

TURING

图灵程序设计丛书

Big Nerd  
Ranch

- Amazon移动开发类榜首畅销书
- Android开发入门与进阶不二之选
- 简单易懂的示例+实战项目，涵盖Android开发方方面面

# Android 编程权威指南

(第3版)

[美] Bill Phillips Chris Stewart Kristin Marsicano 著  
王明发 译

Android Programming

The Big Nerd Ranch Guide, Third Edition



中国工信出版集团



人民邮电出版社  
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

# Android 编程权威指南 (第3版)

[美] Bill Phillips, Chris Stewart, Kristin Marsicano 著

王明发 译

**Android Programming**

The Big Nerd Ranch Guide, Third Edition

人民邮电出版社

北京

## 图书在版编目 (CIP) 数据

Android编程权威指南 / (美) 比尔·菲利普斯  
(Bill Phillips), (美) 克里斯·斯图尔特  
(Chris Stewart), (美) 克莉丝汀·马西卡诺  
(Kristin Marsicano) 著; 王明发 译. — 3版. — 北  
京: 人民邮电出版社, 2017.6  
(图灵程序设计丛书)  
ISBN 978-7-115-45759-2

I. ①A… II. ①比… ②克… ③克… ④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(2017)第114263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。本书主要以其 Android 训练营教学课程为基础,融合了几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 36 章,详细介绍了 8 个 Android 应用的开发过程。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,获得宝贵的开发经验。

第 3 版较之前版本增加了对数据绑定等新工具的介绍,同时新增了针对单元测试、辅助功能和 MVVM 架构等主题的章节。如果你熟悉 Java 语言,或者了解面向对象编程,那就立刻开始 Android 编程之旅吧!

- 
- ◆ 著 [美] Bill Phillips Chris Stewart Kristin Marsicano  
译 王明发  
责任编辑 谢婷婷  
责任印制 彭志环
- ◆ 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>  
三河市海波印务有限公司印刷
- ◆ 开本: 800×1000 1/16  
印张: 37.25 彩插: 1  
字数: 884千字 2017年6月第3版  
印数: 38 501 - 42 000册 2017年6月河北第1次印刷  
著作权合同登记号 图字: 01-2017-3139号
- 

定价: 129.00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: 京东工商广登字 20170147 号

# 版 权 声 明

Authorized translation from the English language edition, entitled *Android Programming: The Big Nerd Ranch Guide, Third Edition* by Bill Phillips, Chris Stewart, Kristin Marsicano, published by The Big Nerd Ranch, Inc., publishing as Big Nerd Ranch Guides, Copyright © 2017 Big Nerd Ranch, LLC.

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 Pearson Education, Inc.

Simplified Chinese-language edition Copyright © 2017 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由Pearson Education, Inc.授权人民邮电出版社独家出版。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。

# 献词

献给我桌上的唱片机。感谢你陪伴我完成本书。我保证，很快你就会有新唱针了。

——B.P.

献给我的爸爸David，他教我懂得辛苦工作的意义。献给我的妈妈Lisa，她一直督促我去做正确的事。

——C.S.

献给我的爸爸Dave Vadas，他激励并支持我投身计算机行业。献给我的妈妈Joan Vadas，在这么多年的浮浮沉沉里，她总能让我保持乐观。（她会给我支招：心情不好的时候就看集《黄金女郎》吧！）

——K.M.

# 致 谢

这是本书第3版。我们常说，当然，也应再三强调：仅凭作者是无法成书的。这背后是团队的力量：合作者、责任编辑和支持者。没有他们，想抓住重点并撰写出这么多的出版素材肯定不可能。

- Brian Hardy和Bill很有雄心，他们从无到有，写出了本书第1版。真了不起。
- 感谢我们Android开发团队的同事Andrew Lunsford、Bolat Kerimbaev、Brian Gardner、David Greenhalgh、Josh Skeen、Matt Compton、Paul Turner和Rashad Cureton。他们一直用这些还不够完善的材料教学，并提出了不少宝贵建议，也修改了一些错误。能和这样有趣、有才的团队一起工作，此生无憾。在Big Nerd Ranch工作的日子，每一天都是享受。
- 特别感谢Andrew。他为本书同步更新了一大批Android Studio截图。他很细心，不放过任何细节，说话还幽默辛辣，令人欣赏。
- Zack Simon，说起话来轻声细语，是我们Big Nerd Ranch了不起的天才设计师。他不声不响地更新了附在书后的Android开发速查表，给了我们一个大大的惊喜。Zack，谢谢你！如果速查表用着不错，你也去谢谢他吧！
- 感谢Kar Loong Wong，他重新设计了crime应用列表屏。只要他多伸手，本书的应用肯定会越来越好看。
- 感谢Mark Dalrymple，他审阅了constraint layout这部分内容，使之更加准确、完善。凑巧碰到他的话，如果你也在搞constraint layout，千万记得请他把关，他可是这方面的专家。不搞也没关系，可以请多才多艺的他扎些气球小动物玩玩。
- 感谢Aaron Hillegass。他若不创建Big Nerd Ranch公司，这一切都无从谈起。
- 感谢我们的编辑Elizabeth Holaday。据说，著名的“垮掉的一代”文学作家William S. Burroughs有时会把自己的作品分成多个部分，抛向空中，然后以稿件的落地顺序出书。要不是Liz，在遇到困惑、一时冲动时，相信我们也会这么做。在她的指导下，我们才能有的放矢，写出清晰、简洁的书稿。
- 感谢Ellie Volckhausen为本书设计了封面。
- 感谢我们的文字编辑Anna Bentley和审稿编辑Simone Payment。感谢她俩的打磨完善。
- 感谢IntelligentEnglish.com网站的Chris Loper。他设计并制作了本书的纸质版和电子版。他的DocBook工具简直太好用了。

最后感谢我们的学员。我们之间有个反馈环：我们以本书内容教学，他们不断给予反馈。没有这个反馈环，就没有这本书，即便有，也不会越来越完善。如果说Big Nerd Ranch公司的图书够特别（希望如此），功劳就在于这个反馈环。再次感谢。

# 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的人所理解的东西。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——使用Java语言。但仅掌握Java远远不够，还需要通过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。

该我们登场了。在Big Nerd Ranch，我们认为，要成为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千名专业的Android开发人员。本书将指导你完成多个Android应用的开发，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。以此为起点，深入学习，逐渐成长为一名合格的Android开发者。

## 阅读前提

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。

如果不熟悉这些概念，很可能刚翻几页就已无法继续下去。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知识忘得差不多了，阅读本书应该不会有太大的问题。对于接口、匿名内部类等重要的Java语言点，我们会提供必要的简短回顾。建议在学习过程中手边备上一本Java参考书，方便查阅。

## 第3版有哪些新内容

本书第3版介绍了一些新工具：constraint layout（包括其编辑器）和数据绑定（data binding）。



新增加了几个章节，内容涉及单元测试、辅助功能（accessibility）、MVVM架构和应用本地化。本书末尾还介绍了Android的新运行时权限系统。此外，我们还进一步扩充了挑战练习和深入学习部分的内容，并修订了全书的一些不够完善的部分。

## 如何使用本书

本书基于Big Nerd Ranch培训机构的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，建议不要跳读，以免学习效果大打折扣。显然，本书不适合作为参考书。本书旨在帮你跨越学习的初始障碍，进而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。

我们为学员提供了良好的培训环境：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，以及随时答疑解惑的指导老师。

你同样需要类似的良好环境。因此，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) 参与本书论坛的交流讨论（[forums.bignerdranch.com](http://forums.bignerdranch.com)）；
- (4) 寻求Android开发高手的帮助。

## 本书内容

本书带你学习开发8个Android应用。有些应用很简单，一章即可讲完；有些则相对复杂。最复杂的一个应用跨越了13章。通过这些精心编排的应用，你能学到很多重要的理论知识和开发技巧，并获得最直接的开发经验。

### ❑ GeoQuiz

本书的第一个应用，借此学习Android应用的基本组成、activity、界面布局以及显式intent。

### ❑ CriminalIntent

本书中最复杂的应用，用来记录办公室同事的种种陋习。借此应用学习fragment、主从用户界面、list-backed用户界面、菜单选项、相机调用、隐式intent等内容。

### ❑ BeatBox

一个可以震慑坏人的应用，借此深入学习fragment、媒体文件的播放与控制、MVVM架构、数据绑定、单元测试、主题以及drawable资源。

### ❑ NerdLauncher

一个个性化启动器，借此深入学习intent以及Android任务。

### ❑ PhotoGallery

一个从Flickr网站下载并显示照片的客户端应用，借此学习Android服务、多线程、网络内容下载等知识。

#### ❑ DragAndDraw

一个简单的画图应用，借此学习触摸手势事件处理以及如何创建个性化视图。

#### ❑ Sunset

一个漂亮的日落动画应用，借此学习Android动画。

#### ❑ Locatr

一个查询当前位置的Flickr图片并显示在地图上的应用。借此应用学习如何使用定位服务和地图。

## 挑战练习

大部分章末都配有练习题。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。强烈建议你完成这些挑战练习。在练习过程中，尝试另辟蹊径，探索自己独特的学习之路。这有助于巩固所学知识，增强未来开发应用的信心。

若遇到一时难以解决的问题，请访问论坛<http://forums.bignerdranch.com>求助。

## 深入学习

部分章末还包含一块名为“深入学习”的内容。这些内容针对相应章节的知识点，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，但还是希望你有兴趣阅读并有所收获。

## 编码风格

有别于其他Android开发学习社区的编码风格，我们有自己的偏好，主要有以下两个方面。

#### ❑ 在监听器代码部分使用匿名内部类

这纯属个人偏好。我们认为，使用匿名内部类，代码可以更简练，监听器实现方法更一目了然。尽管在高性能要求的场景下或大型应用中，匿名内部类可能会有一些问题，但大多数情况下没有问题。

#### ❑ 自第7章引入fragment后，后续所有用户界面都使用它

我们有理由坚持这一点。相信我们，使用得当的话，fragment就是Android开发人员手中的一大利器。一旦适应，用起来也没那么难。相比activity，fragment在创建和显示用户界面时更加灵活，优势明显，值得为此付出努力。

## 版式说明

为方便阅读，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会打上删除线。例如，在以下实现代码里，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);
}
```

## Android 版本

本书教学主要针对当前广泛使用的各个系统版本（Android 4.4 至 Android 7.1）。虽然更老的系统版本仍有人在用，但对于大多数开发者来说，为这部分人开发应用就是个赔本的买卖。如果应用确实需要支持 Android 4.4 之前的系统版本，请参考本书第2版（Android 4.1及以上版本）和第1版（Android 2.3及以上版本）的相关内容。

Google还会不断地发布新版本的Android系统。请放心，Android支持向后兼容（详见第6章），即便有了新系统，本书所授知识也不会过时。而且，通过[forums.bignerdranch.com](http://forums.bignerdranch.com)论坛，我们也会不断跟踪Android开发新动向，及时为你提供开发指导和支持。

## 电子书

扫描如下二维码，即可购买本书电子版。



# 开发必备工具

开始学习前，你需要安装Android Studio。基于流行的IntelliJ IDEA创建，Android Studio是用于Android开发的一套集成开发工具。

Android Studio的安装包括如下内容。

- ❑ Android SDK

最新版本的Android SDK。

- ❑ Android SDK工具和平台工具

用来测试与调试应用的一套工具。

- ❑ Android模拟器系统镜像

用来在不同虚拟设备上开发和测试应用。

撰写本书时，Google一直在积极开发和更新Android Studio版本。因此，请注意了解当前版本和本书所用版本之间的差异。如需帮助，请访问[forums.bignerdranch.com](http://forums.bignerdranch.com)论坛。

## Android Studio的下载与安装

可以从Android开发者网站下载Android Studio：[developer.android.com/sdk/](http://developer.android.com/sdk/)。

首次安装的话，还需从[www.oracle.com](http://www.oracle.com)下载并安装Java开发工具箱（JDK 8）。

如仍有疑问，请访问网址[developer.android.com/sdk/](http://developer.android.com/sdk/)寻求帮助。

## 下载早期版本的SDK

Android Studio自带最新版本的SDK和模拟器系统镜像。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。在Android Studio中，选择Tools → Android → SDK Manager菜单项，如图0-1所示。（已创建并打开了新项目时，Tools菜单才可见。还没创建过项目的話，可在Android开发向导界面，在Quick Start区域，选择Configure → SDK Manager来启动SDK管理器。）

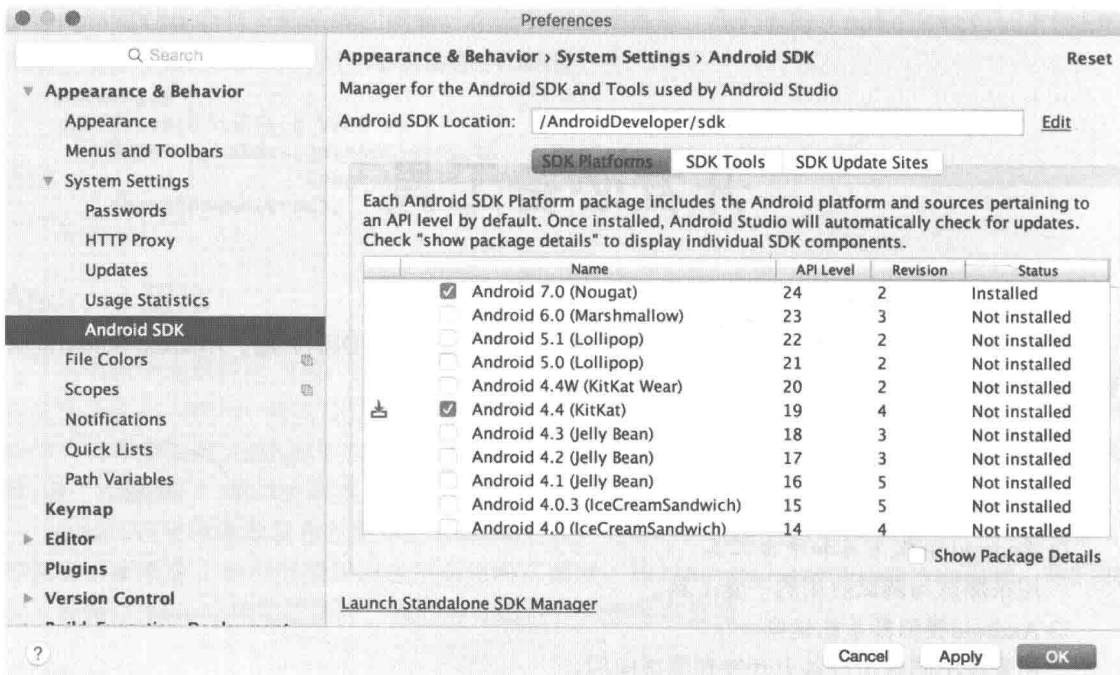


图0-1 Andriod SDK管理器

选择并安装需要的Android版本和工具。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最新发布内容，比如新系统平台或新版本工具等。

## 硬件设备

模拟器是测试应用的好帮手，但需测试应用性能时，物理Android设备无可替代。如果手头有物理设备，建议按需使用。

# 目 录

|                                |    |                                 |    |
|--------------------------------|----|---------------------------------|----|
| 第 1 章 Android 开发初体验 .....      | 1  | 2.7 挑战练习：为 TextView 添加监听器 ..... | 42 |
| 1.1 Android 开发基础 .....         | 2  | 2.8 挑战练习：添加后退按钮 .....           | 42 |
| 1.2 创建 Android 项目 .....        | 2  | 2.9 挑战练习：从按钮到图标按钮 .....         | 42 |
| 1.3 Android Studio 使用导航 .....  | 6  | 第 3 章 activity 的生命周期 .....      | 45 |
| 1.4 用户界面设计 .....               | 6  | 3.1 日志跟踪理解 activity 生命周期 .....  | 46 |
| 1.4.1 视图层级结构 .....             | 10 | 3.1.1 输出日志信息 .....              | 46 |
| 1.4.2 组件属性 .....               | 10 | 3.1.2 使用 LogCat .....           | 48 |
| 1.4.3 创建字符串资源 .....            | 11 | 3.1.3 activity 生命周期实例解析 .....   | 50 |
| 1.4.4 预览布局 .....               | 12 | 3.2 设备旋转与 activity 生命周期 .....   | 52 |
| 1.5 从布局 XML 到视图对象 .....        | 13 | 3.3 保存数据以应对设备旋转 .....           | 56 |
| 1.6 组件的实际应用 .....              | 16 | 3.4 再探 activity 生命周期 .....      | 58 |
| 1.6.1 引用组件 .....               | 17 | 3.5 深入学习：activity 内存清理现状 .....  | 59 |
| 1.6.2 设置监听器 .....              | 17 | 3.6 深入学习：日志记录的级别与方法 .....       | 60 |
| 1.7 创建提示消息 .....               | 19 | 3.7 挑战练习：禁止一题多答 .....           | 61 |
| 1.8 使用模拟器运行应用 .....            | 21 | 3.8 挑战练习：评分 .....               | 61 |
| 1.9 深入学习：Android 编译过程 .....    | 23 | 第 4 章 Android 应用的调试 .....       | 62 |
| 1.10 关于挑战练习 .....              | 25 | 4.1 异常与栈跟踪 .....                | 63 |
| 1.11 挑战练习：定制 toast 消息 .....    | 25 | 4.1.1 诊断应用异常 .....              | 64 |
| 第 2 章 Android 与 MVC 设计模式 ..... | 26 | 4.1.2 记录栈跟踪日志 .....             | 65 |
| 2.1 创建新类 .....                 | 26 | 4.1.3 设置断点 .....                | 66 |
| 2.2 Android 与 MVC 设计模式 .....   | 29 | 4.1.4 使用异常断点 .....              | 68 |
| 2.3 更新视图层 .....                | 31 | 4.2 Android 特有的调试工具 .....       | 70 |
| 2.4 更新控制器层 .....               | 33 | 4.2.1 使用 Android Lint .....     | 70 |
| 2.5 在物理设备上运行应用 .....           | 37 | 4.2.2 R 类的问题 .....              | 71 |
| 2.5.1 连接设备 .....               | 37 | 4.3 挑战练习：探索布局检查器 .....          | 72 |
| 2.5.2 配置设备用于应用开发 .....         | 37 | 4.4 挑战练习：探索内存分配跟踪 .....         | 72 |
| 2.6 添加图标资源 .....               | 38 | 第 5 章 第二个 activity .....        | 74 |
| 2.6.1 向项目中添加资源 .....           | 39 | 5.1 创建第二个 activity .....        | 75 |
| 2.6.2 在 XML 文件中引用资源 .....      | 41 |                                 |    |

|                                   |     |                                       |     |
|-----------------------------------|-----|---------------------------------------|-----|
| 5.1.1 创建新的 activity               | 75  | 7.5.1 定义 CrimeFragment 的布局            | 115 |
| 5.1.2 创建新的 activity 子类            | 78  | 7.5.2 创建 CrimeFragment 类              | 117 |
| 5.1.3 在 manifest 配置文件中声明 activity | 78  | 7.6 向 FragmentManager 添加 UI fragment  | 122 |
| 5.1.4 为 QuizActivity 添加 CHEAT 按钮  | 79  | 7.6.1 fragment 事务                     | 123 |
| 5.2 启动 activity                   | 81  | 7.6.2 FragmentManager 与 fragment 生命周期 | 125 |
| 5.3 activity 间的数据传递               | 83  | 7.7 采用 fragment 的应用架构                 | 126 |
| 5.3.1 使用 intent extra             | 83  | 7.8 深入学习: fragment 与支持库               | 127 |
| 5.3.2 从子 activity 获取返回结果          | 86  | 7.9 深入学习: 为什么优先使用支持库版 fragment        | 128 |
| 5.4 activity 的使用与管理               | 91  |                                       |     |
| 5.5 挑战练习: 堵住漏洞                    | 93  |                                       |     |
| 第 6 章 Android SDK 版本与兼容           | 94  |                                       |     |
| 6.1 Android SDK 版本                | 94  | 第 8 章 使用 RecyclerView 显示列表            | 129 |
| 6.2 Android 编程与兼容性问题              | 95  | 8.1 升级 CriminalIntent 应用的模型层          | 130 |
| 6.2.1 比较合理的版本                     | 95  | 8.2 使用抽象 activity 托管 fragment         | 132 |
| 6.2.2 SDK 最低版本                    | 96  | 8.2.1 通用型 fragment 托管布局               | 133 |
| 6.2.3 SDK 目标版本                    | 97  | 8.2.2 抽象 activity 类                   | 133 |
| 6.2.4 SDK 编译版本                    | 97  | 8.3 RecyclerView、ViewHolder 和 Adapter | 137 |
| 6.2.5 安全添加新版本 API 中的代码            | 97  | 8.3.1 ViewHolder 和 Adapter            | 138 |
| 6.3 使用 Android 开发者文档              | 100 | 8.3.2 使用 RecyclerView                 | 141 |
| 6.4 挑战练习: 报告编译版本                  | 101 | 8.3.3 列表项视图                           | 142 |
| 6.5 挑战练习: 限制作弊次数                  | 102 | 8.3.4 实现 ViewHolder 和 Adapter         | 143 |
| 第 7 章 UI fragment 与 fragment 管理器  | 103 | 8.4 绑定列表项                             | 145 |
| 7.1 UI 设计的灵活性需求                   | 104 | 8.5 响应点击                              | 147 |
| 7.2 引入 fragment                   | 104 | 8.6 深入学习: ListView 和 GridView         | 148 |
| 7.3 着手开发 CriminalIntent           | 105 | 8.7 深入学习: 单例                          | 148 |
| 7.3.1 创建新项目                       | 107 | 8.8 挑战练习: RecyclerView ViewType       | 149 |
| 7.3.2 两类 fragment                 | 108 |                                       |     |
| 7.3.3 在 Android Studio 中增加依赖关系    | 108 | 第 9 章 使用布局与组件创建用户界面                   | 150 |
| 7.3.4 创建 Crime 类                  | 111 | 9.1 使用图形布局工具                          | 151 |
| 7.4 托管 UI fragment                | 112 | 9.2 引入 ConstraintLayout               | 151 |
| 7.4.1 fragment 的生命周期              | 112 | 9.2.1 使用 ConstraintLayout             | 152 |
| 7.4.2 托管的两种方式                     | 113 | 9.2.2 约束编辑器                           | 153 |
| 7.4.3 定义容器视图                      | 113 | 9.2.3 腾出空间                            | 155 |
| 7.5 创建 UI fragment                | 115 | 9.2.4 添加组件                            | 157 |
|                                   |     | 9.2.5 约束的 XML 形式                      | 159 |
|                                   |     | 9.2.6 编辑属性                            | 160 |
|                                   |     | 9.2.7 动态设置列表项                         | 162 |

|                                                                |     |                                                |     |
|----------------------------------------------------------------|-----|------------------------------------------------|-----|
| 9.3 深入学习布局属性 .....                                             | 163 | 第 12 章 对话框 .....                               | 189 |
| 9.3.1 dp、sp 以及屏幕像素密度 .....                                     | 163 | 12.1 创建 DialogFragment .....                   | 190 |
| 9.3.2 边距与内边距 .....                                             | 165 | 12.1.1 显示 DialogFragment .....                 | 192 |
| 9.3.3 样式、主题及主题属性 .....                                         | 166 | 12.1.2 设置对话框的显示内容 .....                        | 194 |
| 9.3.4 Android 应用的设计原则 .....                                    | 166 | 12.2 fragment 间的数据传递 .....                     | 196 |
| 9.4 图形布局工具使用小结 .....                                           | 167 | 12.2.1 传递数据给<br>DatePickerFragment .....       | 197 |
| 9.5 挑战练习：日期格式化 .....                                           | 167 | 12.2.2 返回数据给 CrimeFragment .....               | 199 |
| 第 10 章 使用 fragment argument .....                              | 168 | 12.3 挑战练习：更多对话框 .....                          | 205 |
| 10.1 从 fragment 中启动 activity .....                             | 168 | 12.4 挑战练习：实现响应式<br>DialogFragment .....        | 205 |
| 10.1.1 附加 extra 信息 .....                                       | 169 | 第 13 章 工具栏 .....                               | 207 |
| 10.1.2 获取 extra 信息 .....                                       | 170 | 13.1 AppCompat .....                           | 207 |
| 10.1.3 使用 Crime 数据更新<br>CrimeFragment 视图 .....                 | 171 | 13.2 工具栏菜单 .....                               | 209 |
| 10.1.4 直接获取 extra 信息的缺点 .....                                  | 172 | 13.2.1 在 XML 文件中定义菜单 .....                     | 210 |
| 10.2 fragment argument .....                                   | 172 | 13.2.2 创建菜单 .....                              | 214 |
| 10.2.1 附加 argument 给 fragment .....                            | 173 | 13.2.3 响应菜单项选择 .....                           | 217 |
| 10.2.2 获取 argument .....                                       | 174 | 13.3 实现层级式导航 .....                             | 218 |
| 10.3 刷新显示列表项 .....                                             | 174 | 13.4 可选菜单项 .....                               | 219 |
| 10.4 通过 fragment 获取返回结果 .....                                  | 176 | 13.4.1 切换菜单项标题 .....                           | 220 |
| 10.5 深入学习：为何要用 fragment<br>argument .....                      | 177 | 13.4.2 “还有个问题” .....                           | 222 |
| 10.6 挑战练习：实现高效的 RecyclerView<br>刷新 .....                       | 178 | 13.5 深入学习：工具栏与操作栏 .....                        | 224 |
| 10.7 挑战练习：优化 CrimeLab 的表现 .....                                | 178 | 13.6 挑战练习：删除 crime 记录 .....                    | 224 |
| 第 11 章 使用 ViewPager .....                                      | 179 | 13.7 挑战练习：复数字符串资源 .....                        | 224 |
| 11.1 创建 CrimePagerActivity .....                               | 180 | 13.8 挑战练习：用于 RecyclerView 的<br>空视图 .....       | 225 |
| 11.1.1 ViewPager 与<br>PagerAdapter .....                       | 181 | 第 14 章 SQLite 数据库 .....                        | 226 |
| 11.1.2 整合并配置使用<br>CrimePagerActivity .....                     | 182 | 14.1 定义 schema .....                           | 226 |
| 11.2 FragmentStatePagerAdapter 与<br>FragmentPagerAdapter ..... | 184 | 14.2 创建初始数据库 .....                             | 227 |
| 11.3 深入学习：ViewPager 的工作原理 .....                                | 186 | 14.2.1 使用 Android Device Monitor<br>查看文件 ..... | 230 |
| 11.4 深入学习：以代码的方式创建视图 .....                                     | 187 | 14.2.2 处理数据库相关问题 .....                         | 231 |
| 11.5 挑战练习：恢复 CrimeFragment 的<br>边距 .....                       | 187 | 14.3 修改 CrimeLab 类 .....                       | 232 |
| 11.6 挑战练习：添加 Jump to First 按钮和<br>Jump to Last 按钮 .....        | 188 | 14.4 写入数据库 .....                               | 233 |
|                                                                |     | 14.4.1 使用 ContentValues .....                  | 233 |
|                                                                |     | 14.4.2 插入和更新记录 .....                           | 234 |
|                                                                |     | 14.5 读取数据库 .....                               | 235 |
|                                                                |     | 14.5.1 使用 CursorWrapper .....                  | 236 |



|                                            |     |                            |     |
|--------------------------------------------|-----|----------------------------|-----|
| 14.5.2 创建模型层对象.....                        | 238 | 第 18 章 应用本地化.....          | 286 |
| 14.6 深入学习：数据库高级主题介绍.....                   | 241 | 18.1 资源本地化.....            | 286 |
| 14.7 深入学习：应用上下文.....                       | 241 | 18.1.1 默认资源.....           | 289 |
| 14.8 挑战练习：删除 crime 记录.....                 | 242 | 18.1.2 检查资源本地化完成情况.....    | 291 |
| 第 15 章 隐式 intent.....                      | 243 | 18.1.3 区域修饰符.....          | 292 |
| 15.1 添加按钮组件.....                           | 244 | 18.2 配置修饰符.....            | 294 |
| 15.2 添加嫌疑人信息至模型层.....                      | 244 | 18.2.1 可用资源优先级排定.....      | 295 |
| 15.3 使用格式化字符串.....                         | 246 | 18.2.2 多重配置修饰符.....        | 296 |
| 15.4 使用隐式 intent.....                      | 248 | 18.2.3 寻找最匹配的资源.....       | 297 |
| 15.4.1 隐式 intent 的组成.....                  | 248 | 18.3 测试备选资源.....           | 298 |
| 15.4.2 发送消息.....                           | 249 | 18.4 挑战练习：日期本地化.....       | 298 |
| 15.4.3 获取联系人信息.....                        | 252 | 第 19 章 Android 辅助功能.....   | 299 |
| 15.4.4 检查可响应任务的 activity.....              | 254 | 19.1 TalkBack.....         | 299 |
| 15.5 挑战练习：ShareCompat.....                 | 256 | 19.1.1 点击浏览.....           | 301 |
| 15.6 挑战练习：又一个隐式 intent.....                | 256 | 19.1.2 线性浏览.....           | 301 |
| 第 16 章 使用 intent 拍照.....                   | 258 | 19.2 实现非文字型元素可读.....       | 303 |
| 16.1 布置照片.....                             | 258 | 19.2.1 添加内容描述.....         | 303 |
| 16.2 文件存储.....                             | 260 | 19.2.2 实现组件可聚焦.....        | 305 |
| 16.2.1 使用 FileProvider.....                | 261 | 19.3 提升辅助体验.....           | 305 |
| 16.2.2 指定照片存放位置.....                       | 262 | 19.4 深入学习：使用辅助功能扫描器.....   | 309 |
| 16.3 使用相机 intent.....                      | 263 | 19.5 挑战练习：优化列表项.....       | 311 |
| 16.4 缩放和显示位图.....                          | 265 | 19.6 挑战练习：补全上下文信息.....     | 311 |
| 16.5 功能声明.....                             | 268 | 19.7 挑战练习：事件主动通知.....      | 311 |
| 16.6 挑战练习：优化照片显示.....                      | 269 | 第 20 章 数据绑定与 MVVM.....     | 313 |
| 16.7 挑战练习：优化缩略图加载.....                     | 269 | 20.1 为何要用 MVVM 架构.....     | 314 |
| 第 17 章 双版面主从用户界面.....                      | 270 | 20.2 创建 BeatBox 应用.....    | 314 |
| 17.1 增加布局灵活性.....                          | 271 | 20.3 导入 assets.....        | 319 |
| 17.1.1 修改 SingleFragment-<br>Activity..... | 272 | 20.4 处理 assets.....        | 321 |
| 17.1.2 创建包含两个 fragment 容器<br>的布局.....      | 272 | 20.5 使用 assets.....        | 323 |
| 17.1.3 使用别名资源.....                         | 274 | 20.6 绑定数据.....             | 325 |
| 17.1.4 创建平板设备专用可选<br>资源.....               | 275 | 20.6.1 创建视图模型.....         | 327 |
| 17.2 activity：fragment 的托管者.....           | 276 | 20.6.2 绑定至视图模型.....        | 328 |
| 17.3 深入学习：设备屏幕尺寸的确定.....                   | 284 | 20.6.3 绑定数据观察.....         | 330 |
| 17.4 挑战练习：添加滑动删除功能.....                    | 285 | 20.7 访问 assets.....        | 332 |
|                                            |     | 20.8 深入学习：数据绑定再探.....      | 333 |
|                                            |     | 20.8.1 lambda 表达式.....     | 333 |
|                                            |     | 20.8.2 更多语法糖.....          | 333 |
|                                            |     | 20.8.3 BindingAdapter..... | 333 |