

# 第一行代码 Android 第2版 (图灵原创)

作者：郭霖

## 版权信息

书名：第一行代码——Android（第2版）

作者：郭霖

ISBN：978-7-115-43978-9

本书由北京图灵文化发展有限公司发行数字版。版权所有，侵权必究。

---

您购买的图灵电子书仅供您个人使用，未经授权，不得以任何方式复制和传播本书内容。

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。

---

图灵社区会员 人民邮电出版社（zhanghaichuan@ptpress.com.cn）专享 尊重版权

[前言](#)

[第2版的变化](#)

[读者对象](#)

[本书内容](#)

[源码下载](#)

[致谢](#)

[第1章 开始启程——你的第一行Android代码](#)

[1.1 了解全貌——Android王国简介](#)

[1.1.1 Android系统架构](#)

[1.1.2 Android已发布的版本](#)

[1.1.3 Android应用开发特色](#)

[1.2 手把手带你搭建开发环境](#)

[1.2.1 准备所需要的工具](#)

[1.2.2 搭建开发环境](#)

[1.3 创建你的第一个Android项目](#)

[1.3.1 创建HelloWorld项目](#)

[1.3.2 启动模拟器](#)

[1.3.3 运行HelloWorld](#)

[1.3.4 分析你的第一个Android程序](#)

[1.3.5 详解项目中的资源](#)

[1.3.6 详解build.gradle文件](#)

[1.4 前行必备——掌握日志工具的使用](#)

[1.4.1 使用Android的日志工具Log](#)

[1.4.2 为什么使用Log而不使用System.out](#)

[1.5 小结与点评](#)

[第2章 先从看得到的入手——探究活动](#)

[2.1 活动是什么](#)

[2.2 活动的基本用法](#)

[2.2.1 手动创建活动](#)

[2.2.2 创建和加载布局](#)

[2.2.3 在AndroidManifest文件中注册](#)

[2.2.4 在活动中使用Toast](#)

[2.2.5 在活动中使用Menu](#)

[2.2.6 销毁一个活动](#)

## [2.3 使用Intent在活动之间穿梭](#)

### [2.3.1 使用显式Intent](#)

### [2.3.2 使用隐式Intent](#)

### [2.3.3 更多隐式Intent的用法](#)

### [2.3.4 向下一个活动传递数据](#)

### [2.3.5 返回数据给上一个活动](#)

## [2.4 活动的生命周期](#)

### [2.4.1 返回栈](#)

### [2.4.2 活动状态](#)

### [2.4.3 活动的生存期](#)

### [2.4.4 体验活动的生命周期](#)

### [2.4.5 活动被回收了怎么办](#)

## [2.5 活动的启动模式](#)

### [2.5.1 standard](#)

### [2.5.2 singleTop](#)

### [2.5.3 singleTask](#)

### [2.5.4 singleInstance](#)

## [2.6 活动的最佳实践](#)

### [2.6.1 知晓当前是在哪一个活动](#)

### [2.6.2 随时随地退出程序](#)

### [2.6.3 启动活动的最佳写法](#)

## [2.7 小结与点评](#)

## [第3章 软件也要拼脸蛋——UI开发的点点滴滴](#)

### [3.1 如何编写程序界面](#)

### [3.2 常用控件的使用方法](#)

#### [3.2.1 TextView](#)

#### [3.2.2 Button](#)

#### [3.2.3 EditText](#)

#### [3.2.4 ImageView](#)

#### [3.2.5 ProgressBar](#)

#### [3.2.6 AlertDialog](#)

#### [3.2.7 ProgressDialog](#)

### [3.3 详解4种基本布局](#)

#### [3.3.1 线性布局](#)

#### [3.3.2 相对布局](#)

#### [3.3.3 帧布局](#)

#### [3.3.4 百分比布局](#)

### [3.4 系统控件不够用？创建自定义控件](#)

#### [3.4.1 引入布局](#)

#### [3.4.2 创建自定义控件](#)

### [3.5 最常用和最难用的控件——ListView](#)

#### [3.5.1 ListView的简单用法](#)

#### [3.5.2 定制ListView的界面](#)

#### [3.5.3 提升ListView的运行效率](#)

#### [3.5.4 ListView的点击事件](#)

### [3.6 更强大的滚动控件——RecyclerView](#)

#### [3.6.1 RecyclerView的基本用法](#)

[3.6.2 实现横向滚动和瀑布流布局](#)

[3.6.3 RecyclerView的点击事件](#)

[3.7 编写界面的最佳实践](#)

[3.7.1 制作Nine-Patch图片](#)

[3.7.2 编写精美的聊天界面](#)

[3.8 小结与点评](#)

[第4章 手机平板要兼顾——探究碎片](#)

[4.1 碎片是什么](#)

[4.2 碎片的使用方式](#)

[4.2.1 碎片的简单用法](#)

[4.2.2 动态添加碎片](#)

[4.2.3 在碎片中模拟返回栈](#)

[4.2.4 碎片和活动之间进行通信](#)

[4.3 碎片的生命周期](#)

[4.3.1 碎片的状态和回调](#)

[4.3.2 体验碎片的生命周期](#)

[4.4 动态加载布局的技巧](#)

[4.4.1 使用限定符](#)

[4.4.2 使用最小宽度限定符](#)

[4.5 碎片的最佳实践——一个简易版的新闻应用](#)

[4.6 小结与点评](#)

[第5章 全局大喇叭——详解广播机制](#)

[5.1 广播机制简介](#)

[5.2 接收系统广播](#)

[5.2.1 动态注册监听网络变化](#)

[5.2.2 静态注册实现开机启动](#)

[5.3 发送自定义广播](#)

[5.3.1 发送标准广播](#)

[5.3.2 发送有序广播](#)

[5.4 使用本地广播](#)

[5.5 广播的最佳实践——实现强制下线功能](#)

[5.6 Git时间——初识版本控制工具](#)

[5.6.1 安装Git](#)

[5.6.2 创建代码仓库](#)

[5.6.3 提交本地代码](#)

[5.7 小结与点评](#)

[第6章 数据存储全方案——详解持久化技术](#)

[6.1 持久化技术简介](#)

[6.2 文件存储](#)

[6.2.1 将数据存储到文件中](#)

[6.2.2 从文件中读取数据](#)

[6.3 SharedPreferences存储](#)

[6.3.1 将数据存储到SharedPreferences中](#)

[6.3.2 从SharedPreferences中读取数据](#)

[6.3.3 实现记住密码功能](#)

[6.4 SQLite数据库存储](#)

[6.4.1 创建数据库](#)

- [6.4.2 升级数据库](#)
- [6.4.3 添加数据](#)
- [6.4.4 更新数据](#)
- [6.4.5 删除数据](#)
- [6.4.6 查询数据](#)
- [6.4.7 使用SQL操作数据库](#)
- [6.5 使用LitePal操作数据库](#)
  - [6.5.1 LitePal简介](#)
  - [6.5.2 配置LitePal](#)
  - [6.5.3 创建和升级数据库](#)
  - [6.5.4 使用LitePal添加数据](#)
  - [6.5.5 使用LitePal更新数据](#)
  - [6.5.6 使用LitePal删除数据](#)
  - [6.5.7 使用LitePal查询数据](#)
- [6.6 小结与点评](#)
- [第7章 跨程序共享数据——探究内容提供者](#)
  - [7.1 内容提供者简介](#)
  - [7.2 运行时权限](#)
    - [7.2.1 Android权限机制详解](#)
    - [7.2.2 在程序运行时申请权限](#)
  - [7.3 访问其他程序中的数据](#)
    - [7.3.1 ContentResolver的基本用法](#)
    - [7.3.2 读取系统联系人](#)
  - [7.4 创建自己的内容提供者](#)
    - [7.4.1 创建内容提供器的步骤](#)
    - [7.4.2 实现跨程序数据共享](#)
  - [7.5 Git时间——版本控制工具进阶](#)
    - [7.5.1 忽略文件](#)
    - [7.5.2 查看修改内容](#)
    - [7.5.3 撤销未提交的修改](#)
    - [7.5.4 查看提交记录](#)
  - [7.6 小结与点评](#)
- [第8章 丰富你的程序——运用手机多媒体](#)
  - [8.1 将程序运行到手机上](#)
  - [8.2 使用通知](#)
    - [8.2.1 通知的基本用法](#)
    - [8.2.2 通知的进阶技巧](#)
    - [8.2.3 通知的高级功能](#)
  - [8.3 调用摄像头和相册](#)
    - [8.3.1 调用摄像头拍照](#)
    - [8.3.2 从相册中选择照片](#)
  - [8.4 播放多媒体文件](#)
    - [8.4.1 播放音频](#)
    - [8.4.2 播放视频](#)
  - [8.5 小结与点评](#)
- [第9章 看看精彩的世界——使用网络技术](#)
  - [9.1 WebView的用法](#)

## [9.2 使用HTTP协议访问网络](#)

### [9.2.1 使用URLConnection](#)

#### [9.2.2 使用OkHttp](#)

## [9.3 解析XML格式数据](#)

### [9.3.1 Pull解析方式](#)

#### [9.3.2 SAX解析方式](#)

## [9.4 解析JSON格式数据](#)

### [9.4.1 使用JSONObject](#)

#### [9.4.2 使用GSON](#)

## [9.5 网络编程的最佳实践](#)

## [9.6 小结与点评](#)

## [第10章 后台默默的劳动者——探究服务](#)

### [10.1 服务是什么](#)

## [10.2 Android多线程编程](#)

### [10.2.1 线程的基本用法](#)

#### [10.2.2 在子线程中更新UI](#)

### [10.2.3 解析异步消息处理机制](#)

#### [10.2.4 使用AsyncTask](#)

## [10.3 服务的基本用法](#)

### [10.3.1 定义一个服务](#)

#### [10.3.2 启动和停止服务](#)

#### [10.3.3 活动和服务进行通信](#)

## [10.4 服务的生命周期](#)

## [10.5 服务的更多技巧](#)

### [10.5.1 使用前台服务](#)

#### [10.5.2 使用IntentService](#)

## [10.6 服务的最佳实践——完整版的下载示例](#)

## [10.7 小结与点评](#)

## [第11章 Android特色开发——基于位置的服务](#)

### [11.1 基于位置的服务简介](#)

## [11.2 申请API Key](#)

## [11.3 使用百度定位](#)

### [11.3.1 准备LBS SDK](#)

#### [11.3.2 确定自己位置的经纬度](#)

#### [11.3.3 选择定位模式](#)

#### [11.3.4 看得懂的位置信息](#)

## [11.4 使用百度地图](#)

### [11.4.1 让地图显示出来](#)

#### [11.4.2 移动到我的位置](#)

#### [11.4.3 让“我”显示在地图上](#)

## [11.5 Git时间——版本控制工具的高级用法](#)

### [11.5.1 分支的用法](#)

#### [11.5.2 与远程版本库协作](#)

## [11.6 小结与点评](#)

## [第12章 最佳的UI体验——Material Design实战](#)

### [12.1 什么是Material Design](#)

## [12.2 Toolbar](#)

- [12.3 滑动菜单](#)
- [12.3.1 DrawerLayout](#)
- [12.3.2 NavigationView](#)
- [12.4 悬浮按钮和可交互提示](#)
- [12.4.1 FloatingActionButton](#)
- [12.4.2 Snackbar](#)
- [12.4.3 CoordinatorLayout](#)
- [12.5 卡片式布局](#)
- [12.5.1 CardView](#)
- [12.5.2 AppBarLayout](#)
- [12.6 下拉刷新](#)
- [12.7 可折叠式标题栏](#)
- [12.7.1 CollapsingToolbarLayout](#)
- [12.7.2 充分利用系统状态栏空间](#)
- [12.8 小结与点评](#)
- [第 13 章 继续进阶——你还应该掌握的高级技巧](#)
- [13.1 全局获取Context的技巧](#)
- [13.2 使用Intent传递对象](#)
- [13.2.1 Serializable方式](#)
- [13.2.2 Parcelable方式](#)
- [13.3 定制自己的日志工具](#)
- [13.4 调试Android程序](#)
- [13.5 创建定时任务](#)
- [13.5.1 Alarm机制](#)
- [13.5.2 Doze模式](#)
- [13.6 多窗口模式编程](#)
- [13.6.1 进入多窗口模式](#)
- [13.6.2 多窗口模式下的生命周期](#)
- [13.6.3 禁用多窗口模式](#)
- [13.7 Lambda表达式](#)
- [13.8 总结](#)
- [第 14 章 进入实战——开发酷欧天气](#)
- [14.1 功能需求及技术可行性分析](#)
- [14.2 Git时间——将代码托管到GitHub上](#)
- [14.3 创建数据库和表](#)
- [14.4 遍历全国省市县数据](#)
- [14.5 显示天气信息](#)
- [14.5.1 定义GSON实体类](#)
- [14.5.2 编写天气界面](#)
- [14.5.3 将天气显示到界面上](#)
- [14.5.4 获取必应每日一图](#)
- [14.6 手动更新天气和切换城市](#)
- [14.6.1 手动更新天气](#)
- [14.6.2 切换城市](#)
- [14.7 后台自动更新天气](#)
- [14.8 修改图标和名称](#)
- [14.9 你还可以做的事情](#)

## [第15章 最后一步——将应用发布到360应用商店](#)

### [15.1 生成正式签名的APK文件](#)

#### [15.1.1 使用Android Studio生成](#)

#### [15.1.2 使用Gradle生成](#)

#### [15.1.3 生成多渠道APK文件](#)

### [15.2 申请360开发者账号](#)

### [15.3 发布应用程序](#)

### [15.4 嵌入广告进行盈利](#)

#### [15.4.1 注册腾讯广告联盟账号](#)

#### [15.4.2 新建媒体和广告位](#)

#### [15.4.3 接入广告SDK](#)

#### [15.4.4 重新发布应用程序](#)

### [15.5 结束语](#)

## 前言

虽然我从事Android开发工作已经很多年了，但是之前从来没有想过自己要去写一本Android技术相关的书。在我看来，写一本书可以算是一个很庞大的工程，写一本好书的难度并不亚于开发一款好的应用程序。

由于我长期坚持在CSDN上发表技术博文，因而得到了大量网友的认可，也积累了一定的名气。很荣幸的是，人民邮电出版社图灵公司的前副总编辑陈冰老师联系上了我，希望我可以写一本关于Android开发技术的书，这着实让我受宠若惊。

在写本书第1版的时候，我可以说是费了相当大的心思。写书和写博客最大的区别在于，书的内容不能像博客那样散乱，不能想到哪里写到哪里，而是一定要系统化，要循序渐进，基本上在写第1章的时候就应该把全书的内容都确定下来了。

令我非常欣慰的是，本书的第1版在推出之后获得了广大读者的强烈认可，在短短两年时间内，已经成为了国内最畅销的Android技术书。各大书店、图书馆都能看到《第一行代码》的身影，许多学校和培训机构也纷纷将《第一行代码》选为Android课程的教材。

不过，在科技高速发展的今天，各种技术的发展都是日新月异的。在两年的时间里，Android操作系统经历了5.0、6.0、7.0的飞速升级。不可否认的是，本书第1版中的不少知识都已经过时，而且这两年间出现的很多新知识，第1版中也没有涵盖。因此，这让我坚定了写作本书第2版的想法。

刚开始写的时候，我以为只是小修小补，但事实上并没有我想象得那么轻松。除了介绍新知识点以外，书中之前的所有项目都需要重新编写和测试，以保证代码在新老系统上的兼容性。另外，由于Android从5.0系统开始，UI风格变化很大，因此第1版中所有的截图都需要更新。毫不夸张地说，我几乎重写了整本书。

而现在，你手中捧着的正是全新版的《第一行代码》，同时这也是国内第一本基于Android 7.0系统写作的技术书。我真诚地希望你可以用心去阅读这本书，因为每多掌握一份知识，你就会多一份喜悦。Enjoy it!

## 第2版的变化

由于第2版修改内容繁多，因此这里我只列举出最主要的变化。首先是开发工具上的改变，本书第1版使用的开发工具是Eclipse，而第2版使用了目前最新的Android Studio 2.2版本。另外，本书第1版是基于Android 4.x系统的，而第2版是基于Android 7.0系统的，其中囊括了新系统中的诸多知识点，包括Android 5.0系统中引入的Material Design、Android 6.0系统中引入的运行时限和Doze模式、Android 7.0系统中引入的多窗口模式等。

除此之外，第2版还加入了Gradle、RecyclerView、百分比布局、OkHttp、Lambda表达式等全新知识点的讲解，内容将前所未有的充实。

## 读者对象

本书内容通俗易懂，由浅入深，既适合初学者阅读，也同样适合专业人员。学习本书内容之前，你并不需要有任何的Android基础，但是你需要有一定的Java基础，因为Android开发都是使用Java语言的，而本书并不会去专门介绍Java方面的知识。

阅读本书时，你可以根据自身的情况来决定如何阅读。如果你是初学者的话，建议你从第1章开始循序渐进地阅读，这样理解起来就不会感到吃力。而如果你已经有了一定的Android基础，那么就可以选择某些你感兴趣的章节进行跳跃式的阅读。但请记住，很多章最后的最佳实践部分一定是你不想错过的。

## 本书内容

正如前面所说，本书的内容是非常系统化的，不仅全面介绍了那些你必须掌握的知识，而且保证了各章的难度都是梯度式上升的。全书一共分为15章，涵盖了四大组件、UI、碎片、数据存储、多媒体、网络、定位服务等方方面面的知识。为了让你在学完所有内容之后还可以有综合运用能力，本书的尾声部分还会带你一起开发一个天气预报程序，并教会你如何将程序发布到应用商店，以及如何在程序中嵌入广告盈利。

除此之外，本书的第5章、第7章、第11章、第14章中都穿插了对Git的讲解，如果想要掌握它的用法，这几章的内容是绝对不能错过的。

本书中各个章节的内容都相对比较独立，因此除了可以循序渐进地学习之外，你还可以把它当成一本参考手册，随时查阅。

## 源码下载

首先，我建议你在学习本书的时候将所有项目的源码都亲手敲一遍，因为只有这样才能加深你对代码的理解。不过为了便于你的学习，我还是提供了书中所有项目的源码，请仅在需要的时候再去参考（如下载项目中的图片资源）。切勿直接将源码复制粘贴当成自己的东西了，只有亲手敲过的代码才真正是你自己的。

源码下载地址：<https://github.com/guolindev/booksource>。

## 致谢

在这近一年的时间里，我又完成了一项浩大的工程。和写作本书第1版时的感觉类似，当全书完稿之后，回顾整本书，我仍然不敢相信这所有的内容竟然是我一字一字地敲出来的。

如今这已经是我写的第二本书了，和写第一本书时的情况不同，现在我有更广的人脉和资源，有了更多的人愿意帮助和支持我来完成一本更好的技术书。因此，我要在这里对很多人表示感谢。

首先我要感谢我的父母，感谢你们将我抚养长大，感谢你们的付出，让我从小不用为生计、上学而发愁，可以一直做我自己想做的事情，也感谢你们指引我走上了技术这条路。

其次我要感谢我的妻子，感谢你每天为我准备好一日三餐，感谢你对我永远的包容，不管是平日的加班还是没日没夜的写书，你都一直默默地理解和支持我。

我还非常感谢本书第1版的编辑陈冰老师，如果没有你当初在CSDN上找到我，并邀请我写书，就不会有现在的《第一行代码》。另外，你也是当时唯一一个坚信这本书一定会大卖的人，甚至连我自己当时都没有如此的眼光。

我也非常感谢本书第2版的编辑张霞，你全程负责了第2版的出版工作，并且完成得非常出色。你对文字的把控能力让我敬佩，感谢你对书中每一章节的尽心审阅，才能让这本书更趋近于完美。

另外我还要特别感谢一部分人，你们在对本书的内容建议、勘误检查、代码纠错，甚至是对我个人的支持等方面都作出了卓越的贡献。有了你们的帮助，才会有这样一本更加出色的书呈现在所有人面前，这本书上也理应有你们的名字（按姓氏拼音排序，排名不分先后）：

陈建林 陈俊杰 陈雷 陈龙 陈琪 陈秀相 陈逸鸣 代云蛟 董霖轩 段郭森

高鹤泉 高太稳 关爱民 何以诚 胡恩泽 黄楠 赖帆 李济洲 李建友 李沛明

李潭 李永鹏 李志云 林火荣 刘萌 刘明渊 刘治国 陆德俊 罗亚超 吕国鑫

马文杰 覃文斌 孙建飞 王柏强 王光东 王杰 王龙 王路路 王鹏 王荣宗

王善昌 韦振南 吴波 吴宏权 吴绍志 徐阳 轩仲宽 杨辉 易静杰 查童

张鸿洋 张英祥 赵翠龙 赵庆元 赵迎超 郑传书 郑敏馨 庄育锋 周苏 朱海丰

## 第1章 开始启程——你的第一行Android代码

欢迎你来到Android世界！Android系统是目前世界上市场占有率最高的移动操作系统，不管你在哪里，都可以看到Android手机几乎无处不在。今天的Android世界可谓欣欣向荣，可是你知道它的过去是什么样的吗？我们一起来看一看它的发展史吧。

2003年10月，Andy Rubin等人一起创办了Android公司。2005年8月谷歌收购了这家仅仅成立了22个月的公司，并让Andy Rubin继续负责Android项目。在经过了数年的研发之后，谷歌终于在2008年推出了Android系统的第一个版本。但自那之后，Android的发展就一直受到重重阻挠。乔布斯自始至终认为Android是一个抄袭iPhone的产品，里面剽窃了诸多iPhone的创意，并声称一定要毁掉Android。而本身就是基于Linux开发的Android操作系统，在2010年被Linux团队从Linux内核主线中除名。又由于Android中的应用程序都是使用Java开发的，甲骨文则针对Android侵犯Java知识产权一事对谷歌提起了诉讼……

可是，似乎再多的困难也阻挡不了Android快速前进的步伐。由于谷歌的开放政策，任何手机厂商和个人都能免费获取到Android操作系统的源码，并且可以自由地使用和定制。三星、HTC、摩托罗拉、索爱等都推出了各自系列的Android手机，Android市场上百花齐放。仅仅推出两年后，Android就超过了已经霸占市场逾十年的诺基亚Symbian，成为了全球第一大智能手机操作系统，并且每天都还会有数百万台新的Android设备被激活。而近几年，国内的手机厂商也是大放异彩，小米、华为、魅族等新兴品牌都推出了相当不错的Android手机，并且也获得了市场的广泛认可，目前Android已经占据了全球智能手机操作系统70%以上的份额。

说了这些，想必你已经体会到Android系统炙手可热的程度，并且迫不及待地想要加入到Android开发者的行列当中了吧。试想一下，十个人中有七个人的手机都可以运行你编写的应用程序，还有什么能比这个更诱人的呢？那么从今天起，我就带你踏上学习Android的旅途，一步步地引导你成为一名出色的Android开发者。

好了，现在我们就来一起初窥一下Android世界吧。

### 1.1 了解全貌——Android王国简介

Android从面世以来到现在已经发布了二十几个版本了。在这几年的发展过程中，谷歌为Android王国建立了一个完整的生态系统。手机厂商、开发者、用户之间相互依存，共同推进着Android的蓬勃发展。开发者在其中扮演着不可或缺的角色，因为如果没有开发者来制作丰富的应用程序，那么不管多么优秀的操作系统，也是难以得到大众用户喜爱的，相信没有多少人能够忍受没有QQ、微信的手机吧。而且，谷歌推出的Google Play更是给开发者带来了大量的机遇，只要你能制作出优秀的产品，在Google Play上获得了用户的认可，你就完全可以得到不错的经济回报，从而成为一名独立开发者，甚至是成功创业！

那我们现在就以一个开发者的角度，去了解一下这个操作系统吧。纯理论型的东西也比较无聊，怕你看睡着了，因此我只挑重点介绍，这些东西跟你以后的开发工作都是息息相关的。

#### 1.1.1 Android系统架构

为了让你能够更好地理解Android系统是怎么工作的，我们先来看一下它的系统架构。Android大致可以分为四层架构：Linux内核层、系统运行库层、应用框架层和应用层。

##### 1. Linux内核层

Android系统是基于Linux内核的，这一层为Android设备的各种硬件提供了底层的驱动，如显示驱动、音频驱动、照相机驱动、蓝牙驱动、Wi-Fi驱动、电源管理等。

##### 2. 系统运行库层

这一层通过一些C/C++库来为Android系统提供了主要的特性支持。如SQLite库提供了数据库的支持，OpenGL|ES库提供了3D绘图的支持，Webkit库提供了浏览器内核的支持等。

同样在这一层还有Android运行时库，它主要提供了一些核心库，能够允许开发者使用Java语言来编写Android应用。另外，Android运行时库中还包含了Dalvik虚拟机（5.0系统之后改为ART运行环境），它使得每一个Android应用都能运行在独立的进程当中，并且拥有一个自己的Dalvik虚拟机实例。相较于Java虚拟机，Dalvik是专门为移动设备定制的，它针对手机内存、CPU性能有限等情况做了优化处理。

##### 3. 应用框架层

这一层主要提供了构建应用程序时可能用到的各种API，Android自带的一些核心应用就是使用这些API完成的，开发者也可以通过使用这些API来构建自己的应用程序。

##### 4. 应用层



所有安装在手机上的应用程序都是属于这一层的，比如系统自带的联系人、短信等程序，或者是你从Google Play上下载的小游戏，当然还包括你自己开发的程序。

结合图1.1你会理解得更加深刻，图片源自维基百科。



图 1.1 Android系统架构

### 1.1.2 Android已发布的版本

2008年9月，谷歌正式发布了Android 1.0系统，这也是Android系统最早的版本。随后的几年，谷歌以惊人的速度不断地更新Android系统，2.1、2.2、2.3系统的推出使Android占据了大量的市场。2011年2月，谷歌发布了Android 3.0系统，这个系统版本是专门为平板电脑设计的，但也是Android为数不多的比较失败的版本，推出之后一直不见什么起色，市场份额也少得可怜。不过很快，在同年的10月，谷歌又发布了Android 4.0系统，这个版本不再对手机和平板进行差异化区分，既可以应用在手机上，也可以应用在平板上。2014年Google I/O大会上，谷歌推出了号称史上版本改动最大的Android 5.0系统，其中使用ART运行环境替代了Dalvik虚拟机，大大提升了应用的运行速度，还提出了Material Design的概念来优化应用的界面设计。除此之外，还推出了Android Wear、Android Auto、Android TV系统，从而进军可穿戴设备、汽车、电视等全新领域。之后Android的更新速度更加迅速，2015年Google I/O大会上推出了Android 6.0系统，加入运行时权限功能，2016年Google I/O大会上推出了Android 7.0系统，加入多窗口模式功能，这也是目前最新的Android系统版本。

下表列出了目前主要的Android系统版本及其详细信息。你看到这张表格时，数据可能已经发生了变化，查看最新的数据可以访问<http://developer.android.google.cn/about/dashboards/>。

版本号	系统代号	API	市场占有率
2.2	Froyo	8	0.1%
2.3.3 – 2.3.7	Gingerbread	10	1.5%
4.0.3 – 4.0.4	Ice Cream Sandwich	15	1.3%
4.1.x	Jelly Bean	16	5.6%
4.2.x		17	7.7%
4.3		18	2.3%
4.4	KitKat	19	27.7%
5.0	Lollipop	21	13.1%
5.1		22	21.9%
6.0	Marshmallow	23	18.7%
7.0	Nougat	24	0.1%

从上表中可以看出，目前4.0以上的系统已经占据了超过98%的Android市场份额，因此我们本书中开发的程序也只面向4.0以上的系统，2.x的系统就不再兼容了。

### 1.1.3 Android应用开发特色

预告一下，你马上就要开始真正的Android开发旅程了。不过先别急，在开始之前我们再一起来看一看，Android系统到底提供了哪些东西，可供我们开发出优秀的应用程序。

#### 1. 四大组件

Android系统四大组件分别是活动（Activity）、服务（Service）、广播接收器（Broadcast Receiver）和内容提供者（Content Provider）。其中活动是所有Android应用程序的门面，凡是在应用中你看得到的东西，都是放在活动中的。而服务就比较低调了，你无法看到它，但它会一直在后台默默地运行，即使用户退出了应用，服务仍然是可以继续运行的。广播接收器允许你的应用接收来自各处的广播消息，比如电话、短信等，当然你的应用同样也可以向外发出广播消息。内容提供者则为应用程序之间共享数据提供了可能，比如你想要读取系统电话簿中的联系人，就需要通过内容提供者来实现。

#### 2. 丰富的系统控件

Android系统为开发者提供了丰富的系统控件，使得我们可以很轻松地编写出漂亮的界面。当然如果你品味比较高，不满足于系统自带的控件效果，也完全可以定制属于自己的控件。

#### 3. SQLite数据库

Android系统还自带了这种轻量级、运算速度极快的嵌入式关系型数据库。它不仅支持标准的SQL语法，还可以通过Android封装好的API进行操作，让存储和读取数据变得非常方便。

#### 4. 强大的多媒体

Android系统还提供了丰富的多媒体服务，如音乐、视频、录音、拍照、闹铃，等等，这一切你都可以在程序中通过代码进行控制，让你的应用变得更加丰富多彩。

#### 5. 地理位置定位

移动设备和PC相比起来，地理位置定位功能应该可以算是很大的一个亮点。现在的Android手机都内置有GPS，走到哪儿都可以定位到自己的位置，发挥你的想象就可以做出创意十足的应用，如果再结合功能强大的地图功能，LBS这一领域潜力无限。

既然有Android这样出色的系统给我们提供了这么丰富的工具，你还担心做不出优秀的应用吗？好了，纯理论的东西就介绍到这里，我知道你已经迫不及待想要开始真正的开发之旅了，那我们就开始启程吧！

## 1.2 手把手带你搭建开发环境

俗话说得好，“工欲善其事，必先利其器”，开着记事本就想开发Android程序显然不是明智之举，选择一个好的IDE可以极大地提高你的开发效率，因此本节我就将手把手带着你把开发环境搭建起来。

### 1.2.1 准备所需要的工具

我现在对你了解还并不多，但我希望你已经是一个颇有经验的Java程序员，这样你理解本书的内容时将会轻而易举，因为Android程序都是使用Java语言编写的。如果你对Java只是略有了解，那阅读本书应该会有点困难，不过一边阅读一边补充Java知识也是可以的。但如果你对Java完全没有了解，那么我建议你可以暂时将本书放下，先买本介绍Java基础知识的书学上两个星期，把Java的基本语法和特性都学会了，再继续阅读这本书。

好了，既然你已经阅读到这里，说明你已经掌握Java的基本用法了，下面我们就来看看开发Android程序需要准备哪些工具。

- **JDK**。JDK是Java语言的软件开发工具包，它包含了Java的运行环境、工具集合、基础类库等内容。需要注意的是，本书中的Android程序必须要使用JDK 8或以上版本

才能进行开发。

- **Android SDK**。Android SDK是谷歌提供的Android开发工具包，在开发Android程序时，我们需要通过引入该工具包，来使用Android相关的API。
- **Android Studio**。在很早之前，Android项目都是用Eclipse来开发的，相信所有Java开发者都一定会对这个工具非常熟悉，它是Java开发神器，安装ADT插件后就可以用来开发Android程序了。而在2013年的时候，谷歌推出了一款官方的IDE工具Android Studio，由于不再是以插件的形式存在，Android Studio在开发Android程序方面要远比Eclipse强大和方便得多。不过由于Android Studio早期的测试版本并不是非常稳定，所以本书的第1版仍然选用Eclipse来作为开发工具。而如今，Android Studio已经推出了2.2版本，稳定性完全不再是问题，普及程度方面也远超Eclipse，没有比现在更适合的时机来换用Android Studio了，因此本书中所有的代码都将在Android Studio上进行开发。

## 1.2.2 搭建开发环境

当然，上述软件并不需要你去一个个地下载，因为谷歌为了简化搭建开发环境的过程，将所有需要用到的工具都帮我们集成好了，到Android官网就可以下载最新的开发工具，下载地址是：<https://developer.android.google.cn/studio/index.html>。不过，Android官网通常都需要科学上网才能访问，如果你无法访问的话，也可以直接到我的百度网盘去下载，下载地址是：<https://pan.baidu.com/s/1nuABMDb>。（注意网址中是阿拉伯数字1，而不是英文字母l。）

你下载下来的将是一个安装包，安装的过程也很简单，一直点击Next就可以了。其中选择安装组件时建议全部勾选上，如图1.2所示。



图 1.2 选择安装组件

接下来还会让你选择Android Studio的安装地址以及Android SDK的安装地址，这些根据你自己电脑的实际情况选择就行了，不想改动的话就保持默认，如图1.3所示。



图 1.3 选择安装地址

后面就没什么需要注意的了，全部保持默认项，一直点击Next即可完成安装，如图1.4所示。



图 1.4 安装完成

现在点击Finish按钮来启动Android Studio，一开始会让你选择是否导入之前Android Studio版本的配置，由于这是我们首次安装，这里选择不导入就可以了，如图1.5所示。



图 1.5 选择不导入配置

点击OK按钮会进入到Android Studio的配置界面，如图1.6所示。



图 1.6 Android Studio的配置界面

然后点击Next开始进行具体的配置，如图1.7所示。



图 1.7 选择安装类型

这里我们可以选择Android Studio的安装类型，有Standard和Custom两种。Standard表示一切都使用默认的配置，比较方便；Custom则可以根据用户的特殊需求进行自定义。简单起见，这里我们就选择Standard类型了，继续点击Next完成配置工作，如图1.8所示。



图 1.8 完成Android Studio配置

现在点击Finish按钮，配置工作就全部完成了。然后Android Studio会尝试联网下载一些更新，等待更新完成后再点击Finish按钮就会进入Android Studio的欢迎界面，如图1.9所示。



图 1.9 Android Studio的欢迎界面

目前为止，Android开发环境就已经全部搭建完成了。那现在应该做什么？当然是写下你的第一行Android代码了，让我们快点开始吧。

## 1.3 创建你的第一个Android项目

任何一个编程语言写出的第一个程序毫无疑问都会是Hello World，这已经是自20世纪70年代一直流传下来的传统，在编程界已成为永恒的经典，那我们当然也不会搞例外了。

### 1.3.1 创建HelloWorld项目

在Android Studio的欢迎界面点击Start a new Android Studio project，会打开一个创建新项目的界面，如图1.10所示。



图 1.10 创建新项目

其中Application name表示应用名称，此应用安装到手机之后会在手机上显示该名称，这里我们填入HelloWorld。Company Domain表示公司域名，如果是个人开发者，没有公司域名的话，那么就像我一样填example.com就可以了。Package name表示项目的包名，Android系统就是通过包名来区分不同应用程序的，因此包名一定要具有唯一性。Android Studio会根据应用名称和公司域名来自动帮我们生成合适的包名，如果你不想使用默认生成的包名，也可以点击右侧的Edit按钮自行修改。最后，Project location表示项目代码存放的位置，如果没有特殊要求的话，这里也保持默认就可以了。

接下来点击Next可以对项目的最低兼容版本进行设置，如图1.11所示。



图 1.11 设置项目的最低兼容版本

欢迎访问：电子书学习和下载网站 (<https://www.shgis.cn>)

文档名称：《第一行代码Android》郭霖(第2版).epub

请登录 <https://shgis.cn/post/1857.html> 下载完整文档。

手机端请扫码查看：

