



移动开发丛书

从基本语法

到应用开发

拒绝纸上谈兵

电商App 开发实录

尽现Kotlin魅力

Kotlin

从零到精通

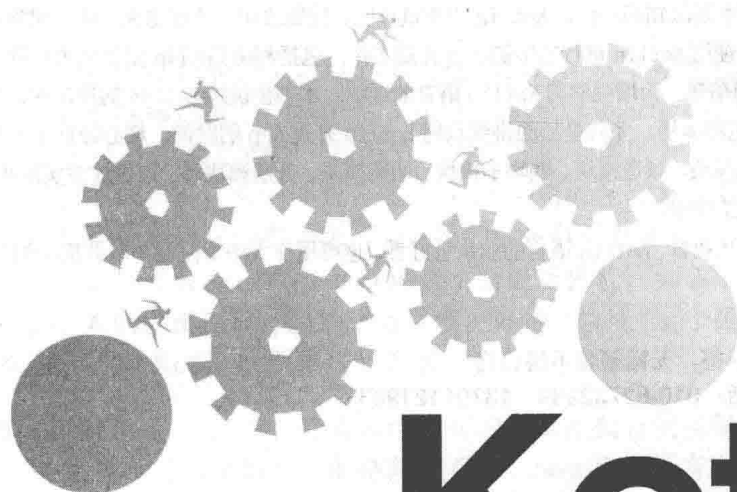
Android开发

• 欧阳燊 编著 •

Android老司机带你玩转Kotlin



清华大学出版社



移动开发丛书

Kotlin

从零到精通

Android开发

常州大学图书馆藏书章

• 欧阳燊 编著 •

清华大学出版社
北京

内 容 简 介

本书是一部讲解 Kotlin 语言的入门书籍,从 Kotlin 语言的基本语法一直讲到如何将其运用于 Android 开发。由浅入深、从理论到实战,帮助读者快速掌握 Kotlin 开发技巧。

全书共有 10 章内容,可分为三大部分:第一部分即第 1 章,主要介绍 Kotlin 语言的开发环境搭建;第二部分包含第 2~5 章,主要介绍 Kotlin 的基本语法知识,包括 Kotlin 的变量声明、控制语句、函数定义、类与对象等;第三部分包含第 6~10 章,主要介绍如何使用 Kotlin 进行实际的 App 开发工作,包括利用 Kotlin 操作简单控件、复杂控件、数据存储、自定义控件、网络通信等。为增强学习 Kotlin 语言的趣味,本书在讲解 Kotlin 的用法时,特别注意结合生活中的具体案例,并加以示范和运用。尤其是后面讲到利用 Kotlin 开发 App 的时候,精心设计了数个电商 App 的实战模块,例如电商 App 的登录模块、频道模块、购物车模块、团购模块、升级模块等。通过这些实战小项目,读者可迅速将 Kotlin 应用于 App 开发工作中。

本书适用于 Android 开发的广大从业者、Kotlin 语言的业余爱好者,也可用作大中专院校与培训机构的 Kotlin 课程教材。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Kotlin 从零到精通 Android 开发/欧阳桑编著. —北京:清华大学出版社, 2018

(移动开发丛书)

ISBN 978-7-302-49814-8

I. ①K… II. ①欧… III. ①JAVA 语言—程序设计 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2018)第 037614 号

责任编辑:王金柱

封面设计:王 翔

责任校对:闫秀华

责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:190mm×260mm

印 张:21.75

字 数:557 千字

版 次:2018 年 4 月第 1 版

印 次:2018 年 4 月第 1 次印刷

印 数:1~3500

定 价:79.00 元

产品编号:077880-01

前言

新技术的发展日新月异，编程语言也不例外，从早期的机器语言到汇编语言，再到以 C 语言为代表的高级语言，一路衍生出了 C++、Java、Objective-C 等庞大的编程语言家族。其中，Java 经过多年的发展已经是一枝独秀，不但在服务端的开发中占据优势，而且在客户端的安卓开发上也形成垄断之势。不过，由于 Java 语言诞生较早（诞生于 20 世纪 90 年代中期），使得它不可避免地存在一些先天不足，比如业务代码过于冗长、处理逻辑不够灵活、安全隐患层出不穷等。鉴于此，一方面 Java 语言不断更新换代，到 2017 年已经迭代到了 Java 9 版本；另一方面，人们也试图设计新的语言以便更好地“填坑”，于是涌现了 Scala、Groovy、Clojure 等新兴语言，而 Kotlin 就是这些新兴语言中的佼佼者。

Kotlin 问世于 2011 年，作为后起之秀的它虽然拥有代码简洁、函数式编程、更安全健壮、百分百兼容 Java 等诸多特性，但是前有 C++、Java 等老语言根深叶茂，后有 Python、Go 等新兴语言紧追不舍，Kotlin 头几年的发展一直不温不火。直到这两年，在 JetBrains、Google 等公司的大力扶持之下，Kotlin 的发展才驶上了快车道，先是在 2016 年 2 月推出 Kotlin 1.0 发布版，再是谷歌公司在 2017 年 5 月宣布将 Kotlin 作为 Android 的官方开发语言，然后在 2017 年 10 月推出的 Android Studio 3.0 正式集成了 Kotlin 开发环境，紧接着更完善的 Kotlin 1.2 正式版在 2017 年 11 月发布。正如当初 Android Studio 取代 Eclipse 成为 Android 开发的主流开发工具一样，在可预见的未来，Kotlin 必将逐步取代 Java 成为主流的 App 开发语言。

被寄予厚望的 Kotlin 在编程工作中给开发者带来的巨大便利毋庸置疑，大量的开发实践表明，实现同样功能的 Kotlin 代码往往只有对应 Java 代码的三分之一。并且 Kotlin 的语法兼容并蓄、易懂易学，只要开发者拥有任何一门高级语言的编程基础，再配合一本合适的 Kotlin 入门教程，短时间内即可成为 Kotlin 熟练开发者。正因为 Kotlin 是如此的简单易用，它的代码也是如此的简洁明了，所以倘若介绍 Kotlin 语法的教程还在长篇大论，那它一定是在夸夸其谈地“耍流氓”。Kotlin 的设计理念是尽可能的简易，而不是抛出一堆令人生畏的烦琐概念，因此本书在介绍 Kotlin 用法的时候，也秉承了与之相符的一贯理念，即凡是能够简单处理的事情，决不拐弯抹角、拖泥带水。

本书既是一本 Kotlin 语法的入门教程，也是讲述 Kotlin 开发 App 的教程，一方面介绍 Kotlin 语言的基本语法，另一方面介绍 Kotlin 在安卓开发中的实际应用，可谓是结合理论、联系实战，方便读者迅速将 Kotlin 运用于日常的 App 开发工作之中，更好、更快地将学习成果展现出来，起到立竿见影的工作成效。当然，本书的侧重点在于教会读者利用 Kotlin 进行安卓开发工作，故而在有限的内容篇幅上有所取舍，比如服务端的 Kotlin 开发着墨不多，另外阐述了如何使用 Kotlin 实现常见的几种 App 开发技术，其余的 App 开发涉及的 Kotlin 技术即可触类旁通。如果读者想要了解更详细、更具体的 App 开发技能，可参见笔者的另一部 App 开发专著《Android Studio 开发实战：从零基础到 App 上线》。

全书共有 10 章内容，循序渐进，可分为三大部分：第一部分即第 1 章，主要介绍 Kotlin 语言的开发环境搭建；第二部分包含第 2~5 章，主要介绍 Kotlin 的基本语法知识，包括 Kotlin 的变量声明、控制语句、函数定义、类与对象等；第三部分包含第 6~10 章，主要介绍如何使用 Kotlin 进行实际的 App 开发工作，包括如何通过 Kotlin 使用简单控件、如何通过 Kotlin 操纵复杂控件、如何通过 Kotlin 进行数据存储、如何通过 Kotlin 自定义控件、如何通过 Kotlin 实现网络通信等。通过本书这 10 章的学习，读者应该能够掌握 Kotlin 的大部分常用语法，并将其得心应手地运用于 App 开发工作中，你会发现多了一门可供选择的 App 开发语言是多么奇妙的事情。

从零开始学 Kotlin 其实指的是 Kotlin 零基础，并非编程零基础。在学习本书之前，读者应当掌握至少一门高级开发语言。如果没有任何编程基础就来学习 Kotlin，这是不现实也是不可取的，因为短期之内各公司不会招聘只会 Kotlin 的程序员，而且 Kotlin 在 Android 开发中取代 Java 也必然是个缓慢的进程。所以学习 Kotlin 不提倡急于求成，但这并不意味着 App 开发者可以对 Kotlin 熟视无睹，任何一个新事物都有其发展壮大过程，同时机会都是留给有准备的人。与其等 Kotlin 形成燎原之势才后知后觉地学习它，不如现在就未雨绸缪地掌握它，技术投资得越早，未来的开发收益就越大。

本书的所有代码例子都基于 Android Studio 3.0 和 Kotlin 1.2 开发，并使用 API 27 的 SDK (Android 8.1) 编译与调试通过。所有的附录源代码均可在网络上下载，具体下载方式可访问笔者的博客 <http://blog.csdn.net/aqi00>。读者也可以从以下地址下载本书源代码：

https://pan.baidu.com/s/1ceRZzDK4_zT-uQHqy2WFHw (注意区分数字和英文字母大小写)

如果下载有问题，请发送电子邮件至 booksaga@126.com，邮件标题为“Kotlin 从零到精通 Android 开发配书源代码”获得帮助。

读者在阅读本书时，若对书中内容有疑问，也可在该博客上留言。或者关注笔者的微信公众号“老欧说安卓”，更快更方便地阅读技术干货。

最后感谢王金柱编辑以及各位出版社同仁的热情指点和密切配合，感谢我的家人一直以来的支持，如果没有大家的鼎力协助，就没有本书的顺利完成。

欧阳荣

2018 年 1 月

目 录

第 1 章 搭建 Kotlin 开发环境	1
1.1 Kotlin 与 Android 开发的关系	1
1.1.1 Kotlin 语言简介	1
1.1.2 Android Studio 的官方开发语言	2
1.2 Kotlin 开发工具	2
1.2.1 安装 Android Studio	2
1.2.2 启动 Android Studio	4
1.2.3 创建 Kotlin 工程	5
1.2.4 新建 Kotlin 文件	8
1.3 SDK 安装与插件升级	10
1.3.1 安装最新版 SDK	10
1.3.2 升级 Gradle 插件	11
1.3.3 升级 Kotlin 插件	12
1.4 Kotlin 简单配置	14
1.4.1 调整 Kotlin 编译配置	14
1.4.2 修改编译配置文件	15
1.4.3 Java 代码转 Kotlin 代码	16
1.5 Kotlin 相关技术	18
1.5.1 Kotlin 代码与 Java 代码 PK	18
1.5.2 Anko 库	21
1.5.3 Lambda 表达式	22
1.6 小结	23
第 2 章 数据类型	24
2.1 基本数据类型	24
2.1.1 基本类型的变量声明	24
2.1.2 简单变量之间的转换	25
2.2 数组	26
2.2.1 数组变量的声明	27
2.2.2 数组元素的操作	28
2.3 字符串	29
2.3.1 字符串与基本类型的转换	29
2.3.2 字符串的常用方法	29
2.3.3 字符串模板及其拼接	30
2.4 容器	31
2.4.1 容器的基本操作	31

2.4.2	集合 Set/MutableSet	32
2.4.3	队列 List/MutableList	34
2.4.4	映射 Map/MutableMap	36
2.5	小结	38
第 3 章	控制语句	39
3.1	条件分支	39
3.1.1	简单分支	39
3.1.2	多路分支	40
3.1.3	类型判断	42
3.2	循环处理	43
3.2.1	遍历循环	44
3.2.2	条件循环	45
3.2.3	跳出多重循环	46
3.3	空安全	48
3.3.1	字符串的有效性判断	48
3.3.2	声明可空变量	49
3.3.3	校验空值的运算符	50
3.4	等式判断	52
3.4.1	结构相等	52
3.4.2	引用相等	53
3.4.3	s 和 in	55
3.5	小结	57
第 4 章	函数运用	58
4.1	函数的基本用法	58
4.1.1	与 Java 声明方式的区别	58
4.1.2	输入参数的格式	59
4.1.3	输出参数的格式	60
4.2	输入参数的变化	62
4.2.1	默认参数	62
4.2.2	命名参数	63
4.2.3	可变参数	64
4.3	几种特殊函数	66
4.3.1	泛型函数	66
4.3.2	内联函数	67
4.3.3	简化函数	68
4.3.4	尾递归函数	69
4.3.5	高阶函数	69
4.4	增强系统函数	71
4.4.1	扩展函数	71

4.4.2 扩展高阶函数	72
4.4.3 日期时间函数	73
4.4.4 单例对象	75
4.5 小结	77
第 5 章 类和对象	78
5.1 类的构造	78
5.1.1 类的简单定义	78
5.1.2 类的构造函数	79
5.1.3 带默认参数的构造函数	81
5.2 类的成员	83
5.2.1 成员属性	83
5.2.2 成员方法	86
5.2.3 伴生对象	87
5.2.4 静态属性	88
5.3 类的继承	89
5.3.1 开放性修饰符	89
5.3.2 普通类继承	91
5.3.3 抽象类	93
5.3.4 接口	94
5.3.5 接口代理	96
5.4 几种特殊类	99
5.4.1 嵌套类	100
5.4.2 内部类	100
5.4.3 枚举类	101
5.4.4 密封类	103
5.4.5 数据类	104
5.4.6 模板类	105
5.5 小结	107
第 6 章 Kotlin 使用简单控件	108
6.1 使用按钮控件	108
6.1.1 按钮 Button	108
6.1.2 复选框 CheckBox	111
6.1.3 单选按钮 RadioButton	112
6.2 使用页面布局	114
6.2.1 线性布局 LinearLayout	114
6.2.2 相对布局 RelativeLayout	118
6.2.3 约束布局 ConstraintLayout	119
6.3 使用图文控件	124
6.3.1 文本视图 TextView	124

6.3.2	图像视图 ImageView	127
6.3.3	文本编辑框 EditText	128
6.4	Activity 活动跳转	130
6.4.1	传送配对字段数据	130
6.4.2	传送序列化数据	132
6.4.3	跳转时指定启动模式	134
6.4.4	处理返回数据	137
6.5	实战项目：电商 App 的登录页面	138
6.5.1	需求描述	138
6.5.2	开始热身：提醒对话框 AlertDialog	139
6.5.3	控件设计	141
6.5.4	关键代码	141
6.6	小结	144
第 7 章	Kotlin 操纵复杂控件	145
7.1	使用视图排列	145
7.1.1	下拉框 Spinner	145
7.1.2	列表视图 ListView	149
7.1.3	网格视图 GridView	154
7.1.4	循环视图 RecyclerView	156
7.2	使用材质设计 MaterialDesign	165
7.2.1	协调布局 CoordinatorLayout	165
7.2.2	工具栏 Toolbar	167
7.2.3	应用栏布局 AppBarLayout	169
7.2.4	可折叠工具栏布局 CollapsingToolbarLayout	173
7.2.5	仿支付宝首页的头部伸缩特效	177
7.3	实现页面切换	181
7.3.1	翻页视图 ViewPager	182
7.3.2	碎片 Fragment	184
7.3.3	标签布局 TabLayout	187
7.4	广播收发 Broadcast	190
7.4.1	收发临时广播	191
7.4.2	接收系统广播	194
7.5	实战项目：电商 App 的商品频道	196
7.5.1	需求描述	196
7.5.2	开始热身：下拉刷新布局 SwipeRefreshLayout	197
7.5.3	控件设计	201
7.5.4	关键代码	201
7.6	小结	203

第 8 章 Kotlin 进行数据存储	205
8.1 使用共享参数 SharedPreferences	205
8.1.1 共享参数读写模板 Preference	205
8.1.2 属性代理等黑科技	208
8.1.3 实现记住密码功能	210
8.2 使用数据库 SQLite	211
8.2.1 数据库帮助器 SQLiteOpenHelper	211
8.2.2 更安全的 ManagedSQLiteOpenHelper	213
8.2.3 优化记住密码功能	220
8.3 文件 I/O 操作	222
8.3.1 文件保存空间	222
8.3.2 读写文本文件	224
8.3.3 读写图片文件	225
8.3.4 遍历文件目录	227
8.4 Application 全局变量	228
8.4.1 Application 单例化	228
8.4.2 利用 Application 实现全局变量	231
8.5 实战项目：电商 App 的购物车	232
8.5.1 需求描述	232
8.5.2 开始热身：选项菜单 OptionsMenu	233
8.5.3 控件设计	235
8.5.4 关键代码	236
8.6 小结	240
第 9 章 Kotlin 自定义控件	242
9.1 自定义普通视图	242
9.1.1 构造对象	242
9.1.2 测量尺寸	245
9.1.3 绘制部件	249
9.2 自定义简单动画	252
9.2.1 任务 Runnable	252
9.2.2 进度条 ProgressBar	255
9.2.3 自定义文本进度条	257
9.2.4 实现进度条动画	258
9.3 自定义通知栏	259
9.3.1 通知推送 Notification	260
9.3.2 大视图通知	262
9.3.3 三种特殊的通知类型	265
9.3.4 远程视图 RemoteViews	269
9.3.5 自定义折叠式通知	272

9.4	Service 服务启停	274
9.4.1	普通方式启动服务	274
9.4.2	绑定方式启动服务	277
9.4.3	推送服务到前台	279
9.5	实战项目：电商 App 的生鲜团购	283
9.5.1	需求描述	283
9.5.2	开始热身：震动器 Vibrator	284
9.5.3	控件设计	287
9.5.4	关键代码	287
9.6	小结	289
第 10 章	Kotlin 实现网络通信	291
10.1	多线程技术	291
10.1.1	大线程 Thread 与消息传递	291
10.1.2	进度对话框 ProgressDialog	295
10.1.3	异步任务 doAsync 和 doAsyncResult	297
10.2	访问 HTTP 接口	300
10.2.1	移动数据格式 JSON	301
10.2.2	JSON 串转数据类	303
10.2.3	HTTP 接口调用	304
10.2.4	HTTP 图片获取	306
10.3	文件下载操作	308
10.3.1	下载管理器 DownloadManager	308
10.3.2	自定义文本进度圈	313
10.3.3	在页面上动态显示下载进度	316
10.4	ContentProvider 内容提供	319
10.4.1	内容提供者 ContentProvider	319
10.4.2	内容解析器 ContentResolver	322
10.4.3	内容观察器 ContentObserver	325
10.5	实战项目：电商 App 的自动升级	329
10.5.1	需求描述	329
10.5.2	开始热身：可变字符串 SpannableString	330
10.5.3	控件设计	333
10.5.4	关键代码	334
10.6	小结	337

第 1 章

搭建 Kotlin 开发环境

本章主要介绍 Kotlin 开发环境的搭建过程，首先阐述 Kotlin 语言与 Android 开发之间的关系，接着描述 Kotlin 开发工具，也就是 Android Studio 的安装和启动步骤，然后说明 SDK 及其相关插件的安装与升级方法，接着论述如何对 Kotlin 工程的编译配置进行调整，最后演示 Kotlin 新技术带来哪些革命性的变化。

1.1 Kotlin 与 Android 开发的关系

本节主要介绍 Kotlin 语言与 Android 开发之间的关系，包括 Kotlin 的基本概念及其特殊优势，以及 Kotlin 被谷歌钦定为 Android Studio 官方开发语言之后的发展大事。

1.1.1 Kotlin 语言简介

Kotlin 是一种基于 JVM 的新型编程语言，它完全兼容 Java 语言，Kotlin 代码可以编译成 Java 字节码，也可以编译成 JavaScript，方便在没有 JVM 的设备上运行。与流行的 Java 语言比较，Kotlin 具备下列优势：

- (1) Kotlin 更简洁，完成同样的业务功能，Kotlin 代码通常只有对应 Java 代码的三分之一。
- (2) Kotlin 更安全，它能够在编码阶段就自动检测常见的 BUG，比如引用了空指针等。
- (3) Kotlin 更强大，它提供了扩展函数、默认参数、接口委托、属性代理等 Java 所不具备的高级特性，从而可以完成更复杂的业务逻辑。

1.1.2 Android Studio 的官方开发语言

Kotlin 很早就被运用到 Android 开发之中, 之前一直作为 Android Studio 的插件提供下载, Android Studio 只要安装了 Kotlin 插件, 就能用来开发 Kotlin 编码的 App 工程。

2017 年 5 月, 谷歌宣布将 Kotlin 纳入 Android Studio 开发的官方语言, 这意味着 Android Studio 对 Kotlin 的编译支持会大大增强, 由此掀起了广大安卓开发者学习 Kotlin 编程的热潮。

2017 年 10 月, Android Studio 推出 3.0 正式版, 从 3.0 版本开始, Android Studio 自动集成 Kotlin 插件, 在安装 Android Studio 3.0 时就连带配置了 Kotlin 的开发环境。

2017 年 11 月, Kotlin 语言推出 1.2 发布版, 该版本的 Kotlin 具备更好的跨平台特性, 编译性能也比 1.1 版提高了 25% 左右, 同时也更好地支持 Android 开发。

1.2 Kotlin 开发工具

本节主要介绍 Kotlin 开发环境的搭建以及 Kotlin 工程的基本操作, 包括安装 Android Studio 的具体步骤、启动 Android Studio 的详细配置、如何创建一个 Kotlin 工程、如何新建各种 Kotlin 文件等。

1.2.1 安装 Android Studio

Android Studio 的官方下载页面是 <https://developer.android.google.cn/studio/index.html>, 在这里可以找到 Android Studio 的下载地址与使用教程。首先把最新版本的 Android Studio 下载到电脑本地, 然后双击下载完成的 Android Studio 安装程序, 弹出安装欢迎对话框, 如图 1-1 所示。单击该对话框右下方的“Next”按钮, 跳到下一页的许可同意对话框, 单击“Agree”按钮, 进入下一页的组件选择对话框, 如图 1-2 所示。



图 1-1 Android Studio 的安装欢迎对话框

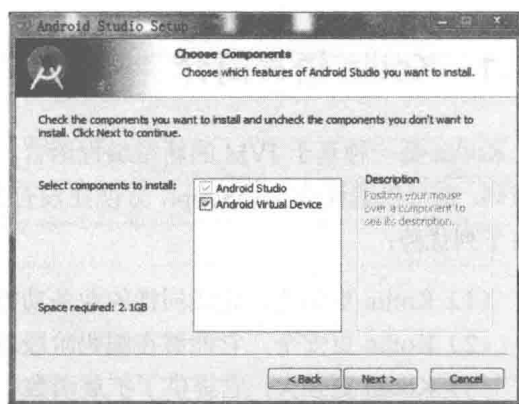


图 1-2 Android Studio 的组件选择对话框

全部勾选, 然后单击“Next”按钮, 跳转到安装路径配置对话框。建议将 Android Studio 安装

在除系统盘外的其他磁盘，比如 D 盘，如图 1-3 所示，然后单击“Next”按钮。在下一个对话框中选择开始菜单的目录，这里使用默认的“Android Studio”，如图 1-4 所示，然后单击“Install”按钮，等待安装过程进行。

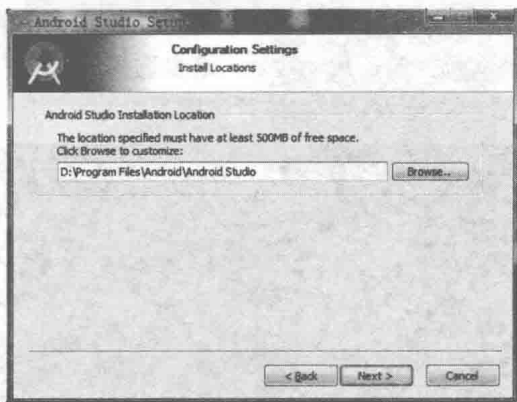


图 1-3 Android Studio 的安装目录对话框

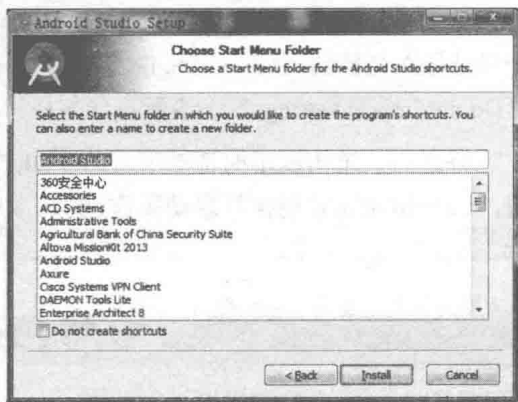


图 1-4 Android Studio 的安装设置对话框

安装过程的进度对话框如图 1-5 所示，进度完成的结果对话框如图 1-6 所示，单击“Next”按钮结束安装操作。

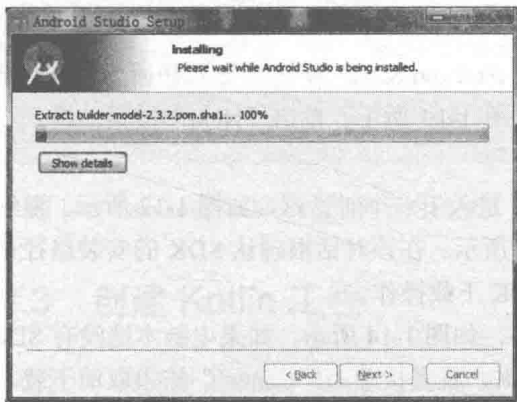


图 1-5 Android Studio 的安装进度对话框

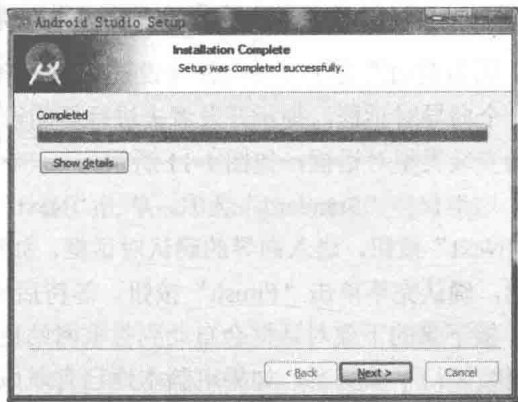


图 1-6 Android Studio 的安装结果对话框

安装完毕后弹出一个提示对话框，如图 1-7 所示，上面默认勾选了“Start Android Studio”，单击右下角的“Finish”按钮即可启动 Android Studio。



图 1-7 Android Studio 的安装完成对话框

1.2.2 启动 Android Studio

首次安装 Android Studio 3.0 会弹出一个提示对话框, 如图 1-8 所示, 在此开发者可以选择从哪个目录导入之前的 Android Studio 设置。为了更好地演示完整的启动过程, 这里选择最下面的选项 “Do not import settings”, 表示不导入任何已有设置, 完全重新开始进行设置。

选好之后, 单击提示对话框下方的 “OK” 按钮, Android Studio 便执行启动操作, 如图 1-9 所示。

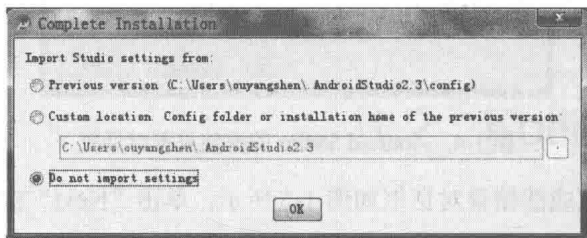


图 1-8 Android Studio 首次安装后的设置导入提示对话框 图 1-9 Android Studio 正在启动的提示对话框

因为前面选定了不导入任何设置重新开始, 所以 Android Studio 将不会导入任何工程, 而是弹出一个向导对话框, 提示开发者去进行新的设置, 如图 1-10 所示。单击 “Next” 按钮, 进入下一页的安装类型对话框, 如图 1-11 所示。

这里保持 “Standard” 选项, 单击 “Next” 按钮, 进入下一个对话框, 如图 1-12 所示。继续单击 “Next” 按钮, 进入向导的确认对话框, 如图 1-13 所示。在该对话框确认 SDK 的安装路径是否正确, 确认完毕单击 “Finish” 按钮, 等待后续的 SDK 下载操作。

接下来的下载对话框会自动到谷歌网站更新组件, 如图 1-14 所示。如果电脑本地没有 SDK, 就继续等待下载更新, 如果电脑本地已有现成的 SDK, 就直接单击 “Cancel” 按钮取消下载, 然后单击 “Finish” 按钮结束设置。最后弹出一个 “Welcome to Android Studio” 欢迎对话框, 如图 1-15 所示。单击第一项的 “Start a new Android Studio project” 即可开始你的 Android 开发之旅。

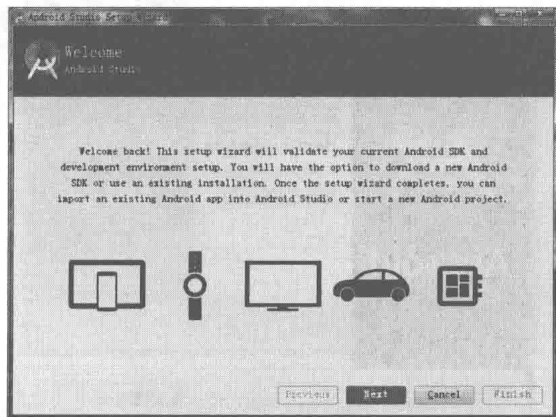


图 1-10 Android Studio 的启动向导对话框 1

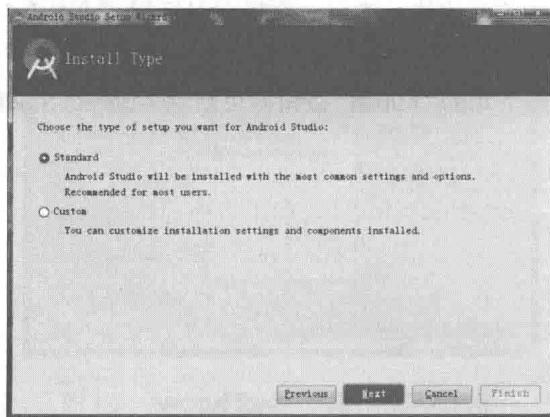


图 1-11 Android Studio 的启动向导对话框 2

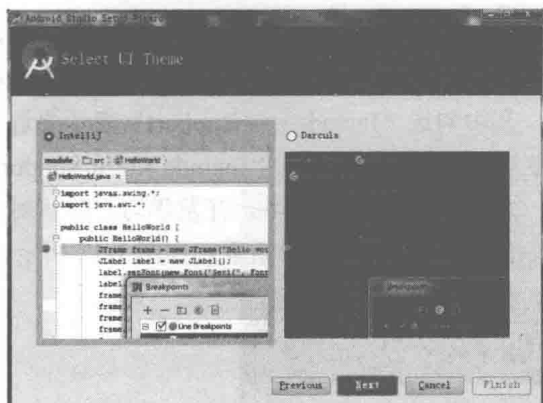


图 1-12 Android Studio 的启动向导对话框 3

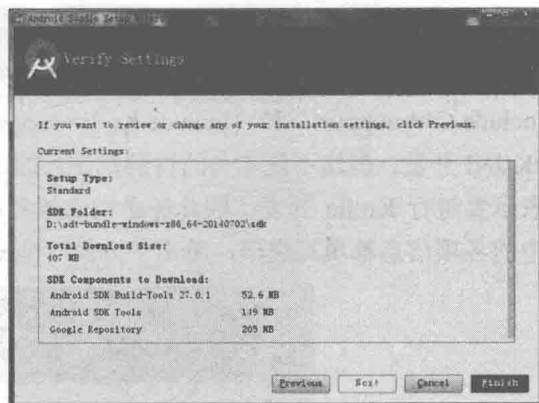


图 1-13 Android Studio 的启动向导对话框 4

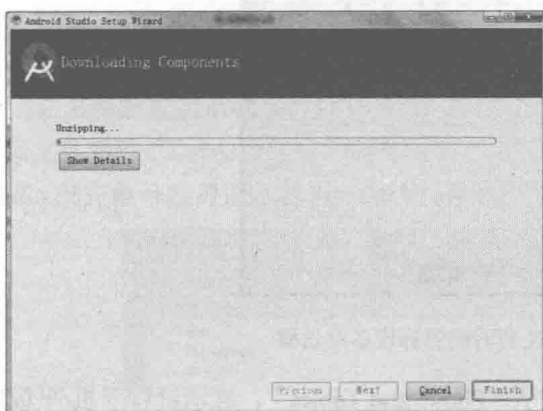


图 1-14 Android Studio 的启动向导对话框 5

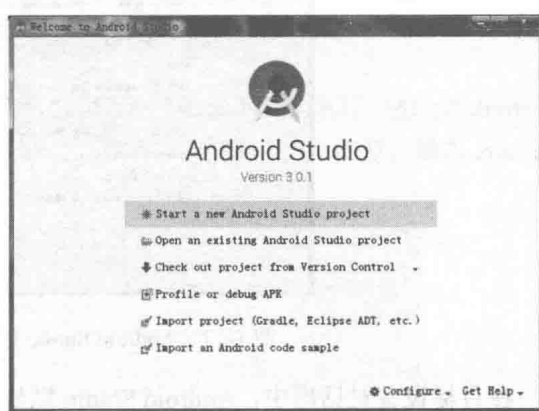


图 1-15 Android Studio 的启动欢迎对话框

1.2.3 创建 Kotlin 工程

1.2.2 小节提到 Android Studio 启动设置完成之后，会弹出欢迎对话框提示创建新的 Android 工程，此时单击第一项的“Start a new Android Studio project”打开工程创建对话框，如图 1-16 所示。

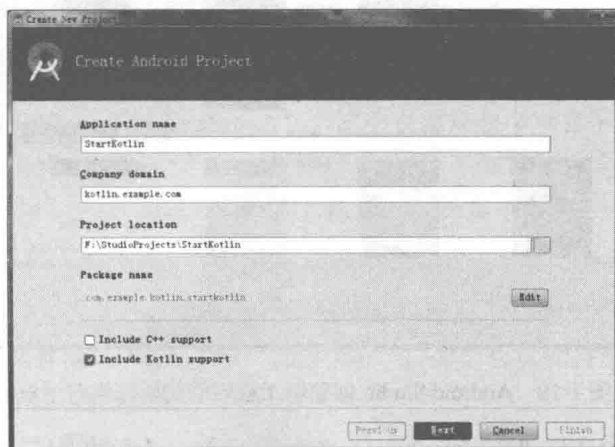


图 1-16 Android Studio 的工程创建对话框

在工程创建对话框中填写应用名称“Application name”以及公司域名“Company domain”，并选择或填写 Android 工程的本地保存路径“Project location”。注意，创建页面下方有两个选项“Include C++ support”和“Include Kotlin support”，其中勾选“Include C++ support”表示要进行 NDK/JNI 开发，但这不是本书的讲解范围，因此不必勾选该复选框；勾选“Include Kotlin support”则表示要进行 Kotlin 开发，因此务必勾选该复选框，才能继续后面的 Kotlin 开发学习。确认对话框中的各项信息都填写完毕，单击下方的“Next”按钮，进入目标设备对话框，如图 1-17 所示。

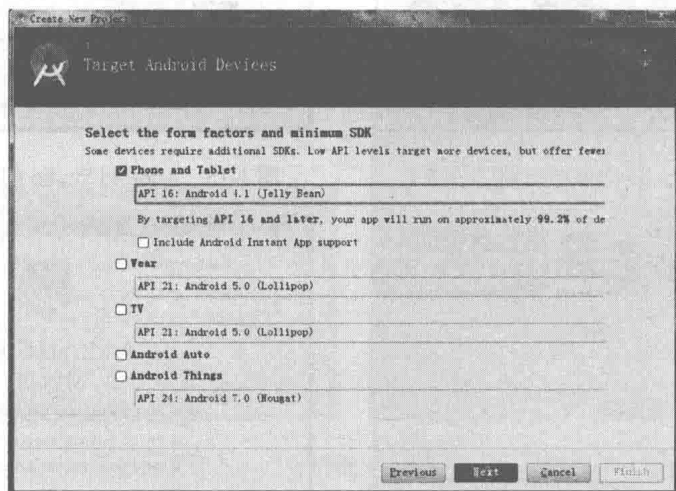


图 1-17 Android Studio 创建新工程时的目标设备对话框

在目标设备对话框中，Android Studio 默认勾选了“Phone and Tablet”，表示进行手机/平板应用开发，下面的 API 最低支持版本原本默认是 API 15，不过因为如通知的新特性从 API 16 开始才支持，所以这里建议把最低版本改为 API 16，接着单击“Next”按钮，进入初始风格对话框，如图 1-18 所示。

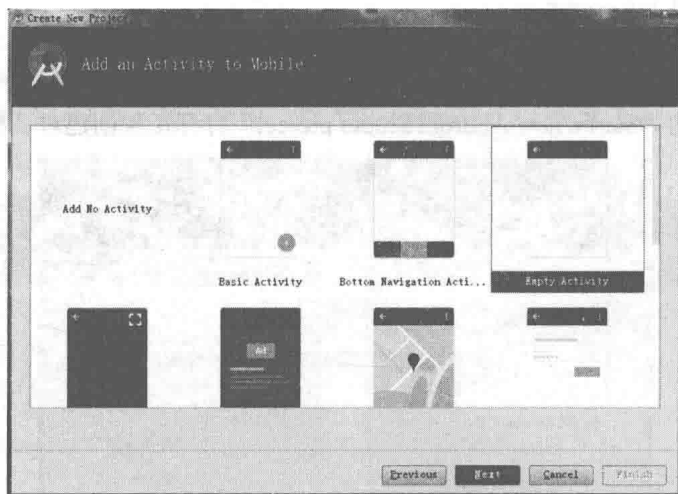


图 1-18 Android Studio 创建新工程时的初始风格对话框

在初始风格对话框中选择“Empty Activity”，然后单击下方的“Next”按钮，进入名称配置对话框，如图 1-19 所示。