

什么？你们的教程都从0开始？起点太高了！明摆着不想让人看懂！  
我们从-1开始！



Using Android Pie to Learn Android Development

# Android 9编程通俗演义

牛搞 著

  
集成开发环境、源代码



清华大学出版社



# Android 9

## 编程通俗演义

牛 搞 著

清华大学出版社  
北京

## 内 容 简 介

本书严格参考 Android 9 官方开发文档的逻辑, 全面讲解 Android 开发中的各种技术, 章节内容循序渐进, 精心安排, 翔实全面, 且又通俗易懂, 既不是术语的罗列, 也不是不知所云的翻译。

本书分为 18 章, 内容包括配置 Android 9 开发环境、第一个 App、UI 资源与 Layout、各种 Layout 控件、代码操作控件、Activity 导航、Theme、Fragment、菜单、动画、自定义控件、RecyclerView、模仿 QQApp 界面、实现聊天界面、多线程、网络通信、异步调用库 RxJava、实现聊天功能等。

本书适合 Android 编程初学者、Android 应用开发人员, 也适合高等院校和培训学校相关专业的师生教学参考。

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

Android 9 编程通俗演义 / 牛犒著. —北京: 清华大学出版社, 2019

ISBN 978-7-302-52393-2

I. ①A… II. ①牛… 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 (2019) 第 038774 号

责任编辑: 夏毓彦

封面设计: 王 翔

责任校对: 闫秀华

责任印制: 刘海龙

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](mailto:c-service@tup.tsinghua.edu.cn)

质量反馈: 010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

印 装 者: 三河市龙大印装有限公司

经 销: 全国新华书店

开 本: 190mm×260mm 印 张: 28 字 数: 717 千字

版 次: 2019 年 4 月第 1 版 印 次: 2019 年 4 月第 1 次印刷

定 价: 89.00 元

---

产品编号: 082568-01

# 前言

自从 iOS 横空出世，移动应用开发持续火爆，人才需求量节节攀升，开发人员的薪资也勇攀高峰。但是，随着一批跨平台移动开发框架（如基于 JavaScript 的 PhoneGap、React Native，基于 .Net 的 Xarmain 等）的出现，企业对 iOS 与 Android 原生开发的需求量下降，其实大家在招聘网站上就可以感受到相关职位的减少。然而，所谓的跨平台移动开发其实是个大坑！原因很简单：没有一个操作系统愿意与其他的系统兼容、统一。比如 Android 与 iOS，即使它们在不停地互相学习，功能越来越相似，但是它们的开发语言、SDK、API 等不论在哪个层面都绝不兼容。所以当使用跨平台框架开发同时兼容 iOS 和 Android 的 App 时，就会踩到很多坑。更悲催的是，一旦某个操作系统升级了，你使用的框架可能马上会出现兼容性问题，你可以等待框架开发者把这个问题修正，但不知何年何月，实在等不了，你只能自己修正问题，于是你需要对这个框架的底层很熟悉，并且还要同时熟悉 iOS 与 Android 的原生开发，也就是说，你买了一个复杂的工具，你需要用它做两样不同的产品，你既需要学习如何使用这个工具，还要学习这两个产品的制作流程，还要学会修理和改进这个工具，有点恐怖啊！当然这可以锻炼你的能力，让你成为牛人中的牛人，但是这会拖延开发进度，你的老板能接受吗？

最近出现了很多反思这些框架的声音，而且已经有国外公司放弃 React Native 的事件发生，同时我在各技术群中感受到 Android 和 iOS 开发的招聘数量比过去两年有明显的增加，这都说明大家正在回归原生开发。当然我不是在完全否定跨平台开发框架，它们有它们的适应场景，比如一个电子商城 App，只提供商品展示、拍照、收藏、购物等常见功能，跨平台框架是完全能胜任的，但问题是，你依然需要熟悉原生开发，才能用好跨平台开发框架！本书讲的就是 Android 原生开发的故事，情节跌宕起伏，一波三折，相信你会喜欢。

## 作者心声

如何才能轻松学会一门开发技术？估计这个问题很多人都思考过，因为学技术或者说研究技术真的很难！（是不是说出了大家的心声？）大家应该都有感受：真正掌握一门开发技术其实需要很长时间。即使你是一只长期浸淫各种技术的“千年老妖”，给你一门陌生的技术，你还是会感受到入门的痛苦，你虽然了解各种模式、玩过各种知识，但是你就是无法在短时间内真正参透它。

为什么会这样？原因很简单：技术本来就是复杂的！但大家经常会听到有人说，某某开发

很简单，怎样怎样做就行了，随便学学就会了……这种鬼话，谁信谁上当！因为你真正动手使用它时，发现几乎一步一个坑！实际存在这样一个规律：仅学习如何使用一门技术而不真正搞懂其原理，你是不会用这门技术的，那个说简单的人，因为他已经完全掌握了这项技术，但他忘了他入门时所花费的脑力、时间以及经受的痛苦。

我说技术本来就复杂，可能有人不服，但我相信你仔细思考之后，就会同意这个观点。一项技术可能用一句话就能说清楚它的用途或概括它的原理，但当你真正运用它时，你就会发现里面隐藏了无数的细节，而且它还依赖很多其他的技术，你要一步步跨越这些沟沟坎坎，填平你的技术洼地，才能俘获它。

但是，学习技术难，把技术用文字讲明白更难！我到现在也没读到能让我轻轻松松看明白一门技术的书。尤其对于基础差的人来说，他们喜欢凑热闹买很多“技术名著”，但最终发现能看懂的内容寥寥无几！

为什么技术书籍都那么晦涩难懂呢？我想有三方面的主要原因：一是技术黑话（就是术语）太多；二是没有为读者补齐知识差距，作者只在自己的高度上讲啊讲，读者可能跟你隔着一层天；三是太多概括和抽象，把人整得云里雾里。

所以，我尝试改变技术书籍中的这些问题，写一本老少皆宜、童叟无欺、雅俗共赏的书，为大家讲明白一门复杂而庞大的技术：Android 开发。本书对读者的知识基础也仅要求会用 Java 语言，希望大家读起来轻轻松松。在书中作者尽量以通俗的语言讲述各种概念，每个技术点都以具体的案例引出，尽量不劳您费神思考。本书中还配了大量的截图，就是希望读者即使不动手操作，也能学个八九不离十。

本书的定位是 Android 开发入门，但是其中也涉及很多高级的技术内容和热门第三方库，比如多线程、RxJava、网络通信、Retrofit、前后台结合等，所以绝不仅仅适合没有基础的人。本书也适合那些未接触 Android 开发的其他领域的高手们，如果他们要快速了解 Android 开发的方方面面，这本书绝对是非常好的选择。

本书以 App 实例开发驱动，带领读者一步步完成一个仿 QQApp 的应用，保证让读者轻松搞懂每种技术的用途，并体验到每种技术的使用模式。本书紧跟 Android SDK 的更新脚步，所有例子都可在 Android 9 开发环境下编译和运行。

## 代码下载

本书中的示例都配有源码，地址分别是：

Android 起步+RecyclerView: <https://gitee.com/nnn/AndFirstStep/repository/archive/master.zip>

QQApp: <https://gitee.com/nnn/QQApp/repository/archive/master.zip>

QQApp 后台: <https://gitee.com/nnn/QQAppServer/repository/archive/master.zip>

## 联系作者

作者在 IT 开发领域工作近 20 年，由于对技术的爱好，一直没有脱离开发一线。近几年转向 IT 教学方向，一直致力于解决教学中的痛点与难点，总结 IT 学习的规律，并创建“被动式 IT 教学法”，能在降低教师工作量的同时有效提高学生学习效果。限于作者的水平，书中难免存在疏漏之处，还望各位读者批评指正。

最后，感谢各位朋友的大力帮助，此书的顺利面世离不开各位朋友的共同努力！

著者

2019 年 2 月

# 目 录

第 1 章 配置 Android 开发环境.....	1
1.1 下载 Android Studio .....	1
1.2 安装 Android Studio .....	2
1.3 配置 Android SDK.....	4
1.4 四原则.....	6
第 2 章 第一个 App .....	8
2.1 创建第一个 App .....	8
2.2 运行 App .....	12
2.2.1 在真实设备上调试 .....	13
2.2.2 配置虚拟机 .....	15
2.2.3 启动 App .....	18
2.2.4 x86 虚拟机加速 .....	19
2.2.5 App 的样子 .....	21
2.3 工程里面有什么 .....	22
第 3 章 UI 资源与 Layout .....	24
3.1 Layout.....	24
3.2 改动 Layout.....	27
3.2.1 添加图像资源 .....	30
3.2.2 显示自己的图像 .....	32
3.2.3 XML 小解 .....	35
3.2.4 Layout 源码解释 .....	36
3.3 排版姿方法之 ConstraintLayout .....	37
3.3.1 ConstraintLayout 的原理 .....	38
3.3.2 子控件在 ConstraintLayout 中居左或居右 .....	39
3.3.3 子控件在 ConstraintLayout 中横向居中 .....	40
3.3.4 子控件在 ConstraintLayout 中居中偏左 .....	41
3.3.5 子控件 A 在子控件 B 的上面.....	42
3.3.6 子控件 A 与子控件 B 左边对齐.....	43
3.3.7 设置子控件的宽和高 .....	44
3.3.8 子控件的宽和高保持一定比例 .....	45
3.4 排版方法之 RelativeLayout .....	48

3.4.1	把 ConstraintLayout 改为 RelativeLayout.....	49
3.4.2	左右对齐与居中 .....	51
3.4.3	充满整个父控件 .....	52
3.4.4	兄弟之间相对排 .....	53
3.4.5	dp 是什么 .....	55
3.4.6	使用 RelativeLayout 设计登录页面.....	56
3.5	让内容“滚” .....	63
3.5.1	添加 ScrollView 作为最外层容器 .....	63
3.5.2	改正在 ScrollView 下的排版 .....	66
3.6	添加新的 Layout 资源.....	70
<b>第 4 章</b>	<b>各种 Layout 控件 .....</b>	<b>72</b>
4.1	FrameLayout .....	72
4.2	LinearLayout .....	72
4.2.1	纵向 LinearLayout 中子控件横向居中.....	74
4.2.2	子控件均匀分布 .....	75
4.2.3	子控件按比例分布 .....	76
4.2.4	用 LinearLayout 实现登录界面.....	77
4.3	GridLayout .....	79
4.4	TableLayout.....	80
<b>第 5 章</b>	<b>代码操作控件.....</b>	<b>81</b>
5.1	在 Activity 中创建界面 .....	81
5.1.1	类 R .....	82
5.1.2	Activity 的父类.....	82
5.1.3	四大组件 .....	82
5.2	在代码中操作控件 .....	83
5.2.1	获取 View.....	84
5.2.2	响应 View 的事件.....	86
5.2.3	添加依赖库 .....	87
5.2.4	显示提示 .....	90
<b>第 6 章</b>	<b>Activity 导航.....</b>	<b>93</b>
6.1	创建注册页面 .....	93
6.2	启动注册页面 .....	94
6.3	设计注册页面 .....	98
6.4	响应注册按钮进行注册 .....	102
6.5	获取页面返回的数据 .....	103
6.5.1	避免常量重复出现 .....	105
6.5.2	日志输出 .....	106
6.5.3	将返回的数据设置到控件中 .....	107
6.6	Action Bar 上的返回图标 .....	109



6.6.1	原生 Action Bar 与 MaterialDesign Action Bar .....	109
6.6.2	登录页面显示返回图标 .....	111
6.6.3	注册页面显示返回图标 .....	112
<b>第 7 章</b>	<b>Theme .....</b>	<b>113</b>
<b>第 8 章</b>	<b>Fragment .....</b>	<b>115</b>
8.1	弄巧成拙的 Activity .....	115
8.2	使用 Fragment .....	117
8.3	改造登录页面 .....	120
8.3.1	添加 layout 文件 .....	120
8.3.2	改变 layout 文件的内容 .....	121
8.3.3	添加 Fragment 类 .....	122
8.3.4	将 Fragment 放到 Activity 中 .....	126
8.3.5	创建注册 Fragment .....	126
8.3.6	显示 RegisterFragment .....	128
8.3.7	通过 AppBar 控制页面导航 .....	129
8.3.8	实现 RegisterFragment 的逻辑 .....	129
8.3.9	LoginFragment 中读出用户名和密码 .....	131
8.3.10	Fragment 的生命周期 .....	132
8.3.11	Fragment 状态保存与恢复 .....	133
8.3.12	总结 .....	134
8.4	对话框 .....	138
8.4.1	创建子类 .....	138
8.4.2	显示对话框 .....	140
8.4.3	响应返回键 .....	141
8.4.4	取消输入控件的焦点 .....	142
<b>第 9 章</b>	<b>菜单 .....</b>	<b>143</b>
9.1	添加菜单资源 .....	144
9.2	重写 onCreateOptionsMenu() .....	147
9.3	嵌套菜单 .....	148
9.4	菜单项分组 .....	150
9.5	响应菜单项 .....	150
9.6	其他菜单类型 .....	152
<b>第 10 章</b>	<b>动画 .....</b>	<b>153</b>
10.1	动画原理 .....	153
10.2	三种动画 .....	154
10.3	View 动画 .....	155
10.3.1	绕着中心转 .....	156
10.3.2	不要反向转 .....	157

10.3.3	举一反三 .....	158
10.3.4	动画组 .....	158
10.4	属性动画 .....	159
10.4.1	旋转动画 .....	159
10.4.2	动画组 .....	160
10.5	动画资源 .....	164
10.6	Layout 动画 .....	167
10.6.1	向 Layout 控件添加子控件 .....	167
10.6.2	ViewGroup .....	168
10.6.3	设置排版动画 .....	169
10.7	转场动画 .....	171
10.7.1	使用默认转场动画 .....	171
10.7.2	自定义转场动画 .....	172
<b>第 11 章</b>	<b>自定义控件 .....</b>	<b>176</b>
11.1	创建一个 Custom View .....	177
11.2	Custom View 类 .....	179
11.2.1	构造方法 .....	179
11.2.2	onDraw()方法 .....	180
11.2.3	init()方法 .....	182
11.2.4	自定义属性 .....	184
11.2.5	作画 .....	186
11.3	创建圆形图像控件 .....	188
11.3.1	将 Drawable 转成 Bitmap .....	191
11.3.2	变换矩阵 .....	192
11.3.3	自定义属性的改动 .....	193
11.3.4	类的所有代码 .....	195
<b>第 12 章</b>	<b>RecyclerView .....</b>	<b>200</b>
12.1	基本用法 .....	200
12.2	显示多条简单数据 .....	201
12.2.1	添加新页面 .....	201
12.2.2	创建 Adapter 子类 .....	203
12.2.3	设置 RecyclerView .....	205
12.2.4	用集合保存数据 .....	206
12.3	让子控件复杂起来 .....	207
12.3.1	创建条目的 Layout 资源 .....	208
12.3.2	应用条目 Layout 资源 .....	210
12.3.3	明显区分每一行 .....	212
12.3.4	创建音乐信息类 .....	214
12.3.5	使用音乐信息类 .....	215
12.4	增删改 .....	217

12.4.1	增加一条 .....	217
12.4.2	其他操作 .....	219
12.5	局部刷新 .....	219
12.6	运行效率优化 .....	220
12.7	响应 item 选择 .....	221
12.8	显示不同类型的行 .....	223
12.8.1	添加新 Item 数据类 .....	224
12.8.2	添加 Item Layout .....	225
12.8.3	创建新的 ViewHolder 类 .....	226
12.8.4	区分不同的 View Type .....	227
<b>第 13 章</b>	<b>模仿 QQApp 界面 .....</b>	<b>230</b>
13.1	创建新的 Android 项目 .....	230
13.2	设计登录页面 .....	230
13.2.1	创建登录 Fragment .....	230
13.2.2	设计登录界面 .....	232
13.2.3	UI 代码 .....	233
13.2.4	显示登录历史 .....	236
13.2.5	设计历史菜单项 .....	240
13.2.6	实现显示历史的代码 .....	241
13.2.7	selector 资源 .....	243
13.2.8	layer_list 资源 .....	244
13.2.9	定制控件背景 .....	245
13.2.10	动画显示菜单 .....	246
13.2.11	让菜单消失 .....	247
13.2.12	响应选中菜单项 .....	248
13.3	QQ 主页面设计 .....	250
13.3.1	设置导航栏 .....	254
13.3.2	设置 Tab 栏 .....	255
13.3.3	改变 Tab Item 图标 .....	258
13.3.4	为 ViewPager 添加内容 .....	259
13.3.5	ViewPager 与 TabLayout 联动 .....	261
13.3.6	在 Tab Item 中显示图像 .....	263
13.3.7	禁止 ViewPager 滑动翻页 .....	266
13.3.8	创建“消息”页 .....	267
13.3.9	显示气泡菜单 .....	274
13.3.10	抽屉效果 .....	293
13.3.11	创建“联系人”页 .....	308
13.3.12	创建“动态”页 .....	328
13.3.13	实现搜索功能 .....	329

<b>第 14 章</b>	<b>实现聊天界面</b> .....	<b>339</b>
14.1	实现原理分析 .....	339
14.2	创建聊天 Activity .....	340
14.2.1	activity_chat.xml .....	340
14.2.2	类 ChatActivity .....	342
14.2.3	显示消息的 layout .....	344
14.3	启动 ChatActivity .....	346
14.4	模拟聊天 .....	347
<b>第 15 章</b>	<b>多线程</b> .....	<b>349</b>
15.1	线程与进程的概念 .....	349
15.2	创建线程 .....	350
15.3	创建线程的另一种方法 .....	352
15.4	多个线程操作同一个对象 .....	353
15.5	单线程中异步执行 .....	356
15.6	多线程间同步执行 .....	357
15.7	在其他线程中操作界面 .....	358
15.8	HandlerThread .....	360
15.9	线程的退出 .....	361
<b>第 16 章</b>	<b>网络通信</b> .....	<b>363</b>
16.1	网络基础知识 .....	363
16.1.1	IP 地址与域名 .....	363
16.1.2	TCP 与 UDP .....	364
16.1.3	HTTP 协议 .....	364
16.2	Android HTTP 通信 .....	365
16.3	使用“异步任务” .....	369
16.3.1	定义异步任务类 .....	369
16.3.2	使用异步任务类 .....	370
16.3.3	完善异步任务类 .....	371
16.3.4	异步任务的退出 .....	378
16.4	使用 OkHttp 进行网络通信 .....	380
16.4.1	使用 OkHttp 下载图像 .....	381
16.4.2	创建 Web 服务端 .....	383
16.4.3	使用 OkHttp 下载数据 .....	385
16.4.4	JSON 转对象 .....	387
16.4.5	使用 OkHttp 上传文件 .....	388
16.5	使用 Retrofit 进行网络通信 .....	391
16.5.1	加入 Retrofit 的依赖项 .....	391
16.5.2	用 Retrofit 下载文本 .....	392
16.5.3	用 Retrofit 下载图像 .....	393
16.5.4	用 Retrofit 上传图像 .....	394

第 17 章 异步调用库 RxJava .....	397
17.1 小试牛刀 .....	397
17.2 精简发送代码 .....	400
17.3 精简接收代码 .....	401
17.4 RxJava 与 Lamda .....	402
17.5 map 与 flatmap .....	404
17.6 并行 map .....	405
17.7 RxJava 与 Retrofit 合体 .....	406
17.8 RxJava Retrofit 合体并行执行 .....	407
第 18 章 实现聊天功能 .....	409
18.1 改进登录功能 .....	411
18.1.1 制定统一的数据返回结构 .....	411
18.1.2 向 ChatService 中添加方法 .....	413
18.1.3 登录请求 .....	414
18.1.4 保存自己的信息 .....	417
18.1.5 防止按钮重复点击 .....	418
18.1.6 显示进度条 .....	418
18.2 获取联系人 .....	421
18.2.1 修改 Retrofit 接口 .....	422
18.2.2 RxJava 定时器 .....	422
18.2.3 获取并显示联系人 .....	423
18.2.4 出错重试 .....	425
18.2.5 停止网络连接 .....	425
18.3 发出聊天消息 .....	427
18.3.1 定义承载消息的类 .....	427
18.3.2 在接口中添加方法 .....	428
18.3.3 在 ChatActivity 中初始化 Retrofit .....	429
18.3.4 上传消息 .....	429
18.3.5 失败重传 .....	431
18.4 获取聊天消息 .....	431
18.4.1 为 ChatService 增加方法 .....	431
18.4.2 发出请求 .....	431

# 第 1 章

## 配置Android开发环境

Android 开发有两种 IDE（集成开发环境）可以使用，一是 Android Studio，二是 ADT+Eclipse。ADT+Eclipse 这种方式在写此文之前 Google 已经宣布不再更新了，所以其实只有一种选择：Android Studio！

使用 Android Studio 编写 Android App，需要经过以下几步：

- 下载 Android Studio。
- 安装 Android Studio。
- 配置 Android SDK。

本文的操作都是在 Windows 下，其余操作系统上也差不多，只要你熟悉那个系统，参照这个教程也可以配置成功。

### 1.1 下载 Android Studio

我们天朝上国，有长城抵御外夷，外夷指的是 Google、FaceBook 等尚未“开化”的公司，长城指的是那道“防火墙”。在此文写作之时，墙外 Google 家的网站依然上不去，但 Google 正在将部分业务安排回归中国，所以国内的 Android 官网镜像似乎经常能成功访问，所以你首先可以试试登录官网，地址是：<https://developer.android.google.cn/studio/>，进入后可看到如图 1.1.1 所示界面。



图 1.1.1

如果你是 64 位的 Windows 操作系统，你可以直接点（注意，为了形象起见，本书使用“点”代表鼠标单击操作）上面的大绿按钮下载安装包，如果不是，你就需要点“DOWNLOAD OPTIONS”，进入另一个页面选择合适的安装包。

下载完成后，下一步就是安装 Android Studio 了，欲知后事如何，请看下节分解。

## 1.2 安装 Android Studio

找到下载的文件，这就是安装文件，双击运行之。启动时间可能比较长，请耐心等待，启动后出现图 1.2.1 所示的界面。

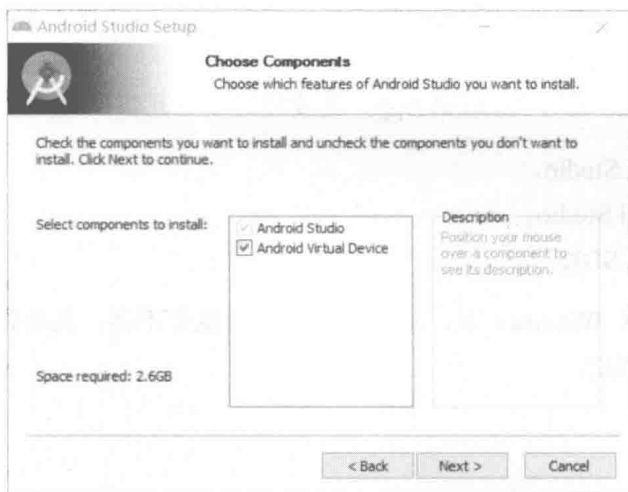


图 1.2.1

什么都不用改动，直接点“Next（下一步）”，进入图 1.2.2 所示页面，在这里选择安装位置。注意安装到的路径中不要有中文或全角字符。如果你的 C 盘剩余空间小于 20GB，就应该选择安装到其他盘了。如果选其他位置安装，点“Browser（浏览）”按钮，默认位置就不错，我就不改了，点“Next”，如图 1.2.3 所示。

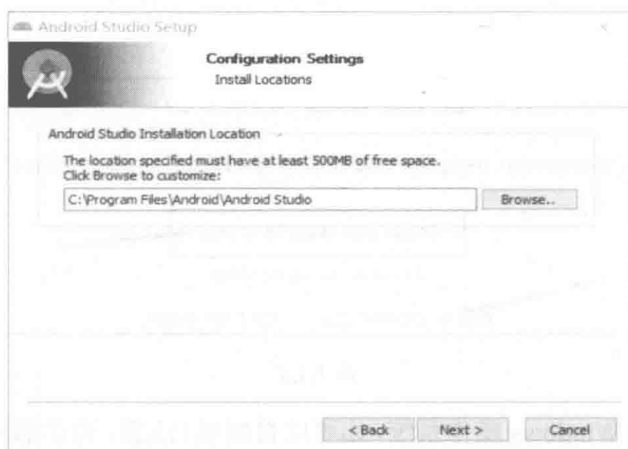


图 1.2.2

进入建立快捷方式的页面，这里不需要动，直接点“Next”，如图 1.2.3 所示。

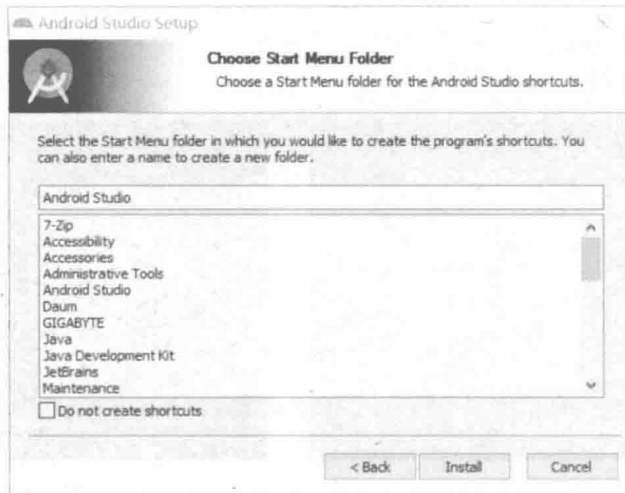


图 1.2.3

进入安装页面，等待安装完成，完成后点“Next”，结果如图 1.2.4 所示。



图 1.2.4

安装完成，如图 1.2.5 所示。点“Finish（完成）”。因为“Start Android Studio（启动 Android Studio）”被选中了，所以点“Finish”后，Android Studio 会运行。

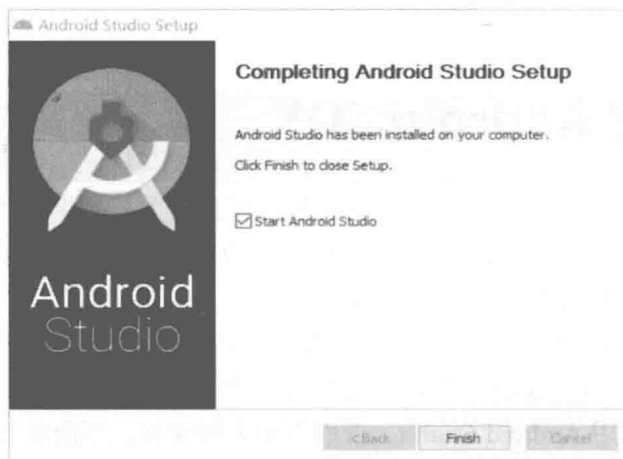


图 1.2.5



也可以去开始菜单找到 Android Studio 的快捷菜单启动它，如图 1.2.6 所示。Android Studio 启动中，如图 1.2.7 所示。

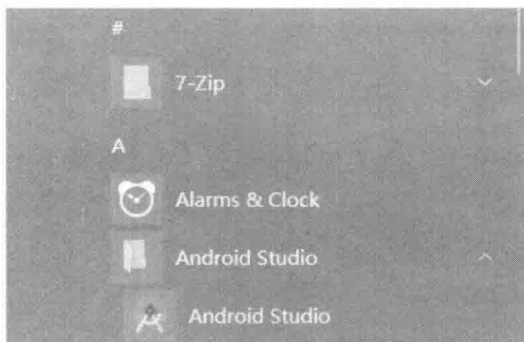


图 1.2.6



图 1.2.7

启动后可能会出现这样的界面，如图 1.2.8 所示。

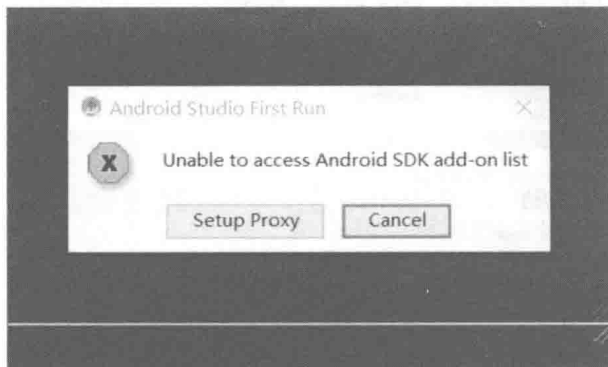


图 1.2.8

“Unable to access Android SDK add.on list”的意思是无法访问 Android SDK 附件列表，这是告诉你需要安装配置 Android SDK。如何搞定 Android SDK 呢？下节分解。

## 1.3 配置 Android SDK

什么是 SDK? SDK 是软件开发工具包的意思，你要基于某个语言开发软件，需要使用一些类、调用一些方法，这些类和方法封装了一些基础功能和操作系统的功能，以这种语言的方式提供给你，这些类、方法等就组成了 SDK。

JDK 是 Java SDK 的意思，就是用 Java 开发程序时所使用的 SDK，要开发 Android 程序，当然得用 Android SDK。而 Android Studio 包中并不带有 Android SDK，需要单独安装，当然 Android SDK 也应该利用 Android Studio，否则弄起来很麻烦。下面就开始安装它，步骤基本是这样的：

在图 1.2.8 中，点“Cancel”，进入如图 1.3.1 所示的页面。