.5 音频播放	205	16.3.2 切换菜单项标题	237
13.6 挑战练习：暂停音频播放	208	16.3.3 “还有个问题”	238
13.7 深入学习：播放视频	208	16.4 挑战练习：用于列表的空视图	239
		第 17 章 存储与加载本地文件	241
		17.1 CriminalIntent 应用的数据存取	241
		17.1.1 保存 crime 数据到 JSON 文件	242

17.1.2 从文件中读取 crime 数据	246
17.2 挑战练习：使用外部存储	248
17.3 深入学习：Android 文件系统与 Java I/O	248
第 18 章 上下文菜单与上下文操作	
模式	250
18.1 定义上下文菜单资源	251
18.2 实施浮动上下文菜单	251
18.2.1 创建上下文菜单	251
18.2.2 为上下文菜单登记视图	252
18.2.3 响应菜单项选择	253
18.3 实施上下文操作模式	254
18.3.1 实现列表视图的多选操作	255
18.3.2 列表视图中的操作模式回调 方法	256
18.3.3 改变已激活视图的显示背景	258
18.3.4 实现其他视图的上下文操作 模式	259
18.4 兼容性问题：回退还是复制	260
18.5 挑战练习：在 CrimeFragment 视图 中删除 crime 记录	261
18.6 深入学习：ActionBarSherlock	261
18.7 挑战练习：使用 ActionBarSherlock	263
18.7.1 CriminalIntent 应用中 ABS 的基本整合	264
18.7.2 ABS 的深度整合	264
18.7.3 ABS 的完全整合	265
第 19 章 相机 I：取景器	266
19.1 创建 Fragment 布局	267
19.2 创建 CrimeCameraFragment	269
19.3 创建 CrimeCameraActivity	269
19.4 使用相机 API	271
19.4.1 打开并释放相机	271
19.4.2 SurfaceView、Surface- Holder 与 Surface	272
19.4.3 确定预览界面大小	276
19.4.4 启动 CrimeCamera- Activity	277
19.5 深入学习：以命令行的方式运行 activity	281
第 20 章 相机 II：拍摄并处理照片	283
20.1 拍摄照片	283
20.1.1 实现相机回调方法	285
20.1.2 设置图片尺寸大小	288
20.2 返回数据给 CrimeFragment	288
20.2.1 以接收返回值的方式启动 CrimeCameraActivity	289
20.2.2 在 CrimeCameraFragment 中设置返回值	290
20.2.3 在 CrimeFragment 中获取 照片文件名	290
20.3 更新模型层	291
20.3.1 新增 Photo 类	292
20.3.2 为 Crime 添加 photo 属性	293
20.3.3 设置 photo 属性	293
20.4 更新 CrimeFragment 的视图	294
20.4.1 添加 ImageView 组件	295
20.4.2 图像处理	296
20.5 在 DialogFragment 中显示大图片	300
20.6 挑战练习：Crime 照片的显示方向	303
20.7 挑战练习：删除照片	303
20.8 深入学习：Android 代码的废弃 处理	303
第 21 章 隐式 intent	305
21.1 添加按钮组件	306
21.2 添加嫌疑人信息至模型层	307
21.3 使用格式化字符串	308
21.4 使用隐式 intent	309
21.4.1 典型隐式 intent 的组成	310
21.4.2 发送陋习报告	311
21.4.3 获取联系人信息	313
21.4.4 检查可以响应的 activity	316
21.5 挑战练习：又一个隐式 intent	317
第 22 章 Master-Detail 用户界面	318
22.1 增加布局灵活性	319
22.1.1 修改 SingleFragment- Activity	320
22.1.2 创建包含两个 fragment 容器的布局	320

22.1.3 使用别名资源.....	322	第 27 章 Looper、Handler 与 HandlerThread.....	387
22.2 Activity: fragment 的托管者	323	27.1 设置 GridView 以显示图片	387
22.3 深入学习: 设备屏幕尺寸的确定	331	27.2 批量下载缩略图	390
第 23 章 深入学习 intent 和任务	333	27.3 与主线程通信	390
23.1 创建 NerdLauncher 项目	333	27.4 创建并启动后台线程	391
23.2 解析隐式 intent	334	27.5 Message 与 message Handler	393
23.3 在运行时创建显式 intent	337	27.5.1 消息的剖析	393
23.4 任务与后退栈	338	27.5.2 Handler 的剖析	393
23.5 使用 NerdLauncher 应用作为设备 主屏幕	341	27.5.3 使用 handler	395
23.6 挑战练习: 应用图标与任务重排	341	27.5.4 传递 handler	397
23.7 进程与任务	341	27.6 深入学习: AsyncTask 与 Thread	401
第 24 章 样式与 include 标签的使用	343	27.7 挑战练习: 预加载以及缓存	401
24.1 创建 RemoteControl 项目	344	第 28 章 搜索	402
24.1.1 编码实现 RemoteControl- Activity	344	28.1 搜索 Flickr 网站	402
24.1.2 创建 RemoteControl- Fragment	345	28.2 搜索对话框	404
24.2 使用样式消除重复代码	348	28.2.1 创建搜索界面	404
24.3 完善布局定义	350	28.2.2 可搜索的 activity	406
24.4 深入学习: 使用 include 与 merge 标签	353	28.2.3 物理搜索键	408
24.5 挑战练习: 样式的继承	354	28.2.4 搜索的工作原理	409
第 25 章 XML Drawable 与 9-Patches	355	28.2.5 启动模式与新的 intent	410
25.1 XML drawable	356	28.2.6 使用 shared preferences 实现轻量级数据存储	412
25.2 state list drawable	358	28.3 在 Android 3.0 以后版本的设备上 使用 SearchView	414
25.3 layer list 与 inset drawable	360	28.4 挑战练习	416
25.4 使用 9-patch 图像	362	第 29 章 后台服务	417
第 26 章 HTTP 与后台任务	368	29.1 创建 IntentService	417
26.1 创建 PhotoGallery 应用	369	29.2 服务的作用	419
26.2 网络连接基本	372	29.3 查找最新返回结果	421
26.3 使用 AsyncTask 在后台线程上运行 代码	373	29.4 使用 AlarmManager 延迟运行服务	422
26.4 线程与主线程	375	29.4.1 PendingIntent	424
26.5 获取 Flickr XML 数据	377	29.4.2 使用 PendingIntent 管理 定时器	424
26.6 从 AsyncTask 回到主线程	382	29.5 控制定时器	425
26.7 深入学习: 再探 AsyncTask	385	29.6 通知信息	428
26.8 挑战练习: 分页	386	29.7 深入学习: 服务细节内容	429
		29.7.1 服务的能与不能	430
		29.7.2 服务的生命周期	430

29.7.3	non-sticky 服务	430	33.1.2	创建 RunFragment	470
29.7.4	sticky 服务	431	33.2	地理位置与 LocationManager	472
29.7.5	绑定服务	431	33.3	接收定位数据更新 broadcast	474
第 30 章	broadcast Intent	433	33.4	使用定位数据刷新 UI 显示	475
30.1	随设备重启而重启的定时器	433	33.5	快速定位：最近一次地理位置	479
30.1.1	配置文件中的 broadcast receiver	434	33.6	在物理和虚拟设备上测试地理位置定位	480
30.1.2	如何使用 receiver	435	第 34 章	使用 SQLite 本地数据库	482
30.2	过滤前台通知消息	436	34.1	在数据库中存储旅程和地理位置信息	482
30.2.1	发送 broadcast intent	437	34.2	查询数据库中的旅程列表	488
30.2.2	动态 broadcast receiver	437	34.3	使用 CursorAdapter 显示旅程列表	490
30.2.3	使用私有权限	440	34.4	创建新旅程	493
30.2.4	使用 ordered broadcast 接收结果	442	34.5	管理现有旅程	494
30.3	receiver 与长时运行任务	446	34.6	挑战练习：识别当前跟踪的旅程	500
第 31 章	网页浏览	447	第 35 章	使用 Loader 加载异步数据	501
31.1	最后一段 Flickr 数据	447	35.1	Loader 与 LoaderManager	501
31.2	简单方式：使用隐式 intent	448	35.2	在 RunTracker 应用中使用 Loader	502
31.3	较难方式：使用 WebView	449	35.3	加载旅程列表	503
31.3.1	使用 WebChromeClient 优化 WebView 的显示	453	35.4	加载单个旅程	506
31.3.2	处理 WebView 的设备旋转问题	455	35.5	加载旅程的最近一次地理位置	509
31.4	深入学习：注入 JavaScript 对象	456	第 36 章	使用地图	511
第 32 章	定制视图与触摸事件	457	36.1	添加 Maps API 给 RunTracker 应用	511
32.1	创建 DragAndDraw 项目	457	36.1.1	使用物理设备测试地图	511
32.1.1	创建 DragAndDraw-Activity	458	36.1.2	安装使用 Google Play services SDK	511
32.1.2	创建 DragAndDraw-Fragment	459	36.1.3	获取 Google Maps API key	512
32.2	创建定制视图	460	36.1.4	更新 RunTracker 应用的 manifest 配置文件	512
32.3	处理触摸事件	462	36.2	在地图上显示用户的地理位置	513
32.4	onDraw(...) 方法内的图形绘制	465	36.3	显示旅程路线	516
32.5	挑战练习：设备旋转问题	467	36.4	为旅程添加开始和结束地图标注	520
第 33 章	跟踪设备的地理位置	468	36.5	挑战练习：实时数据更新	521
33.1	启动 RunTracker 项目	468	第 37 章	编后语	522
33.1.1	创建 RunActivity	469	37.1	终极挑战	522
			37.2	关于我们	523
			37.3	致谢	523