

O'REILLY®



涵盖Android Nougat 7.0
第2版



Android 应用开发实战

Android Cookbook:
Problems and Solutions for
Android Developers



机械工业出版社
China Machine Press

Ian F. Darwin 著
胡训强 夏红梅 张文娟 译

原书第2版

Android 应用开发实战

Ian F. Darwin 著

潮州張寧紅梅 张文娟 译

藏书章

Beijing • Boston • Farnham • Sebastopol • Tokyo

O'REILLY®

O'Reilly Media, Inc. 授权机械工业出版社出版

机械工业出版社

图书在版编目 (CIP) 数据

Android 应用开发实战 (原书第 2 版) / (美) 伊恩 F. 达尔文 (Ian F. Darwin) 著; 胡训强, 夏红梅, 张文娟译. —北京: 机械工业出版社, 2018.1

(O'Reilly 精品图书系列)

书名原文: Android Cookbook: Problems and Solutions for Android Developers,
Second Edition

ISBN 978-7-111-58996-9

I. A… II. ①伊… ②胡… ③夏… ④张… III. 移动终端—应用程序—程序设计 IV. TN929.53

中国版本图书馆 CIP 数据核字 (2018) 第 013975 号

北京市版权局著作权合同登记

图字: 01-2017-6567 号

© 2017 O'Reilly Media, Inc.

Simplified Chinese Edition, jointly published by O'Reilly Media, Inc. and China Machine Press, 2018.
Authorized translation of the English edition, 2017 O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由 O'Reilly Media, Inc. 出版 2017。

简体中文版由机械工业出版社出版 2018。英文原版的翻译得到 O'Reilly Media, Inc. 的授权。此简体中文版的出版和销售得到出版权和销售权的所有者——O'Reilly Media, Inc. 的许可。

版权所有, 未得书面许可, 本书的任何部分和全部不得以任何形式重制。

封底无防伪标均为盗版

本书法律顾问

北京大成律师事务所 韩光 / 邹晓东

书 名 / Android 应用开发实战 (原书第 2 版)

书 号 / ISBN 978-7-111-58996-9

责任编辑 / 张锡鹏

封面设计 / Karen Montgomery, 张健

出版发行 / 机械工业出版社

地 址 / 北京市西城区百万庄大街 22 号 (邮政编码 100037)

印 刷 / 三河市宏图印务有限公司

开 本 / 178 毫米 × 233 毫米 16 开本 43 印张

版 次 / 2018 年 3 月第 1 版 2018 年 3 月第 1 次印刷

定 价 / 139.00 元 (册)

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010)88379426; 88361066

购书热线: (010)68326294; 88379649; 68995259

投稿热线: (010)88379604

读者信箱: hzit@hzbook.com

O'Reilly Media, Inc. 介绍

O'Reilly Media 通过图书、杂志、在线服务、调查研究和会议等方式传播创新知识。自 1978 年开始，O'Reilly 一直都是前沿发展的见证者和推动者。超级极客们正在开创着未来，而我们关注真正重要的技术趋势——通过放大那些“细微的信号”来刺激社会对新科技的应用。作为技术社区中活跃的参与者，O'Reilly 的发展充满了对创新的倡导、创造和发扬光大。

O'Reilly 为软件开发人员带来革命性的“动物书”；创建第一个商业网站（GNN）；组织了影响深远的开放源代码峰会，以至于开源软件运动以此命名；创立了 Make 杂志，从而成为 DIY 革命的主要先锋；公司一如既往地通过多种形式缔结信息与人的纽带。O'Reilly 的会议和峰会集聚了众多超级极客和高瞻远瞩的商业领袖，共同描绘出开创新产业的革命性思想。作为技术人士获取信息的选择，O'Reilly 现在还将先锋专家的知识传递给普通的计算机用户。无论是通过书籍出版，在线服务或者面授课程，每一项 O'Reilly 的产品都反映了公司不可动摇的理念——信息是激发创新的力量。

业界评论

“O'Reilly Radar 博客有口皆碑。”

——Wired

“O'Reilly 凭借一系列（真希望当初我也想到了）非凡想法建立了数百万美元的业务。”

——Business 2.0

“O'Reilly Conference 是聚集关键思想领袖的绝对典范。”

——CRN

“一本 O'Reilly 的书就代表一个有用、有前途、需要学习的主题。”

——Irish Times

“Tim 是位特立独行的商人，他不光放眼于最长远、最广阔的视野并且切实地按照 Yogi Berra 的建议去做了：‘如果你在路上遇到岔路口，走小路（岔路）。’回顾过去 Tim 似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

——Linux Journal

译者序

近十年来，智能移动设备的发展可以说是一日千里，除了硬件设备之外，手机 OS 显然是其中的一个重要引擎，而 Android 无疑是手机 OS 领域的两大王者之一（另一个是 iOS）。尽管几年前人们对 Android 手机的卡顿现象还时有吐槽，但是随着 Android 不断的更新换代以及硬件厂商对自己产品的优化，两者在性能方面的差距逐渐缩小，某些旗舰版 Android 手机的表现已与 iPhone 相差无几。根据市场分析公司凯度移动发布的数据，2016 年 Android 在中国的市场占有率达到了 77.7%，同比增长 6%；在欧洲五大市场这个数字为 74.5%，同比增长 7.1%；即使在 iOS 的老家美国，Android 也拿下了 65.5%，同比增长 7.3%，可见，Android 是移动互联网这个风头正劲的“风口”的一台“强力鼓风机”。而 Android 与生俱来的移动开放开发平台的特性，也使得作为开发人员的我们更愿意利用它创造出一些有趣的新玩意儿出来。为此，本书为广大希望全面掌握这一“风口”上技术的读者奉上了一场饕餮盛宴。

本书虽厚，但有别于市面上一些满页贴代码的所谓“典藏”“大全”之类的 Android 书籍。就译者本人在翻译过程中的感受而言，本书有三大特点：内容全面、逻辑顺畅、重点突出。所谓内容全面，是指本书涵盖了 Android 应用程序开发的方方面面，从 GUI、图形图像、多媒体、数据持久化到各种通信方式、常用传感器等内容无所不包，甚至还囊括了应用程序测试、国际化和面向非 Java 程序员的编程方式等内容，自然应用程序开发环境的搭建和程序的发布也不例外。可以说只要你从事 Android 开发，无论打算使用什么 IDE、准备开发什么样的项目，都能从本书中找到适合自己的内容；所谓逻辑顺畅，指的是本书的编排别具匠心，几乎每一节都是先提出问题，即该节内容所面向的程序应用背景，然后提出总体解决思路，最后再按步实施、解决问题。为什么、怎么办、如何解决可以说是贯穿全书的逻辑思路，清晰明了，即使你由于时间关系不能精读全书，也总会在实际开发工作中遭遇“痛点”之时，回忆起本书中相同或类似的问题场景，快速找到解决方案；重点突出则是由于本书作者是来自于 Android 社区的优秀一线开发人员，对 Android 程序员在日常开发中容易混淆、迷茫、犯错、走弯路的种种问题了然于胸，并对这类问题的解决方案给予了重点“关照”，而对于

大家都耳熟能详的步骤，不会一而再、再而三地喋喋不休。本书的这三个特点，使其既能成为 Android 开发新手的入门读物，又能作为富有经验的开发人员的案头工具书，可以说值得每位 Android 程序员品读。

全书共有 21 章，由胡训强、夏红梅和张文娟共同翻译，其中胡训强翻译了第 1、2 章以及第 5 ~ 18 章，夏红梅翻译了第 3 章和第 4 章，张文娟翻译了第 19 ~ 21 章。囿于译者的技术和语言水平，书中难免会出现不准确甚至错误之处，如果你发现了这些问题，恳请通过邮箱 10185014@qq.com 告知我们以便改进，在此向你表示感谢！

最后，感谢所有为本书顺利付梓而付出艰辛劳动的人们！

译 者

2017 年 9 月于广州

目录

前言	1
第 1 章 入门	9
1.1 理解 Android 应用程序体系结构	9
1.2 理解 Android Activity 的生命周期	11
1.3 了解 Android 发行版本	13
1.4 学习 Java 语言	15
1.5 从命令行创建 “Hello World” 应用程序	16
1.6 用 Apache Maven 创建 “Hello World” 应用程序	20
1.7 为 Android 开发选择集成开发环境 (IDE)	22
1.8 安装 Android Studio	26
1.9 安装 Platform Editions 并保持 SDK 更新	28
1.10 用 Android Studio 创建 “Hello World” 应用程序	31
1.11 将 Eclipse ADT 项目转换成 Android Studio 项目	37
1.12 从 Eclipse 向 Android Studio 转换时保留历史记录	40
1.13 用 Eclipse 和 Android Studio 共同构建 Android 应用程序	42
1.14 安装带有 AndMore 的 Eclipse (代替 ADT)	44
1.15 用 Eclipse 创建 “Hello World” 应用程序	51
1.16 在 Eclipse 中安装 Eclipse Marketplace Client	56
1.17 将项目从 Eclipse ADT 升级到 Eclipse AndMore	59
1.18 用命令行 ADB 控制模拟器 / 设备	62
1.19 共享其他 Eclipse 项目中的 Java 类	64
1.20 引用库实现外围功能	66

1.21 通过兼容库在老设备上使用新功能	70
1.22 应用 SDK 示例帮助避免让人头疼的问题	72
1.23 从模拟器 /Android 设备上截图 / 截取视频	73
1.24 程序：一个简单的定时器示例	78
1.25 程序：Tipster, Android 操作系统上的小费计算器	82
第 2 章 设计成功的应用程序	98
2.1 异常处理	101
2.2 运行时请求 Android 访问权限	104
2.3 以“单例”模式访问 Android 的 Application 对象	106
2.4 在用户旋转设备时保持数据	109
2.5 监控 Android 设备上的电池电量	111
2.6 在 Android 中创建闪屏	112
2.7 设计面向会议 / 野营 / 黑客马拉松 / 公共机构的应用程序	116
2.8 在 Android 应用程序中使用 Google 的分析工具	118
2.9 进行首次运行的偏好设置	120
2.10 格式化数字	122
2.11 用正确的复数形式进行格式化	126
2.12 格式化时间和日期进行显示	128
2.13 用 Java 8 的 java.time API 简化日期 / 时间计算	131
2.14 用键盘监听器控制输入	133
2.15 备份 Android 应用程序数据	136
2.16 用提示 (Hint) 代替工具小贴士	142
第 3 章 应用程序测试	144
3.1 针对测试设置 Android 虚拟设备	145
3.2 在大量设备上进行云测试	151
3.3 用 Eclipse 和 JUnit 进行测试	152
3.4 用 Android Studio 和 JUnit 进行测试	154
3.5 用 Robolectric 和 JUnit 4 进行测试	159
3.6 用 ATSL、Espresso 和 JUnit 4 进行测试	161
3.7 排查应用程序崩溃原因	165
3.8 用 Log.d() 和 LogCat 进行调试	168

3.9 用 Crash Reporting 自动获取崩溃报告	169
3.10 针对现场错误或情况使用本地运行时应用程序日志	172
3.11 为测试重现 Activity 生命周期场景	176
3.12 利用严苛模式让应用程序保持简洁	180
3.13 用 Android Lint 进行静态代码测试	181
3.14 用 Monkey 程序进行动态测试	183
3.15 在 AVD 之间发送文本消息和通话	185
第 4 章 进程间 / 进程内通信	187
4.1 用 Intent 打开网页、电话号码本或其他应用程序	187
4.2 从视图中用电子邮件发送文本	189
4.3 发送带有附件的电子邮件	192
4.4 用 Intent.putExtra() 发送字符串值	194
4.5 从子 Activity 向主 Activity 发回数据	195
4.6 当其他应用程序在屏幕上显示的同时让后台服务保持运行	198
4.7 发送 / 接收广播消息	200
4.8 在设备重启后启动服务	201
4.9 使用线程创建高可响应的应用程序	202
4.10 使用 AsyncTask 进行后台处理	203
4.11 利用 Activity 的线程队列和 Handler 在线程间发送消息	210
4.12 结合 Android 和 HTML/JavaScript 创建一个万年历应用程序	211
第 5 章 图形	218
5.1 使用自定义字体	218
5.2 用 OpenGL ES 绘制旋转正方体	221
5.3 为 OpenGL 旋转正方体添加控制	224
5.4 手绘光滑曲线	227
5.5 用 Intent 拍照	232
5.6 用 android.media.Camera 照相	234
5.7 用 Google 的条形码扫描器 ZXing 扫描条形码和二维码	238
5.8 用 AndroidPlot 显示图表和图形	240
5.9 使用 Inkscape 借助 OpenClipArt.org 创建 Android 启动器图标	244
5.10 使用 Paint.NET 借助 OpenClipArt.org 创建 Android 启动器图标	248

5.11 使用九宫格文件	254
5.12 使用 Android RGraph 创建 HTML5 图表	258
5.13 添加一个简单的光栅动画	262
5.14 用拇指和食指进行缩放	265
第 6 章 图形用户界面	268
6.1 理解和遵循用户界面准则	269
6.2 用原质化设计赋予应用程序良好外观	270
6.3 选择布局管理器 (ViewGroup) 排列组件	274
6.4 通过解耦视图与模型处理配置变化	275
6.5 处理动作栏	277
6.6 向动作栏添加共享动作	281
6.7 用 Fragment API 构建现代 UI	285
6.8 创建按钮及其单击事件监听器	289
6.9 用图像按钮增强 UI 设计	290
6.10 使用 FloatingActionButton	291
6.11 以多种不同方式编写事件监听器	294
6.12 使用复选框和单选按钮	298
6.13 使用卡片窗口部件	302
6.14 通过 Spinner 类提供下拉选择器	304
6.15 处理长按 / 长点事件	307
6.16 用 TextView 和 EditText 显示文本域	308
6.17 用属性和 TextWatcher 接口限制 EditText 的值	309
6.18 实现 AutoCompleteTextView	311
6.19 使用 SQLite 数据库查询填充 AutoCompleteTextView	313
6.20 将文本域转换为密码域	315
6.21 将软键盘上的回车键改为“下一个”	315
6.22 在 Activity 中处理键盘按键事件	318
6.23 让他们看星星：使用 RatingBar	319
6.24 让视图晃动	323
6.25 提供触觉反馈	325
6.26 在 TabView 中的各个 Activity 之间切换	328
6.27 创建在两个 Activity 之间显示的载入界面	329

6.28 为布局添加带有圆角的边框.....331

6.29 在 Android 中检测手势.....333

6.30 创建一个简单的应用程序窗口小部件.....340

第 7 章 GUI 中的警告：菜单、对话框、Toast、Snackbar 和通知.....344

7.1 用 Toast 和 Snackbar 向用户发出警告.....344

7.2 自定义 Toast 的外观.....346

7.3 创建和显示菜单.....348

7.4 处理菜单项的选择.....349

7.5 创建子菜单.....350

7.6 创建弹出式 / 警告对话框.....352

7.7 创建时间选择器窗口部件.....353

7.8 针对选择创建 iPhone 风格的滚轮选择器.....356

7.9 创建选项卡式对话框.....358

7.10 创建进度对话框.....361

7.11 创建带有按钮、图像和文本的自定义对话框.....362

7.12 创建可重用的“关于对话框”类.....364

7.13 在状态栏中创建通知.....368

第 8 章 其他 GUI 组件：列表和视图.....373

8.1 用 RecyclerView 构建基于列表的应用程序.....373

8.2 用 ListView 构建基于列表的应用程序.....376

8.3 为 ListView 创建“无数据”视图.....381

8.4 创建带有图像和文本的高级 ListView.....382

8.5 在 ListView 中使用分组标题.....386

8.6 让 ListView 始终处于用户的关注点上.....390

8.7 编写自定义列表适配器.....391

8.8 利用 SearchView 在 ListView 中查找数据.....394

8.9 处理方向变化：从 ListView 数据值到全局模式中的图表绘制.....396

第 9 章 多媒体.....402

9.1 播放 YouTube 视频.....402

9.2 利用 MediaRecorder 录制视频	403
9.3 使用 Android 的人脸检测功能	406
9.4 播放音频文件	409
9.5 在没有交互的情况下播放音频	411
9.6 使用语音文本转换	412
9.7 利用文本语音转换让设备发声	414
第 10 章 数据持久化	416
10.1 读写内存存储器和外存储器中的文件	416
10.2 获取文件和文件夹信息	421
10.3 读取应用程序自带而非文件系统中的文件	426
10.4 获取 SD 卡的存储空间信息	427
10.5 提供偏好设置 Activity	428
10.6 检查共享偏好设置的一致性	433
10.7 在 Android 应用程序中使用 SQLite 数据库	435
10.8 在 SQLite 数据库上执行高级文本检索	438
10.9 处理 SQLite 中的日期	443
10.10 将非 SQL 数据对外暴露为 SQL 游标	445
10.11 用 CursorLoader 显示数据	448
10.12 利用 JSONObject 解析 JSON	451
10.13 利用 DOM API 解析 XML 文档	453
10.14 利用 ContentProvider 存储和获取数据	455
10.15 编写一个 ContentProvider	456
10.16 通过联系人内容提供者添加联系人	460
10.17 利用 ContentProvider 读取联系人数据	462
10.18 实现拖放	464
10.19 通过 FileProvider 共享文件	468
10.20 利用 SyncAdapter 把 SQLite 中的数据备份到云端	473
10.21 利用 Google Firebase 将数据存放到云端	482
第 11 章 电话应用程序	487
11.1 来电处理	487
11.2 处理去电呼叫	490

11.3 拨打电话	494
11.4 发送单段式或多段式短信	495
11.5 接收短信	498
11.6 利用模拟器控制台向模拟器发送短信	499
11.7 利用 Android 的 TelephonyManager 获取设备信息	500
第 12 章 网络应用程序	511
12.1 借助 URLConnection 利用 RESTful Web 服务	512
12.2 借助 Volley 利用 RESTful Web 服务	515
12.3 利用 Google Cloud Messaging 的“推送消息”向应用程序发出 通知消息	517
12.4 利用正则表达式从非结构化文本中提取信息	525
12.5 利用 ROME 解析 RSS/Atom 订阅内容	526
12.6 利用 MD5 加密纯文本	531
12.7 将文本转换为超链接	531
12.8 利用 WebView 访问网页	532
12.9 自定义 WebView	533
12.10 编写进程间通信服务	534
第 13 章 游戏和动画	539
13.1 利用 flixel-gdx 制作 Android 游戏	540
13.2 利用 AndEngine 制作 Android 游戏	544
13.3 处理与时间相关的键盘输入	549
第 14 章 社交网络	551
14.1 利用 OAuth2 对用户进行身份认证	551
14.2 利用 HTTP 集成社交网络	555
14.3 利用 HTML 或 JSON 加载用户的 Twitter 时间线	557
第 15 章 定位和地图应用程序	559
15.1 获取位置信息	559
15.2 在应用程序内获取 GPS 信息	561
15.3 在设备上模拟 GPS 坐标	563

15.4 使用地理编码和逆向地理编码	565
15.5 为使用 Google Maps API V2 进行开发做好准备	566
15.6 使用 Google Maps API V2.....	571
15.7 利用 OpenStreetMap 显示地图数据	576
15.8 利用 OpenStreetMap 地图创建覆盖图层	578
15.9 在 OpenStreetMap 地图上使用比例尺	581
15.10 在 OpenStreetMap 覆盖图层上处理触摸事件	582
15.11 利用 OpenStreetMap 获取定位更新	584
第 16 章 加速计	588
16.1 检查传感器是否存在	588
16.2 利用加速计检测晃动	589
16.3 检测设备正面朝上还是朝下.....	593
16.4 读取温度传感器	594
第 17 章 蓝牙	595
17.1 启用蓝牙并让设备可被发现.....	595
17.2 连接启用了蓝牙的设备	597
17.3 接受来自蓝牙设备的连接	599
17.4 实现发现蓝牙设备的功能	600
第 18 章 系统和设备控制.....	602
18.1 获取手机的网络 / 连接信息	602
18.2 从应用程序清单文件中获取信息.....	603
18.3 将来电通知改为静音、振动或普通铃声	604
18.4 复制文本并从剪切板中获取文本.....	606
18.5 使用基于 LED 的通知	606
18.6 让设备振动	607
18.7 确定指定的应用程序是否正在运行	608
第 19 章 世上并非只有 Java：其他编程语言和框架	609
19.1 了解跨平台解决方案	610
19.2 在应用程序中执行 Shell 命令	611

19.3 利用 JNI 在 NDK 上运行原生 C/C++ 代码	613
19.4 Android 脚本编程层 SL4A 入门	618
19.5 在 SL4A 中创建警告消息	620
19.6 利用 SL4A 获取 Google 文档并在 ListView 中显示	624
19.7 在二维码中共享 SL4A 脚本	626
19.8 通过 JavaScript 在 WebView 中使用原生的手机功能	628
19.9 利用 Xamarin 构建跨平台应用程序	630
19.10 利用 PhoneGap/Cordova 构建跨平台应用程序	634
第 20 章 世上并非只有英语：字符串和国际化	639
20.1 国际化应用程序中的文本	640
20.2 查找和翻译字符串	643
20.3 处理 strings.xml 中的细微差别	644
第 21 章 打包、部署及发布 / 出售应用程序	650
21.1 创建签名证书并用其对应用程序签名	650
21.2 通过 Google Play 应用商店发布应用程序	653
21.3 通过其他应用商店发布应用程序	655
21.4 利用 AdMob 让你的应用程序赚钱	656
21.5 利用 ProGuard 进行混淆和优化	662
21.6 让应用程序驻留在你自己的服务器上	664
21.7 创建“自行更新”的应用程序	666
21.8 为 Google Play 应用商店中其他已发布的应用程序提供链接	668

前言

Android 是应用于手机和移动运算的“开源革命”，至少是部分革命。为了提供开源手机，人们已经进行了大量努力，其中大部分成果在很大程度上都已经名存实亡，包括 Openmoko Neo FreeRunner、嵌入式 QT、Moblin、LiMo、Debian Mobile、Maemo、Firefox OS、Ubuntu Mobile 以及开源的 Symbian OS 和现在销声匿迹的 HP WebOS。我们也不要忘了已被证实是闭源的坚定分子——Apple 公司的 iOS 以及两个次要角色（从市场占有率而言）——Microsoft 公司的 Windows Phone 以及现在已被抛弃的 BlackBerry OS 10。

在所有被提供的产品当中，有两个脱颖而出成为主角，其中 Android 必定占有一席之地！由于其开源许可，因此 Android 在全世界众多经济型手机上都得到了使用，据估计 Android 在世界智能手机市场中占有的份额高达 90%。写作本书是为了帮助 Android 开发人群共享有助于开发出更好的应用程序的知识，那些在本书中贡献出知识的人使得 Android 开发对于后来者更加简单。

关于 Android

Android 是一个移动电话技术平台，可以为手机、平板电脑以及其他手持和移动设备（甚至上网本）提供 Linux 操作系统的强大功能和便携性、标准高级语言和 API 的可靠性、可移植性以及实用应用程序的巨大生态系统。Android 应用程序大多数都是（利用诸如 Eclipse 和 Android Studio 之类的工具）用 Java 语言编写的，依赖于 Android API 进行编译并被转换成面向 Android 虚拟机的字节码。

因此，从操作系统家族的角度而言，Android 和其他基于 Linux 的手机项目息息相关。从编程语言角度而言，Android 还和 BlackBerry 的老旧的 Java ME 手机以及 Java 及领域更广的 Java 企业版应用程序有关。不要说现在所有的 BlackBerry 设备都能运行 Android

应用程序，实际上在其把剩余的智能手机业务外包出去之前，BlackBerry 的最后一批设备只能运行 Android。

现在大家公认 Android 占有世界智能手机市场几乎 3/4 的份额，但在平板电脑市场上它还没有取代 Apple 公司的 iPad。销售数据随时都在变化，但是有一点非常明确：Android 现在是并且将继续保持为移动领域的主角之一。

Android 还可用于一些专门的平台。Android Wear 将 Android 的编程模型带入了智能手表以及诸如健康跟踪器之类的可穿戴设备。Android Auto 则被设计用于控制汽车里的娱乐单元。Android TV 则在智能电视和不那么智能的电视的控制器上运行。最后，Android Things 则被设计用于嵌入式市场，即现在称为的“物联网”（Internet of Things, IoT）。这中间的每一个平台都引人入胜，但为了让本书的篇幅保持在合理范围内，我们主要关注“常规的 Android”，即面向智能手机和平板电脑应用程序的 Android。

本书读者对象

本书关注利用 Java 构建 Android 应用程序，Java 是 Android 应用程序的原生语言。将 Web 应用程序打包成移动应用程序当然也是可能的（参见 19.10 节），但是想利用这种方式获得有关当前 Android 所有功能特性的非常重要的 100% 正确的用户体验却很困难。

所以本书用到的是 Java。我们假定你已经了解了 Java 语言的基础知识，如果没有，请参阅 1.4 节。我们还假定你已经了解了 Java 标准版 API（因为它构成了 Android 运行时库的基础）以及 Android 的基础知识。虽然你不必非常精通 Activity、Intent、Service 和 ContentProvider 等术语，但至少应该熟悉它们。如果不是这样，我们已经为你准备好了补救措施：请参阅 1.2 节。

本书和 Android SDK 自带示例的不同之处在于，书中尝试更多地关注特定技术的工作原理，而不是给你提供一个完整可运行的例子，这样的例子既被简化了（使用非常简单的数据），同时又因为添加进若干与眼前的问题没有关系的“巧妙的”特性而复杂化了。

本书主要内容

第 1 章带你全面浏览一遍设置 Android 开发环境以及构建几个众所周知的“Hello, World”类型的简单应用程序的所有步骤，“Hello, World”类型程序的概念是由 Brian Kernighan 率先提出的。

第 2 章介绍了移动计算中的一些差异之处，这些差异会使从桌面和企业软件环境转型而来的开发人员不太适应。本章还谈论了移动设计（特别是 Android 设计）是如何与那些