

 高等教育规划教材

Android 程序设计

卫颜俊 编著



免费提供电子教案和源代码

下载网址 <http://www.cmpedu.com>



机械工业出版社
CHINA MACHINE PRESS



高等教育规划教材

Android 程序设计

卫颜俊 编著



机械工业出版社

本书介绍了 Android 应用程序开发所需要的基本知识、基本技术和基本方法, 主要内容包括 Android 基础知识与 Android 程序的基本结构, Java 语言基本语法和面向对象程序设计基础, 可视化程序设计, 多界面程序设计, 以及文件、多媒体、网络、数据库和传感器等技术的应用程序设计, 最后介绍了一个天气预报机器人客户端案例。在附录中还提供了 Android 和 Eclipse 的相关参考资料。

本书的组织形式是以 Android 的程序设计实例为主线, 从程序设计基本知识出发, 到基本程序设计方法, 再到应用程序设计技术, 最后是案例剖析。本书的编写原则是学以致用, 特点是由浅入深、结构清晰、内容实用、例题丰富, 非常适合作为具有一定基础的读者系统学习 Android 的教科书或教辅参考书。

本书配套授课电子课件, 需要的教师可登录 www.cmpedu.com 免费注册, 审核通过后下载, 或联系编辑索取 (QQ: 2850823885, 电话: 010-88379739)。

图书在版编目 (CIP) 数据

Android 程序设计/卫颜俊编著. —北京: 机械工业出版社, 2016. 1

高等教育规划教材

ISBN 978-7-111-53289-7

I. ①A… II. ①卫… III. ①移动终端—应用程序—程序设计—高等学校—教材 IV. ①TN929.53

中国版本图书馆 CIP 数据核字 (2016) 第 058301 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

责任编辑: 郝建伟 责任校对: 张艳霞

责任印制: 常天培

北京机工印刷厂印刷 (三河市南杨庄国丰装订厂装订)

2016 年 5 月第 1 版·第 1 次印刷

184mm × 260mm · 17.5 印张 · 434 千字

0 001—3 000 册

标准书号: ISBN 978-7-111-53289-7

定价: 45.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

电话服务

网络服务

服务咨询热线: (010) 88379833 机工官网: www.cmpbook.com

读者购书热线: (010) 88379649 机工官博: weibo.com/cmp1952

教育服务网: www.cmpedu.com

封面防伪标均为盗版

金书网: www.golden-book.com

出版说明

当前,我国正处在加快转变经济发展方式、推动产业转型升级的关键时期。为经济转型升级提供高层次人才,是高等院校最重要的历史使命和战略任务之一。高等教育要培养基础性、学术型人才,但更重要的是加大力度培养多规格、多样化的应用型、复合型人才。

为顺应高等教育迅猛发展的趋势,配合高等院校的教学改革,满足高质量高校教材的迫切需求,机械工业出版社邀请了全国多所高等院校的专家、一线教师及教务部门,通过充分的调研和讨论,针对相关课程的特点,总结教学中的实践经验,组织出版了这套“高等教育规划教材”。

本套教材具有以下特点:

1) 符合高等院校各专业人才的培养目标及课程体系的设置,注重培养学生的应用能力,加大案例篇幅或实训内容,强调知识、能力与素质的综合训练。

2) 针对多数学生的学习特点,采用通俗易懂的方法讲解知识,逻辑性强、层次分明、叙述准确而精炼、图文并茂,使学生可以快速掌握,学以致用。

3) 凝结一线骨干教师的课程改革和教学研究成果,融合先进的教学理念,在教学内容和方法上做出创新。

4) 为了体现建设“立体化”精品教材的宗旨,本套教材为主干课程配备了电子教案、学习与上机指导、习题解答、源代码或源程序、教学大纲、课程设计和毕业设计指导等资源。

5) 注重教材的实用性、通用性,适合各类高等院校、高等职业学校及相关院校的教学,也可作为各类培训班教材和自学用书。

欢迎教育界的专家和教师提出宝贵的意见和建议。衷心感谢广大教育工作者和读者的支持与帮助!

机械工业出版社

前 言

计算机技术的发展非常快，特别是在网络技术、移动技术、智能技术和嵌入式技术应用领域更凸显出其应用的广阔天地，每个人、每个企业，乃至整个社会都值得去追逐。

互联网技术的发展给人们提供了新的机会、新的思路乃至新的领域、新的世界，电子商务、搜索引擎和博客等已家喻户晓。随着互联网技术与 3G、4G 移动通信技术的强强结合，由此诞生了移动互联网技术，其应用可以说是日新月异，许多机关、企业和学校的业务系统都迈入这个应用领域，移动服务和云计算服务成为新的消费增长点。其中，Android 智能手机应用开发遥遥领先于其他平台，占据着大半江山，因此，在新时代，掌握 Android 开发技术不仅仅是追新，更可以给个人的发展提供新的契机。

Java 技术是当今最流行、最实用、最全面的计算机技术之一，Java 语言是 Android 平台开发的核心语言，Java 适用于互联网、无线移动通信设备、嵌入式和游戏开发等大部分应用领域。因此，通过学习 Java 语言程序设计，在 Android 平台上进行具体的应用开发，既能够锻炼学生的计算机学习应用综合能力，又能使新技术找到应用的场所。打好这方面的基础，可以使学生掌握新技术，也为学生将来的深造研究、工作创业及创新进行技术储备。

本书编者开发过多个项目，给各大学、企业和培训中心培训过多门多次计算机相关课程，目前仍然致力于计算机教学、研究和开发工作。

编写这本书也是对编者自己多年教学、研究和开发的一个总结，希望能够给后来者提供一些经验，使读者少走弯路，这是编者一直乐于做的事情。

本书介绍了 Android 应用程序 (App) 开发所需要的基本知识、基本技术和基本方法，主要内容包括 Android 操作系统、Android SDK 及 Android 程序的基本结构，Android 程序开发中所涉及的 Java 语言的基本语法和面向对象程序设计基础，Android 基本可视化程序设计的组成要素，多界面程序设计，文件应用程序设计，多媒体应用程序设计，网络应用程序设计，以及数据库应用程序设计和传感器应用程序设计，最后介绍了一个天气预报机器人客户端案例。在本书附录中还提供了与 Android、Java 和 Eclipse 相关的一些参考资料。各个章节内容安排如下。

第 1 章 Android 简介，主要内容包括 Android 操作系统、Android SDK 及 Android 程序基本结构的简单介绍。

第 2 章 基本语法，主要内容包括 Java 语言的数据类型、运算符和表达式，以及控制结构等。

第 3 章 面向对象，主要内容包括 Java 语言中的类、对象、继承、覆盖、接口和抽象类等。

第 4 章 基本可视化程序设计，主要内容包括 Android 项目的框架结构、界面控件和典型资源设计。

第 5 章 多界面程序设计，主要内容包括菜单、对话框和多活动。

第 6 章 文件应用程序设计，主要内容包括 Java 语言中的文件类与方法、Android 中的

资源文件、内存储器文件和 SD 卡文件的读写方法。

第 7 章 多媒体应用程序设计，主要内容包括图像显示、图形绘制，以及音频和视频的录制与播放。

第 8 章 网络应用程序设计，主要内容包括局域网内的 C/S 结构、Socket 网络通信，以及 Web 和因特网资源访问等。

第 9 章 数据库应用程序设计，主要内容包括数据库基础知识、SQL 语言、SQLite 数据库管理系统，及其在 Android 中的数据库应用开发。

第 10 章 传感器应用程序设计，主要内容包括手机中传感器的基础知识，加速度、光感应和方向等几种典型的传感器的应用，以及 GPS 位置服务等。

第 11 章 综合案例分析与设计，主要内容包括对一个手机网络版的天气预报客户端案例的分析、设计与解剖。

附录的主要内容包 Android SDK 和 Java 的常用包和类、Android 常用资源索引、Eclipse 开发工具常用快捷键、Eclipse 下调试 Android 程序的方法，以及常用的 ADB 命令格式等参考资料。

本书是编者对多年讲授 Android 和 Java 课程的课件、例题、资料和开发经验的总结，基本以 Android 的活动 (Activity) 类型的程序为主线，从程序设计基本知识，到基本程序设计方法，再到应用型程序设计技术，最后是案例解剖，本着学以致用原则，由浅入深、结构清晰、内容实用、例题丰富。本书适合具有一定计算机的基础知识和程序设计初步知识的读者系统学习 Android 之用，也可作为大中专院校和培训机构的教科书或教辅参考书，建议学时至少 48 学时，其中上课 24 学时，上机实验 24 学时以上。由于各章内容是相对独立的，也可以根据学时适当增减，并建议采用多媒体授课方式。

欢迎读者选用本书，但因编者水平有限，本次编写必有纰漏，请提出宝贵的意见和建议；编者将进一步加以改进并致谢。编者的联系方式为 Mr.Java@163.com。

在本书的编写过程中，恩师冯博琴教授给了许多指导，李波和赵英良两位老师也提供了不少支持，家人也都非常支持这项工作，特此表示感谢。

编 者

目 录

出版说明

前言

第 1 章 Android 简介	1	2.5 综合例题	38
1.1 Android 基本知识	1	2.6 习题 2	43
1.1.1 Android 概述	1	第 3 章 面向对象	45
1.1.2 Android SDK 体系结构	3	3.1 类、成员与对象	45
1.1.3 基本 Android 开发工具介绍	4	3.1.1 包与类	45
1.2 Android 程序开发步骤	4	3.1.2 成员变量	46
1.2.1 Android 基本程序结构	4	3.1.3 成员方法	47
1.2.2 使用 Eclipse 开发最简单的 Android 程序	6	3.1.4 构造方法	48
1.3 综合例题	11	3.1.5 对象	48
1.4 习题 1	17	3.2 类的继承	48
第 2 章 基本语法	18	3.2.1 类的继承概述	49
2.1 基本数据类型与变量	18	3.2.2 抽象类	50
2.1.1 整数类型	18	3.2.3 接口	51
2.1.2 字符类型	19	3.3 综合例题	53
2.1.3 浮点类型	19	3.4 习题 3	59
2.1.4 布尔类型	20	第 4 章 基本可视化程序设计	60
2.2 运算符与表达式	20	4.1 基本布局	60
2.2.1 算术运算符	20	4.1.1 LinearLayout	60
2.2.2 关系运算符	21	4.1.2 GridLayout	63
2.2.3 逻辑运算符	21	4.1.3 FrameLayout	65
2.2.4 位运算符	21	4.1.4 RelativeLayout	66
2.3 控制结构与标准输入/输出	22	4.1.5 AbsoluteLayout	67
2.3.1 顺序控制结构与标准输入/输出	22	4.1.6 TableLayout	68
2.3.2 分支控制结构	23	4.2 基本控件与事件	70
2.3.3 循环控制结构	25	4.2.1 Button 按钮	70
2.3.4 异常控制结构	27	4.2.2 TextView 文本框与 EditText 编辑框	72
2.4 复合数据类型	28	4.2.3 CheckBox 校验框与 RadioButton 单选按钮	73
2.4.1 数组类型	28	4.2.4 ProgressBar 进度条与 SeekBar 拖动条	74
2.4.2 字符串类型	31	4.2.5 Spinner 组合框与 ListView 列表框	76
2.4.3 日期时间类型	33	4.2.6 DatePicker 日期选择器与 TimePicker 时间选择器	78
2.4.4 集合类型	35		
2.4.5 映射类型	36		
2.4.6 数学类型和随机类型	37		

4.2.7 AnologClock 模拟时钟与 DigitalClock 数字时钟	80	6.2.2 内存卡文件	138
4.3 基本资源文件	81	6.2.3 SD 卡文件	138
4.3.1 res/layout/	81	6.3 综合例题	140
4.3.2 res/values/	81	6.4 习题 6	146
4.3.3 res/drawable/	84	第 7 章 多媒体应用程序设计	147
4.3.4 res/menu/	84	7.1 图像	147
4.3.5 res/raw/	85	7.1.1 图像控件与图像显示	147
4.3.6 assets	85	7.1.2 图像的处理	149
4.3.7 res/anim/	86	7.2 图形	149
4.3.8 res/xml/	86	7.2.1 图形的绘制	150
4.4 工程部署	87	7.2.2 图形的保存	152
4.5 综合例题	88	7.3 音频与视频等多媒体应用简介	152
4.6 习题 4	97	7.3.1 音频	153
第 5 章 多界面程序设计	98	7.3.2 视频	153
5.1 标题栏的定制	98	7.4 综合例题	154
5.2 菜单	100	7.5 习题 7	172
5.2.1 选项菜单	100	第 8 章 网络应用程序设计	173
5.2.2 上下文菜单	103	8.1 基础知识	173
5.3 多活动与意愿	104	8.1.1 网络通信基础知识	173
5.4 对话框	106	8.1.2 线程基础知识	174
5.4.1 Toast	106	8.2 网络通信应用程序设计	176
5.4.2 AlertDialog.Builder	107	8.2.1 局域网通信	177
5.4.3 带有进度条的对话框效果	110	8.2.2 访问因特网	179
5.4.4 具有对话框效果的活动	112	8.3 JSP 程序设计	181
5.5 综合例题	113	8.3.1 基本标记的使用	181
5.6 习题 5	129	8.3.2 高级标记的使用	182
第 6 章 文件应用程序设计	130	8.4 综合例题	185
6.1 常用文件类简介	130	8.5 习题 8	199
6.1.1 File	130	第 9 章 数据库应用程序设计	200
6.1.2 PrintStream 与 Scanner	131	9.1 数据库基础知识	200
6.1.3 FileOutputStream 与 DataOutputStream	133	9.1.1 SQLite 数据库管理系统	201
6.1.4 FileInputStream 与 DataInputStream	134	9.1.2 SQLite 中的 SQL 语言	202
6.1.5 FileWriter 与 BufferedWriter	134	9.2 数据库应用程序设计的方法	206
6.1.6 FileReader 与 BufferedReader	135	9.2.1 SQLiteDatabase 类的使用方法	207
6.2 Android 中的文件读写	136	9.2.2 Cursor 接口的使用方法	208
6.2.1 资源文件	137	9.2.3 SQLiteOpenHelper 类的使用 方法	209
		9.2.4 在纯 Java 程序中使用 SQLite 数据库	210

9.2.5 在 JSP 程序中使用 SQLite 数据库.....	213	11.2.1 JSON 串的结构.....	245
9.3 综合例题.....	213	11.2.2 Java 语言与 JSON 串.....	246
9.4 习题 9.....	221	11.3 案例：天气预报机器人客户端.....	248
第 10 章 传感器应用程序设计	222	11.3.1 百度天气预报接口 API.....	249
10.1 传感器简介.....	222	11.3.2 人机界面设计.....	251
10.1.1 传感器.....	222	11.3.3 类设计.....	254
10.1.2 Android 传感器.....	222	11.3.4 运行结果.....	264
10.2 传感器应用程序设计的方法.....	224	11.3.5 扩展思考.....	265
10.2.1 传感器应用程序基本框架.....	224	11.4 习题 11.....	265
10.2.2 加速度传感器应用程序设计.....	225	附录	267
10.2.3 方向传感器应用程序设计.....	227	附录 A 常用的 ADB 命令.....	267
10.2.4 光线感应传感器应用程序设计.....	229	附录 B Android SDK 常用的包与类.....	267
10.2.5 GPS 位置服务应用程序设计.....	231	附录 C Android 常用的资源文件及位置.....	269
10.3 综合例题.....	233	附录 D Eclipse 常用的快捷键.....	269
10.4 习题 10.....	242	附录 E Eclipse 下 Android 程序的调试方法.....	270
第 11 章 综合案例分析与设计	244	参考文献	272
11.1 MVC 设计模式.....	244		
11.2 JSON 简介.....	245		

第1章 Android 简介

随着智能手机的普及，手机操作系统变得越来越重要。目前，三大移动操作系统分别是谷歌公司的 Android、苹果公司的 iOS 和微软公司的 Windows Phone。这三种系统的手机各有优缺点，Android 是目前为止功能最完整的智能手机系统，开源是它最大的特点，其上的开发人数和应用程序数量也是最多的；iOS 是一款非常稳定成熟的平台，但价格偏高，不开放，可定制性不强；Windows Phone 问世的时间虽然短，但在易用性方面较好，如强大的云服务及广受欢迎的 Office 工具对用户都很有吸引力。Android 是一个手机开发平台，它采用 WebKit 浏览器引擎，具备触摸屏、高级图形显示和上网功能，用户能够在手机上查看电子邮件、搜索网址、使用 QQ、微信、云服务和观看视频节目等。总的来讲，Android 最为流行的原因是 Android 支持的厂商最多，在其上的开发也是最容易入门的。

本章首先简要介绍 Android SDK 的组成部分及 Android 程序的基本结构，然后讲述 JDK、ADT、Eclipse 和 AVD 的安装、配置与使用步骤，以及怎样通过 AVD 和手机调试并运行 Android 程序，最后通过几个简单的实例展示 Android 程序的基本结构。

1.1 Android 基本知识

Android 是一种缩小版的 Linux 系列操作系统，其上一般安装了许多硬件驱动程序和大量的 App，为了学习 Android 程序开发，首先需要对 Android 操作系统和开发平台有所了解，熟悉 Android 程序的基本结构，以及一种开发工具（比如 Eclipse）和开发步骤。

1.1.1 Android 概述

Android 是谷歌公司于 2007 年发布的基于 Linux 平台的开源移动操作系统平台，该平台由操作系统、中间件、用户界面和应用软件组成。2008 年美国运营商 T-Mobile USA 在纽约正式发布第一款 Google 手机。随后，谷歌与国际开源手机联盟合作开发了 Android 移动开发平台。Google 与运营商、设备制造商、开发商和其他合作者开发了大量硬件、驱动程序和 App，从而使得 Android 大行兼容之道，普及率大大提高。目前最新版是 Android 6.0，不过本书采用较成熟的 Android 4.4（或称 API 19），对应的 Android SDK 和 ADT 为 v22.3。

Android 的基本底层核心、中间层和虚拟机由谷歌公司控制，其中，Android 的基本底层使用 C 语言开发，而中间层包括函数库和虚拟机使用 C++ 开发，最上层包括通话程序、短信程序等各种应用软件，向各公司开放并采用开源开发方式，这一层以 Java 语言为主流编程语言。

Android 系统特性包括应用程序框架技术、Dalvik 虚拟机、内部集成浏览器、优化的

2D 和 3D 图形库、SQLite 数据库技术、多媒体支持、GSM 电话、蓝牙、3G、Wi-Fi、照相机、GPS、指南针和加速度计，以及丰富的开发环境，比如设备模拟器、调试工具、内存及性能分析图表，以及 Eclipse 集成开发环境插件等。

Android 操作系统的架构是分层的，结构非常清晰，分工也很明确，主要分成 3 大层，即操作系统层、中间件层和应用程序层。还可以细分为 5 个层次结构，即 Linux 内核、Android 运行时、库、应用程序框架和应用程序，如图 1-1 所示。



图 1-1 Android 操作系统的层次结构

下面对图 1-1 中的每一层进行简要介绍。

1. Linux 内核层 (Linux Kernel)

Android 基于 Linux 2.6 内核，提供一些如安全、内存管理、进程管理、网络堆栈和驱动模型等核心系统服务。而其中的 Linux 内核 (Linux Kernel) 作为硬件和软件之间的抽象层，隐藏了具体硬件细节而为 Android 库和运行时层提供统一的服务，其实就是提供各种各样的硬件驱动程序，包括显示驱动程序 (Display Driver)、相机驱动程序 (Camera Driver)、蓝牙驱动程序 (Bluetooth Driver)、Flash 内存驱动程序 (Flash Memory Driver)、特殊驱动程序 (Binder IPC)、USB 驱动程序 (USB Driver)、键盘驱动程序 (Keypad Driver)、Wi-Fi 驱动程序 (Wi-Fi Driver)、音频驱动程序 (Audio Driver) 和电源管理程序 (Power Management)。

2. Android 库层 (Libraries)

在 Android 应用框架中包含一个 C/C++ 库的集合，以 API 方式提供系统的各个组件使用。核心库包括显示、绘图、合成和数据存取管理 (Surface Manager)、多媒体库 (Media Framework)、关系数据库引擎 (SQLite)、3D 效果的支持 (OpenGL ES)、位图和矢量字体渲染 (FreeType)、Web 浏览器引擎 (WebKit)、2D 图形引擎库 (SGL)、安全数据通信 (SSL)，以及标准 C 系统库 (Libe)。

3. Android 运行时层 (Android Runtime)

该层包含一个核心库的集合, 提供 Java 语言核心类库中的大部分功能, 每一个 Android 应用程序又是在 Dalvik 虚拟机中运行的。

4. 应用程序框架层 (Application Framework)

该层包含活动管理器 (Activity Manager)、窗口程序管理器 (Window Manager)、内容提供者 (Content Providers)、视图系统 (View System)、通知管理器 (Notification Manager)、软件包管理器 (Package Manager)、电话管理器 (Telephony Manager)、资源管理器 (Resource Manager)、位置管理器 (Location Manager), 以及传感器管理器 (Sensor Manager)。

5. 应用程序层 (Applications)

应用是使用 Java 语言编写的运行在虚拟机上的程序, Android 预装了一个核心应用程序集合, 包括电子邮件客户端、SMS 程序、日历、地图、浏览器、联系人和其他设置, 进一步还可以由第三方和用户开发更加丰富的应用程序。

1.1.2 Android SDK 体系结构

SDK 是指软件开发工具包 (Software Development Kit), 是一个软件开发工程师用于为特定的软件包、软件框架、操作系统和硬件平台等构建的一组应用软件和开发工具的集合。而 Android SDK 专指 Android 程序的专属软件开发工具包。

完整的 Android SDK 目录包括以下文件和文件夹。

- 1) add-ons 是附加的库, 比如 Google Maps 等。
- 2) docs 是 Android SDK API 参考文档。
- 3) sources 是 Android SDK 源代码。
- 4) samples 是 Android SDK 自带的默认示例工程。
- 5) platforms 是每个平台的 SDK 真正的文件, 里面会根据 API 级别划分 SDK 版本, 比如以 Android 4.4 为例, 进入后有一个 android-19 的文件夹, 它包含 Android 4.4 SDK 的主要文件, 其中 data 保存着一些系统资源, skins 是 Android 模拟器的皮肤, templates 是工程创建的默认模板, android.jar 则是该版本的主要 framework 文件。
- 6) build-tools 目录里面包含了重要的编译工具, 比如 aapt、aidl、逆向调试工具 dexdump 和编译脚本 dx。
- 7) platform-tools 保存着一些通用工具, 比如 adb 等非常重要的程序文件。
- 8) tools 作为 SDK 根目录下的 tools 文件夹, 包含了一些重要工具, 比如 ddms 用于启动 Android 调试工具, 如 logcat、屏幕截图和文件管理器, 而 draw9patch 则是绘制 android 平台的可缩放 png 图片的工具, sqlite3 可以在 PC 上操作 SQLite 数据库, 而 monkeyrunner 则是一个不错的压力测试应用, 模拟用户随机按键, mkshcard 则是模拟器 SD 卡映像的创建工具, emulator 是 Android SDK 模拟器主程序, ant 为 ant 编译脚本。在其中的 lib 文件夹中还包括各种 jar 库文件。

Android SDK 目录结构如图 1-2 所示。

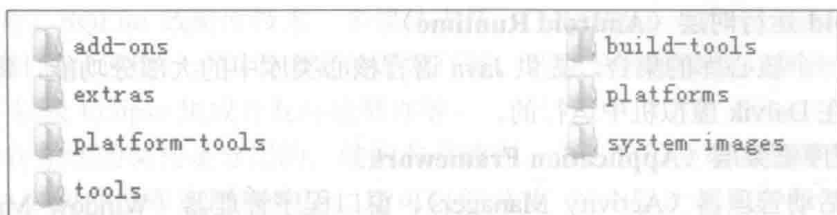


图 1-2 Android SDK 目录结构

1.1.3 基本 Android 开发工具介绍

在开发 Android App 之前，需要安装 Java 和 Android 软件，主要包括 Java JDK、Android SDK 和 API、IDE 集成开发工具 Eclipse，以及 Eclipse 上的 Android 开发插件 ADT。当然也可以选择如 Netbeans、Android Studio 和 IntelliJ IDEA 等其他 IDE，本书所采用的是谷歌公司提供的 Android 与 Eclipse 捆绑套件，是一个免安装版的压缩包，只需要先安装 Java JDK（本书采用“jdk-6u45-windows-i586.exe”），然后解压 Android 与 Eclipse 的捆绑套件压缩包（本书采用“adt-bundle-windows-x86-20131030.zip”），最后在解压后的文件夹中找到 Eclipse.exe，双击这个文件即可打开使用。

1.2 Android 程序开发步骤

一个 Android App 就是一个软件项目，项目中编写的程序代码以 Java 类的源程序文件方式出现，而如人机界面、文本、颜色、菜单、图片和声音等资源却是以 XML 可扩展标记语言文件方式存在的，对于一个较复杂的 App，可能会包含许多 Java 文件和 XML 文件，程序员自己手工维护非常不易，只要使用 Eclipse 开发工具就可以方便管理 Android 项目，大大提高了开发效率，程序员只需要把主要精力放在构思和设计程序上，其他的工作将由 Eclipse 来自动或半自动完成。

1.2.1 Android 基本程序结构

一个简单的 Android 程序只需要关心两个文件即可，一个是称为活动的 Java 类源程序文件，它含有类的定义、类的成员的定义、事件方法的定义、算法代码，以及资源的使用等，是程序的逻辑代码，另一个是人机界面的 XML 描述文件，它是界面大小、布局、字体、颜色和文字等属性的定义，是程序的资源。

活动（Activity）提供抽象的可视化用户界面（或称为图形界面），是一个容器，其中可以放置多个控件，用户与应用程序通过它来交互，活动是 android.app 包中的一个类，需要继承这个类得到活动子类来定制代码，通常一个 Android App 至少编写一个 Activity 子类。所谓活动的生命周期，是指一个图形界面从建立、显示、暂停、恢复、停止到退出等一系列的执行过程，由于这个过程由 Android 操作系统来控制，因此活动的每个生命周期都设计成由操作系统调用的回调方法方式（方法标识符以 on 开头）。表 1-1 列出了活动的 7 个生命周期。

表 1-1 活动的七个生命周期

方法名称	方法体中的代码功能
onCreate()	创建活动完成一些初始化操作
onStart()	活动启动时执行的代码
onResume()	活动获得用户输入焦点时调用
onPause()	活动失去用户输入焦点时调用
onStop()	活动停止时调用
onRestart()	活动重新开始执行时调用
onDestroy()	活动执行完毕或被系统杀掉时调用

表 1-1 中用得最多的是 onCreate()方法，它是程序的图形界面启动时需要执行的第一步，比如设定活动的图形界面布局、打开文件、打开数据库连接、建立连接网络及控件对象建立等代码都在这个方法中编写。最简单的一个活动子类定义如下。

```
import android.os.Bundle;           // 导入 Bundle 系统类
import android.app.Activity;       // 导入 Activity 系统类
public class MainActivity extends Activity { // 继承 Activity 父类
    @Override                       // 声明覆盖父类的 onCreate 方法
    protected void onCreate(Bundle bundle) { // 生命周期方法 onCreate
        super.onCreate(bundle);      // 调用父类的生命周期方法 onCreate
        setContentView(R.layout.activity_main); // 显示来自于资源的图形界面
    }
}
```

其中的 Activity 为系统的活动父类，MainActivity 为自定义的活动子类，由于活动父类已经编写过 onCreate()方法，因此在活动子类中再次编写这个方法时，需要指明强制覆盖 (@Override) 才不容易出错，onCreate()的方法参数为另一个被称为绑定的类 (android.os 包中的 Bundle 类)，此处主要是由 Android 操作系统在建立图形界面时传递给活动的一些值，当前不去使用而已。onCreate()的方法体中仅包含两条代码，第一条为 super.onCreate(bundle); 表示调用父类的同名方法去完成一些默认的任务；第二条为 setContentView(R.layout.activity_main);表示按照所设计好的资源框架布局来显示图形界面，其中的 activity_main 为文件 activity_main.xml 的资源标识符。

布局是图形界面的框架结构，一般是以 XML 文件的形式来保存的，其内容主要是描述在手机屏幕上的图形界面，以及在图形界面中所放置的各个控件的资源 id 号、排列方式、位置、大小、字体、颜色和事件等。布局方式有许多种，最简单的是流线型布局，可以按行或按列排列各个控件，这里的布局文件 activity_main.xml 的内容如下。

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/linearLayout1"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >
```

```

<TextView
    android:id="@+id/textView1"
    android:layout_width="match_content"
    android:layout_height="wrap_content"
    android:text="Hello Android!"
    android:textSize="25dp"
    android:textColor="#ff0000"
    android:background="#ffffff" />
</LinearLayout>

```

以上布局文件中的指定图形界面采用线型（LinearLayout）垂直布局（`android:orientation="vertical"`），布局的资源 id 号通过 `android:id="@+id/linearLayout1"` 设定为 `linearLayout1`，布局的宽度和高度为填满整个手机屏幕（`fill_parent`）；其中放置了一个文本显示框控件（TextView），其资源 id 号通过 `android:id="@+id/textView1"` 设定为 `textView1`，宽度为填满界面的一行（`match_content`），高度为根据自身的文本内容自动调整高度（`wrap_content`），文本内容为 `Hello Android!`，文本大小为 25 个独立像素，文本颜色为红色（`#ff0000`），控件背景为白色（`#ffffff`）。

1.2.2 使用 Eclipse 开发最简单的 Android 程序

本节将使用 Eclipse 开发最简单的 Android 程序，步骤如下。

1) 安装 Java JDK，这里安装的是 JDK 1.6 版 `jdk-6u45-windows-i586.exe`，双击这个可执行文件即可启动安装程序，如图 1-3 所示，然后不需要修改任何配置，一直单击“下一步”到最后即可。

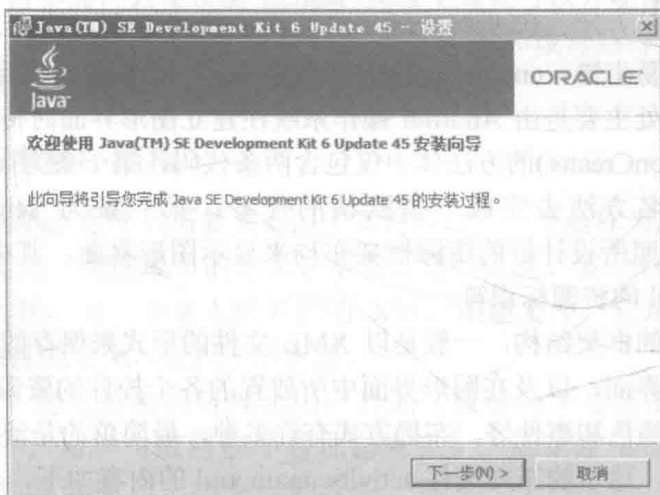


图 1-3 Java JDK 安装起始画面

2) 对 Android 与 Eclipse 的捆绑版 `adt-bundle-windows-x86-20131030.zip` 进行解压，解压到当前文件夹，解压完成后会在当前文件夹中产生一个名为 `adt-bundle-windows-x86-20131030` 的子文件夹。进入这个文件夹，找到 `eclipse` 文件夹再行进入，双击 `eclipse.exe` 可执行文件，进一步会提示确定工作空间（workspace）的位置，用于保存将要开发的每个项

目的全部文件的文件夹，如图 1-4 所示，这里的工作空间名称为 i:\androidworkspace1，然后单击 OK 按钮。

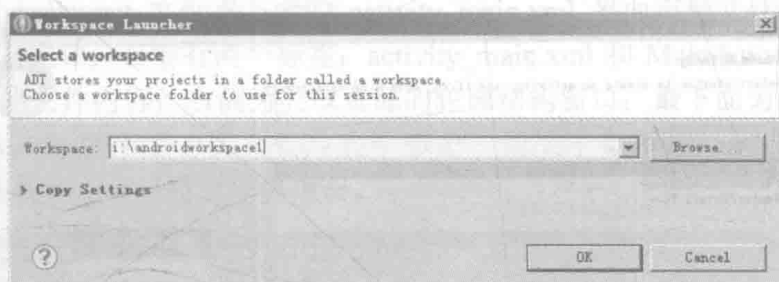


图 1-4 Eclipse 工作空间配置窗口

最后进入 Eclipse 的主界面，如图 1-5 所示。

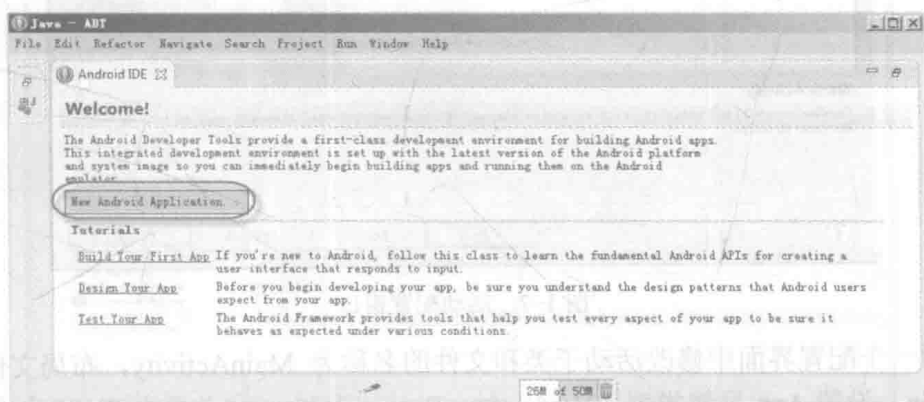


图 1-5 Eclipse 主界面

3) 双击 Eclipse 中的 New Android Application 按钮，新建一个 Android App 工程项目，并设置 App 工程名为 HelloAndroid，包名为 com.example.helloandroid，如图 1-6 所示。

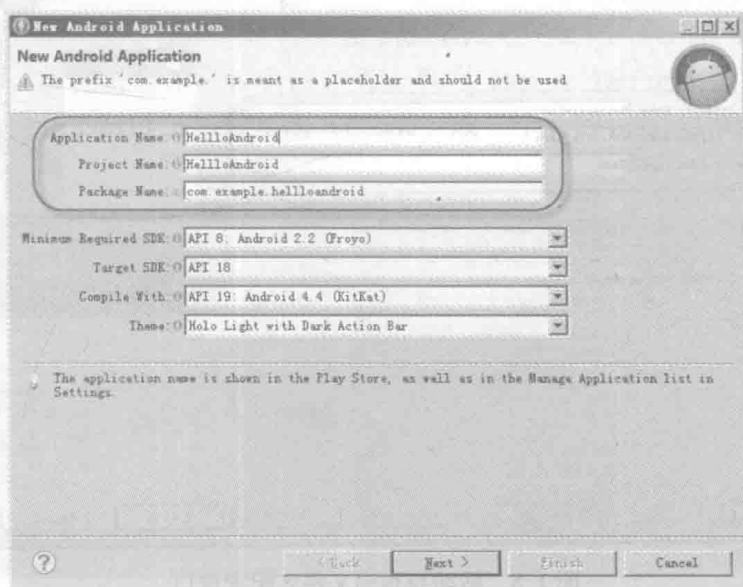


图 1-6 App、工程和包的命名及 SDK 版本配置窗口

接着有好几个工程配置界面，在倒数第二个界面中选择 Create Activity 复选框，并选择 Blank Activity 选项，如图 1-7 所示。

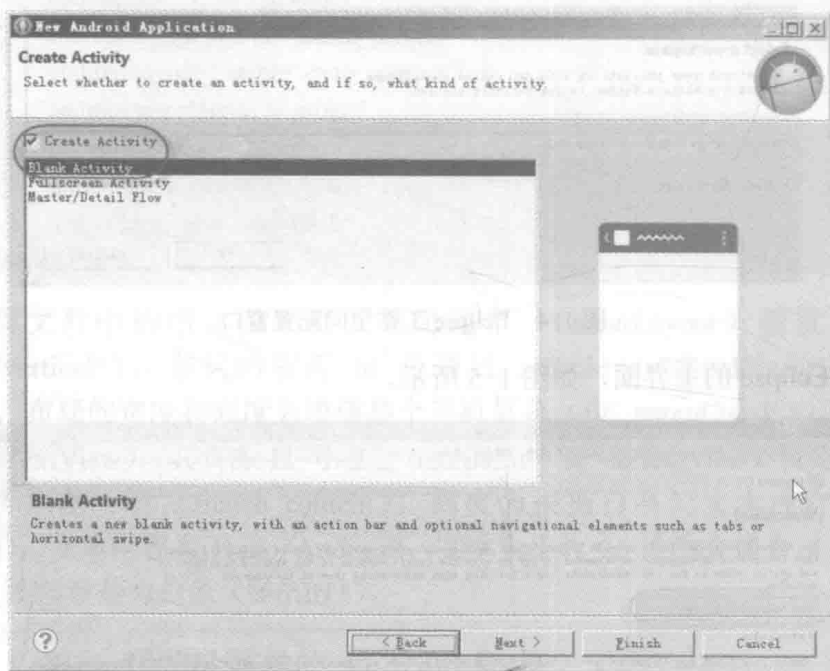


图 1-7 活动配置窗口

在最后一个配置界面中修改活动子类和文件的名称为 MainActivity，布局文件的名称为 activity_main，设置 App 导航类型 (Navigation Type) 为 None，并单击 Finish 按钮，完成最后的配置，如图 1-8 所示。

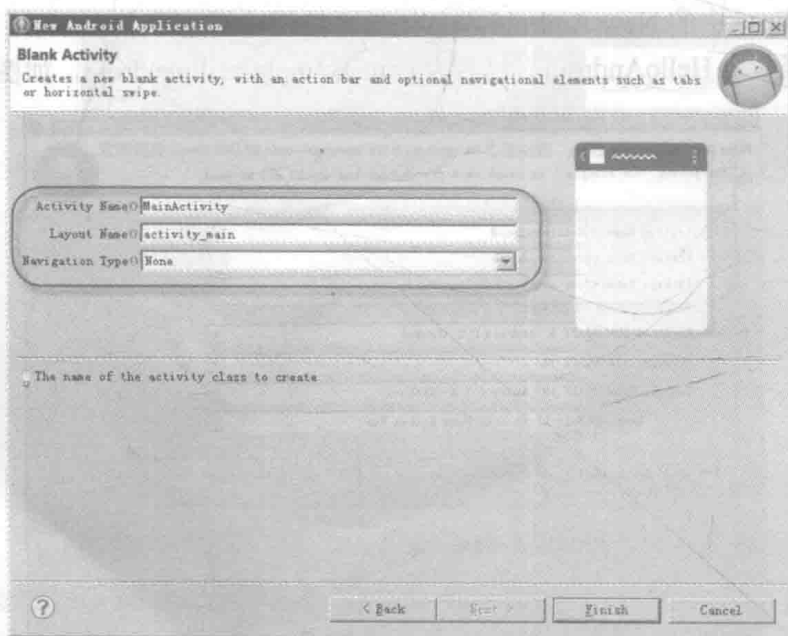


图 1-8 活动与布局文件名配置窗口

配置完成后，Eclipse 会生成所需要的全部文件，稍候会进入 Eclipse 的工程主界面，如