



应用型本科信息大类专业“十三五”规划教材



互联网+



Android

移动应用基础教程

编著 吴晓凌 周方 钟敏 肖立



华中科技大学出版社

<http://www.hustp.com>




应用型本科信息大类专业“十三五”规划教材

Android

移动应用基础教程

编著 吴晓凌 周方 钟敏 肖立



 华中科技大学出版社
<http://www.hustp.com>

中国·武汉

内 容 简 介

本书是面向 Android 应用开发初学者的入门教材,内容基本涵盖了 Android 的主要技术,包括 Android 开发工具、基本界面设计、界面高级技术、Android 生命周期、多媒体和传感器、组件通信及服务、数据存储技术和网络通信技术等。本书可作为高等院校相关课程的教材,也可以作为爱好者的自学用书。

图书在版编目(CIP)数据

Android 移动应用基础教程/吴晓凌等编著. —武汉:华中科技大学出版社,2019.5
应用型本科信息大类专业“十三五”规划教材
ISBN 978-7-5680-5154-5

I. ①A… II. ①吴… 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(2019)第 081898 号

Android 移动应用基础教程

Android Yidong Yingyong Jichu Jiaocheng

吴晓凌 周方 钟敏 肖立 编著

策划编辑: 聂亚文

责任编辑: 史永霞

责任监印: 朱 玢

出版发行: 华中科技大学出版社(中国·武汉)

电话:(027)81321913

武汉市东湖新技术开发区华工科技园

邮编: 430223

录 排: 武汉正风天下文化发展有限公司

印 刷: 武汉华工鑫宏印务有限公司

开 本: 787mm×1092mm 1/16

印 张: 8

字 数: 206 千字

版 次: 2019 年 5 月第 1 版第 1 次印刷

定 价: 30.00 元



本书若有印装质量问题, 请向出版社营销中心调换
全国免费服务热线: 400-6679-118 竭诚为您服务
版权所有 侵权必究

前言

PREFACE

Android 是 Google 公司开发的基于 Linux 平台的开源手机操作系统。自诞生以来,经过不断的发展和完善,其功能日益强大,Android 应用程序开发的需求量也在不断增长,各大高校也逐渐开设了相关课程。

本书主要从教学的角度全面介绍 Android 应用开发的核心知识,共 8 章。第 1 章主要讲解了 Android 的基础知识,包括 Android 起源、Android 体系结构和开发环境搭建等。第 2~3 章主要讲解了如何使用布局和视图创建用户界面,介绍了用户图形界面的常用组件,还包括图像绘制技术和动画技术等较高级的内容。第 4 章主要讲解了 Activity,包括生命周期、创建和使用等,读者可以大致了解 Android 各个组件的工作原理。第 5 章主要讲解了多媒体和传感器的知识,包括播放音频和视频、录音、拍照和传感器检测等。第 6 章主要讲解了 Android 中的两个组件服务和广播接收器,包括服务的创建和生命周期,广播的创建、发送与接收,以及组件间用于通信的 Intent。第 7 章主要讲解了 Android 中的数据存储技术,包括 Shared-Preferences、文件存储、SQLite 数据库和内容提供器等知识。第 8 章主要讲解了 Android 中的网络通信技术,包括使用 Web 视图、基于 HTTP 协议编程和使用 Volley 框架,并介绍了网络传输中的理想数据交换格式 JSON。

本书由武汉生物工程学院组织一批有多年教学经验的计算机专业教师编写。其中,第 1 章由周方和肖立编写,第 2~7 章由吴晓凌编写,第 8 章由钟敏(武汉工程职业技术学院)编写。武汉软帝信息科技有限责任公司李杰对全书进行了审定。

本书的编写受到湖北省高等学校省级教学改革研究项目(2015430)和武汉生物工程学院校本教学研究项目(2018J19)的资助,得到了武汉生物工程学院有关领导和同事以及华中科技大学出版社的领导和编辑的大力支持和帮助,在此向他们一并表示感谢。由于水平有限,书中可能出现错误或不妥之处,敬请批评指正。

编者

2019 年 1 月

目
录

CONTENTS

第 1 章	Android 开发概述	(1)
1.1	Android 简介	(1)
1.2	Android 开发环境	(4)
1.3	Android 开发过程	(9)
1.4	Android 应用程序分析	(17)
第 2 章	UI 基础	(21)
2.1	UI 概述	(21)
2.2	布局管理	(24)
2.3	基本界面组件	(30)
2.4	常用界面组件	(37)
2.5	高级界面组件	(41)
第 3 章	UI 进阶	(45)
3.1	菜单	(45)
3.2	对话框	(48)
3.3	图像绘制技术	(52)
3.4	动画技术	(55)
第 4 章	Android 程序生命周期	(60)
4.1	进程与线程	(60)
4.2	Android 程序的基本组件	(61)
4.3	活动 Activity	(63)
第 5 章	多媒体与传感器	(69)
5.1	音频播放	(69)

5.2	视频播放	(72)
5.3	录音与拍照	(76)
5.4	传感器检测	(79)
第 6 章	组件通信与服务	(82)
6.1	组件通信	(82)
6.2	服务 Service	(85)
6.3	广播接收器 BroadcastReceiver	(88)
6.4	系统服务	(91)
第 7 章	数据存储	(96)
7.1	简单存储 SharedPreferences	(96)
7.2	文件存储	(98)
7.3	数据库存储 SQLite	(100)
7.4	数据共享	(106)
第 8 章	网络通信	(108)
8.1	使用 WebView 显示网页	(108)
8.2	通过 HTTP 访问网络	(112)
8.3	网络通信框架 Volley	(114)
8.4	JSON 数据交换格式	(117)
参考文献	(119)

第 1 章 Android 开发概述

本章主要讲解 Android 开发的基础知识,首先介绍 Android 系统及其体系结构,然后讲解 Android 开发环境的搭建,并通过一个 HelloWorld 程序来讲解如何开发 Android 程序,最后简要分析 Android 应用程序的目录结构和各文件功能。

本章的知识作为 Android 开发者的入门知识,要求初学者对 Android 开发有简单的了解,方便学习后面的知识。

1.1 Android 简介

1.1.1 Android 系统及其特征

Android 是 Google 公司专门为移动设备开发的平台,其中包含了操作系统、中间件和核心应用等。

Android 本义是指“机器人”,Google 公司将 Android 的标识设计为一个绿色机器人,表示 Android 系统符合环保概念,是一个轻薄短小、功能强大的移动系统,是第一个真正为手机打造的开放性系统。

Android 的底层使用开源的 Linux 操作系统,因此具有开放性、平等性、方便性及硬件丰富等特点。

1. 开放性

Android 从底层的操作系统到最上层的应用程序都是开放的,程序开发人员可以很方便地从网络上获取到源代码,可以对源代码进行分析和移植。

2. 平等性

Android 自带的程序和开发人员开发的应用程序是平等的,开发人员可以用自己的应用程序代替系统的程序,构建个性化的 Android 系统。

3. 方便性

Android 为开发人员提供了大量的实用组件库和方便的工具,开发人员只需编写几行代码就可以将功能强大的组件添加到自己的程序中。

4. 丰富的硬件

由于 Android 的开放性,众多的硬件制造商纷纷开发出各种各样的可以与 Android 系统兼容的产品,进一步丰富了 Android 系统的应用。

1.1.2 Android 的发展史

Android 一词最早出现于法国作家利尔·亚当在 1886 年发表的科幻小说《未来夏娃》中,将外表像人的机器起名为 Android。2003 年 10 月,安迪·罗宾(Andy Rubin)等人创建了 Android 公司,并组建了 Android 开发团队,最初的 Android 系统是一款针对数码相机开发的智能操作系统。

Google 公司于 2005 年收购了 Android 公司,并于 2007 年 11 月 5 日正式向外界展示了这

款系统,从此 Android 取得了长足的发展,迅速占领了智能手机的市场份额。在 2010 年底,Android 已经超越称霸 10 年的诺基亚 Symbian 系统,成为全球最受欢迎的智能手机平台。

Android 的第 1 个版本 Android 1.1 在 2008 年 9 月发布,一经推出,Android 的版本升级非常快,几乎每隔半年就有一个新的版本发布。从 Android 1.5 版本开始,Android 用甜点作为系统版本的代号,而且按字母顺序排列。

从早期的 Android 1.5 C、Android 1.6 D 到最近的 Android 9.0 P 一直沿用这个传统,表 1-1 列出了 Android 系统各版本的发布时间及对应的 API 级别。

表 1-1 Android 的各版本及发布时间

版本号	API 级别	别名	发布时间
1.0	1		2008 年 9 月
1.1	2		2009 年 2 月
1.5	3	Cupcake(纸杯蛋糕)	2009 年 4 月
1.6	4	Donut(甜甜圈)	2009 年 9 月
2.0	5	Éclair(闪电泡芙)	2009 年 10 月
2.0.1	6	Éclair(闪电泡芙)	2009 年 12 月
2.1	7	Éclair(闪电泡芙)	2010 年 1 月
2.2~2.2.3	8	Froyo(冻酸奶)	2010 年 5 月
2.3~2.3.2	9	Gingerbread(姜饼)	2010 年 12 月
2.3.3~2.3.7	10	Gingerbread(姜饼)	2011 年 2 月
3.0	11	Honeycomb(蜂窝)	2011 年 2 月
3.1	12	Honeycomb(蜂窝)	2011 年 5 月
3.2	13	Honeycomb(蜂窝)	2011 年 7 月
4.0~4.0.2	14	Ice Create Sandwich(冰激凌三明治)	2011 年 10 月
4.0.3~4.0.4	15	Ice Create Sandwich(冰激凌三明治)	2011 年 12 月
4.1	16	Jelly Bean(果冻豆)	2012 年 6 月
4.2	17	Jelly Bean(果冻豆)	2012 年 10 月
4.3	18	Jelly Bean(果冻豆)	2013 年 7 月
4.4	19	KitKat(奇巧巧克力)	2013 年 10 月
4.4 W	20	KitKat_Watch	2014 年 6 月
5.0	21	Lollipop(棒棒糖)	2014 年 10 月
5.1	22	Lollipop(棒棒糖)	2015 年 3 月
6.0	23	Marshmallow(棉花糖)	2015 年 10 月
7.0	24	Nougat(牛轧糖)	2016 年 8 月
7.1	25	Nougat(牛轧糖)	2016 年 10 月
8.0	26	Oreo(奥利奥)	2017 年 8 月
8.1	27	Oreo(奥利奥)	2017 年 12 月
9.0	28	Pie(馅饼)	2018 年 8 月

从 Android 6.0 开始,Android 系统有了质的飞跃,因此本书主要以 Android 6.0 及其以上版本的开发为例。

1.1.3 Android 的体系结构

Android 系统分为四个层,从高层到低层分别是应用程序层(Applications)、应用程序框架层(Application Framework)、系统运行库层(Libraries)和 Linux 核心层(Linux Kernel),如图 1-1 所示。

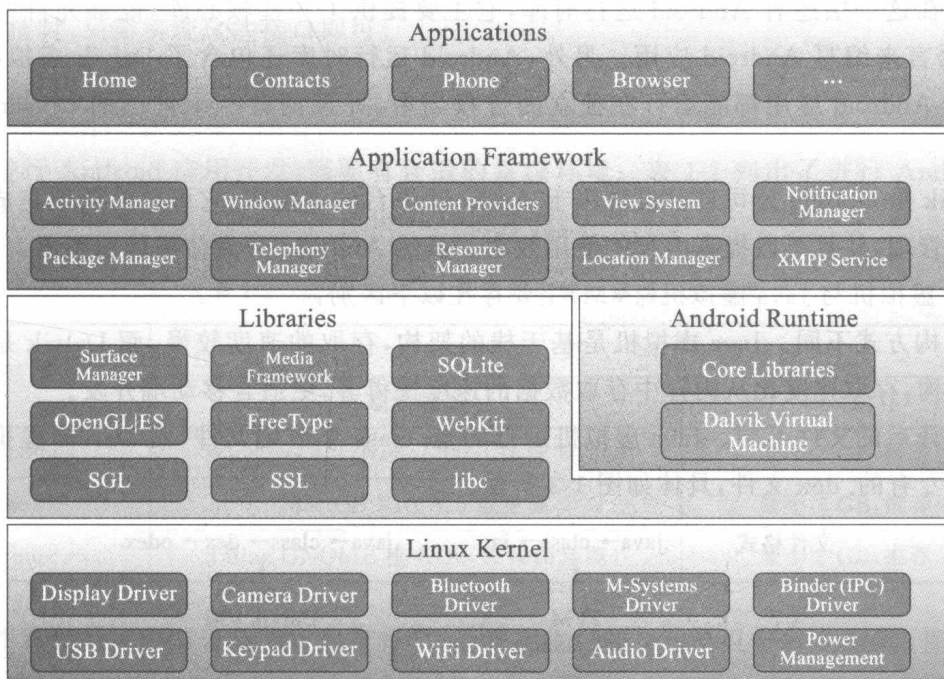


图 1-1 Android 体系结构

1. 应用程序层

应用程序层是核心应用程序的集合,所有安装在手机上的应用程序都是属于这一层的。例如,系统自带的电话拨号程序、短信程序、日历、音乐播放器、浏览器和联系人管理程序等,如图 1-2 所示。所有的应用程序都是用 Java 语言编写的,开发人员自己开发的应用程序也位于这一层。

2. 应用程序框架层

应用程序框架层主要提供构建应用程序时用到的各种 API,Android 自带的一些核心应用就是使用这些 API 完成的,例如视图(View)和活动管理器(Activity Manager)等,开发者也可以使用这些 API 来构建自己的应用程序。本书所讲的程序设计都是基于这个应用程序框架完成的。



图 1-2 Android 系统自带的应用程序

3. 系统运行库层

1) 程序库

系统运行库层通过一些 C/C++ 库来为 Android 系统提供主要的特性支持。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。

2) 运行时库

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 运行时库还包含了 Dalvik 虚拟机,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。

Dalvik 是 Google 公司自己设计用于 Android 平台的虚拟机,它可以简单地完成进程隔离和线程管理,并且可以提高内存的使用效率。虽然 Android 开发使用的编程语言为 Java,但 Dalvik 虚拟机与 Java 虚拟机(JVM)主要存在以下区别:

- 架构方式不同。Java 虚拟机是基于栈的架构,存取的速度较慢,而 Dalvik 是基于寄存器的架构,存取速度比从内存中存取数据的速度快得多,更适合移动端开发。

- 编译后的文件不同。Java 虚拟机运行的是 .class 字节码文件,而 Dalvik 虚拟机运行的则是其专有的 .dex 文件,具体如图 1-3 所示。

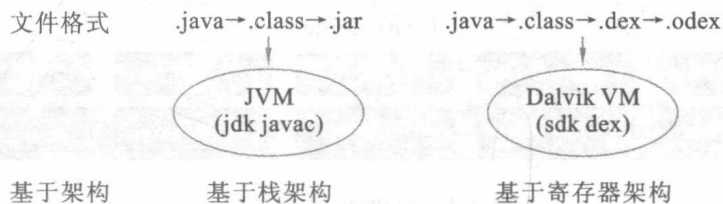


图 1-3 Dalvik 虚拟机与 Java 虚拟机

4. Linux 核心层

Android 系统的核心服务依赖于 Linux 内核,其安全性、内存管理、进程管理、网络协议栈和驱动模型等基本依赖于 Linux。

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、WiFi 驱动和电源管理等。

1.2 Android 开发环境

1.2.1 Android 应用开发

Android 使用 Java 作为程序开发语言,应用程序开发是使用 Android 系统提供的 Java 框架(API)进行开发设计工作。Android 开放了专属的应用程序开发工具 Android SDK,使所有程序开发人员都在统一、开放的平台上进行开发,从而保证了 Android 应用程序的可移植性。

这种开发处于 Android 系统的顶层,开发可以基于硬件设备,即用于测试的实体手机,也可以基于 Android 模拟器。大多数开发者所从事的都是这种开发,本书所介绍的也是这种开发。

除此以外,还有 Android 系统移植开发。移植开发是为了使 Android 系统能在手持移动设备上运行,在具体的硬件系统上构建 Android 软件系统。这种开发在 Android 底层进行,需要移植开发 Linux 中相关的设备驱动程序及 Android 本地框架中的硬件抽象层,也就是需要将设备驱动与 Android 系统联系起来。Android 系统对硬件抽象层都有标准的接口定义,移植时,只需实现这些接口即可。

1.2.2 Android 开发环境分类

要进行 Android 应用开发,需要有合适的系统环境。表 1-2 列出了进行 Android 开发所必需的系统环境需求,本书以 Windows 环境为例。

表 1-2 进行 Android 开发所必需的系统环境需求

操作系统	要求	
	系统版本	内存
Windows	Windows 8/7/Vista/2003(32 或 64 位)	最小 2 GB,推荐 4 GB
Mac OS	Mac OS X10.8.5 或更高	最小 2 GB,推荐 4 GB
Linux	Linux GNOME 或 KDE(K 桌面环境)	最小 2 GB,推荐 4 GB

由于开发过程中需要反复重启模拟器,而每次重启都会消耗几分钟的时间,视机器配置而定,因此使用高配置的机器能节约不少时间。

进行 Android 应用开发,除了要有合适的系统环境外,还需要有一些软件的支持。首先需要有 JDK 和 Android SDK 的支持。JDK 是支持 Java 程序开发和运行的 Java 开发工具包;Android SDK 是 Android 开发工具包,它包括了 Android 开发相关的 API。

此外,还需要准备合适的开发工具,目前有以下两种:

1. Eclipse+ADT

Eclipse 是开源的 Java 集成开发环境(IDE),Google 公司专门为其开发了一个插件来辅助开发,其名称为 Android Development Tools(简称 ADT)。安装 ADT 插件后,不仅可以联机调试,而且还能够模拟各种手机事件和分析程序性能等。一般需要安装 Java SDK 1.5 以上和 Eclipse 3.3 以上版本的环境。

2. Android Studio 系统

Android Studio 系统是 Google 公司推出的一个全新的基于 IntelliJ IDEA 的 Android 开发环境。IntelliJ IDEA 是一种用 Java 语言开发的集成开发环境,被业界公认为最好的 Java 开发工具。

Android Studio 提供了集成的 Android 开发工具,用于应用程序的开发和调试,它提供了功能强大的布局编辑器,可以实现拖拽 UI 组件并进行效果预览。

Android Studio 的优势:

- (1) 比 Eclipse 漂亮；
- (2) 比 Eclipse 速度快；
- (3) 提示和补全功能更加智能和人性化；
- (4) 整合了 Gradle 构建工具；
- (5) 支持 Google Cloud Platform；
- (6) 拥有强大的 UI 编辑器。

Android Studio 系统是 Android 开发工具的未来,本书采用这种开发工具。

1.2.3 Android Studio 的安装与配置

本小节介绍如何搭建 Android 开发环境,其中,JDK 的配置、Android SDK 的下载和 Android Studio 的安装是本小节的难点,需要大家按照书中介绍的步骤仔细操作。若过程中出现问题,不要着急,仔细对照书中介绍的步骤重新配置,最终成功搭建一个好用的 Android 开发环境,为之后的开发打下坚实的基础。

1. 安装与配置 JDK

在安装 Android Studio 之前需要安装 Java JDK 的环境,JDK 原本是 Sun 公司的产品,不过由于 Sun 公司已经被 Oracle 公司收购,因此 JDK 需要到 Oracle 公司的官方网站 (<http://www.oracle.com/index.html>) 下载。下载时选择自己电脑系统的对应版本即可。

双击安装文件进行安装,并选择安装目录。安装完成后,需要在系统的环境变量中进行配置。打开“系统属性”->“高级”->“环境变量”,设置 JAVA_HOME 和 Path 环境变量,如图 1-4 所示。

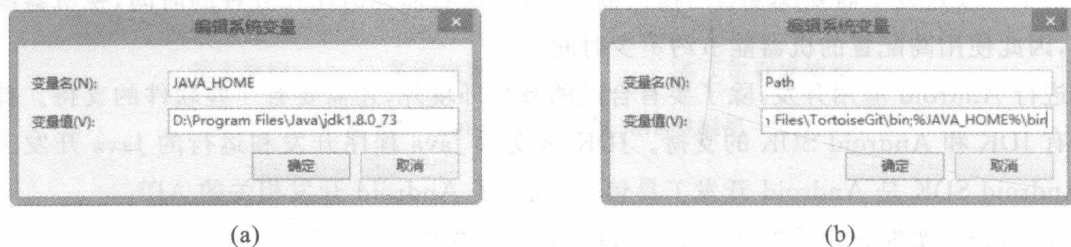


图 1-4 设置 JAVA_HOME 和 Path 环境变量

2. 下载 Android SDK

Android SDK 是 Android 开发工具包,提供了 Android 相关的 API。在 Android 的官方网站 <http://www.android.com/> 中,可以下载到完整版的 Android SDK,也可以下载到包含开发工具 Android Studio 的最新版本的 Android SDK。

在完整版的 Android SDK 中,包含了模拟器、教程、API 文档和示例代码等内容。进行实际项目开发时,建议下载完整版的 Android SDK,这样可以方便查询 API 文档,及时解决遇到的问题。

3. 安装与配置 Android Studio

运行 Android 系统的安装文件,根据安装向导完成安装,如图 1-5 所示。

安装完成后,第一次运行 Android Studio 系统,需要设置 Android SDK,找到 SDK 的存放位置,如图 1-6 所示。



图 1-5 Android Studio 的安装

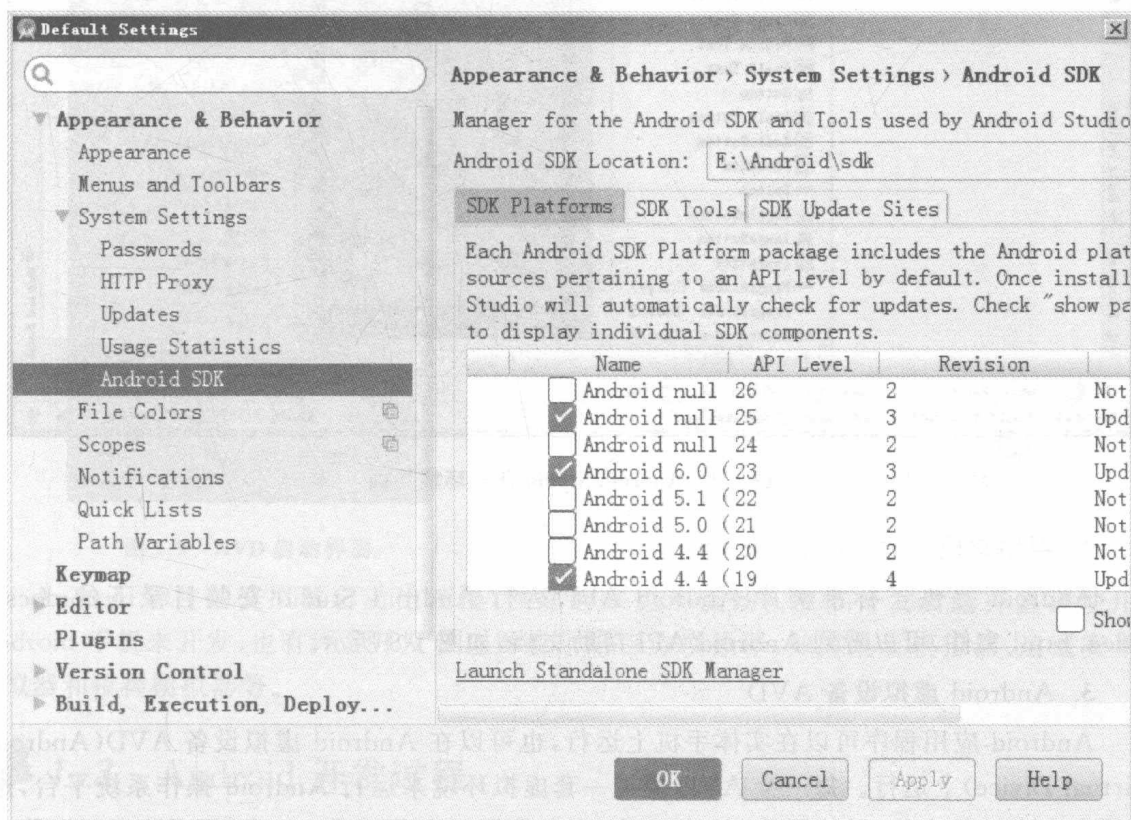


图 1-6 设置 Android SDK 的位置

安装完 Android Studio 系统后还需要设置环境变量：

(1) 在“新建系统变量”对话框中创建变量，名称为“ANDROID_SDK_HOME”，变量值为 Android SDK 的存储位置(如 D:\Android\sdk)。

(2) 在“编辑系统变量”对话框中修改 Path 变量的值，在原变量值最前端添加“%ANDROID_SDK_HOME%\platform-tools; %ANDROID_SDK_HOME%\tools;”。

1.2.4 Android Studio 开发环境

1. 开发环境界面

Android Studio 开发环境界面如图 1-7 所示。

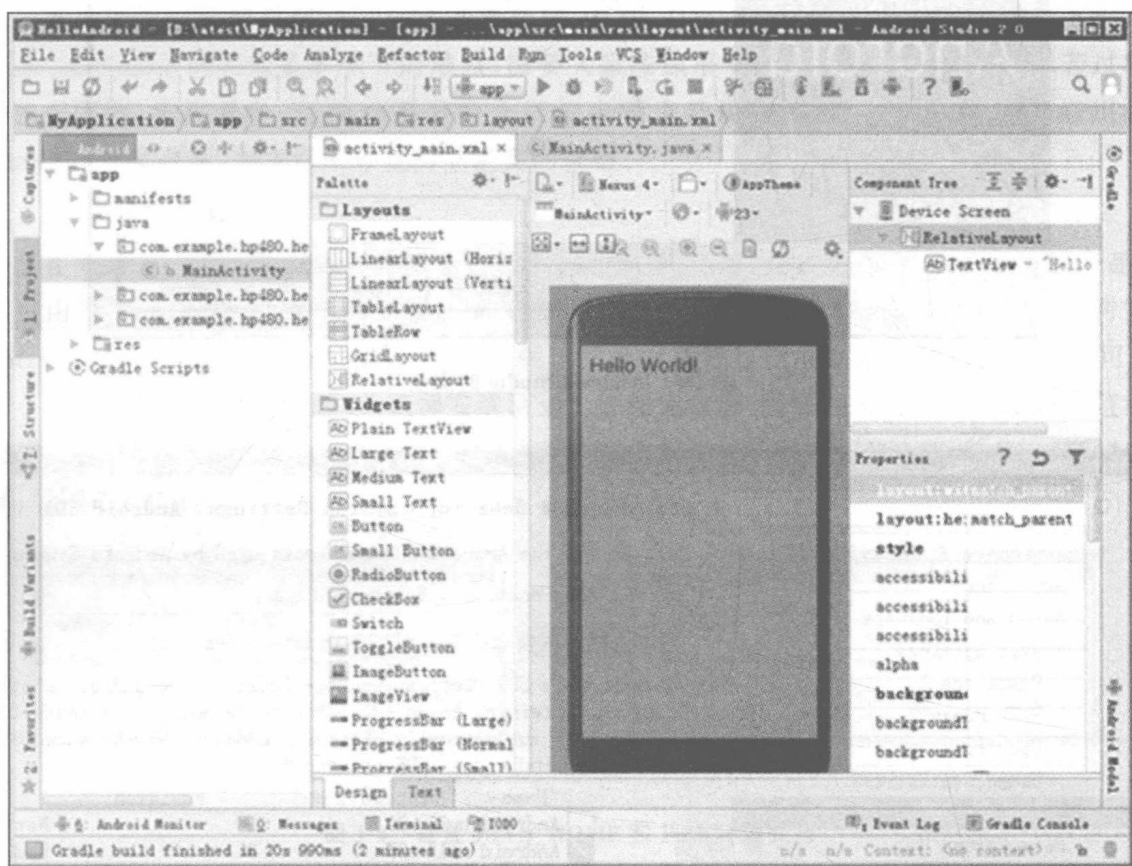


图 1-7 Android Studio 开发环境界面

2. 帮助文档

Android 提供了标准类库 Android API，运行 Android Studio 安装目录下的 docs\index.html 文件，可以看到 Android API 帮助文档，如图 1-8 所示。

3. Android 虚拟设备 AVD

Android 应用程序可以在实体手机上运行，也可以在 Android 虚拟设备 AVD(Android Virtual Device)上运行。每一个 AVD 模拟一套虚拟环境来运行 Android 操作系统平台，这个平台有自己的内核、系统图像、外观显示、用户数据区和仿真的 SD 卡，因此 AVD 也称为模拟器。

AVD 启动界面如图 1-9 所示，项目运行效果如图 1-10 所示。

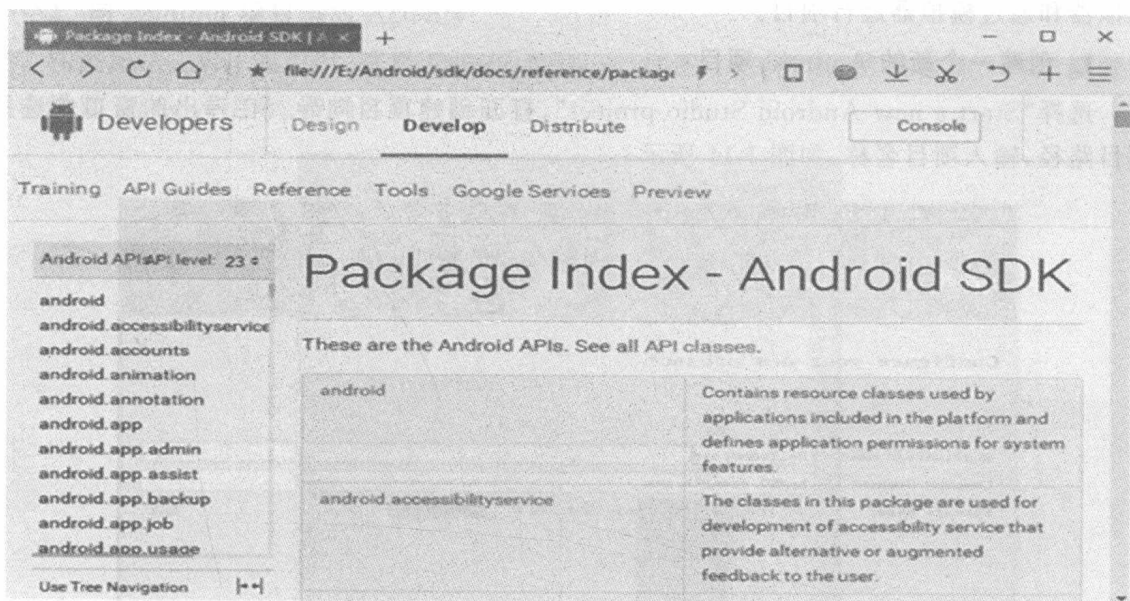


图 1-8 Android API 帮助文档

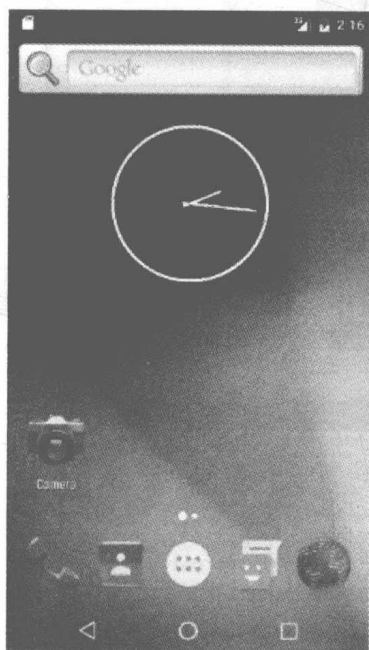


图 1-9 AVD 启动界面



图 1-10 AVD 项目运行效果

Android 原生的模拟器启动比较慢，操作起来也不流畅。很多开发者选择直接使用 Android 手机来开发，也有许多开发者选择使用第三方模拟器，如海马玩模拟器、逍遥安卓模拟器和夜神模拟器等。

1.3 Android 开发过程

1.3.1 第一个 Android 应用程序

本小节通过一个 HelloWorld 程序介绍如何创建一个 Android 程序，如何管理 Android

模拟器和通过模拟器运行项目。

1. 创建一个新的 Android 项目

选择“Start a new Android Studio project”，打开创建项目向导。在弹出的窗口中选择项目路径，输入项目名称，如图 1-11 所示。

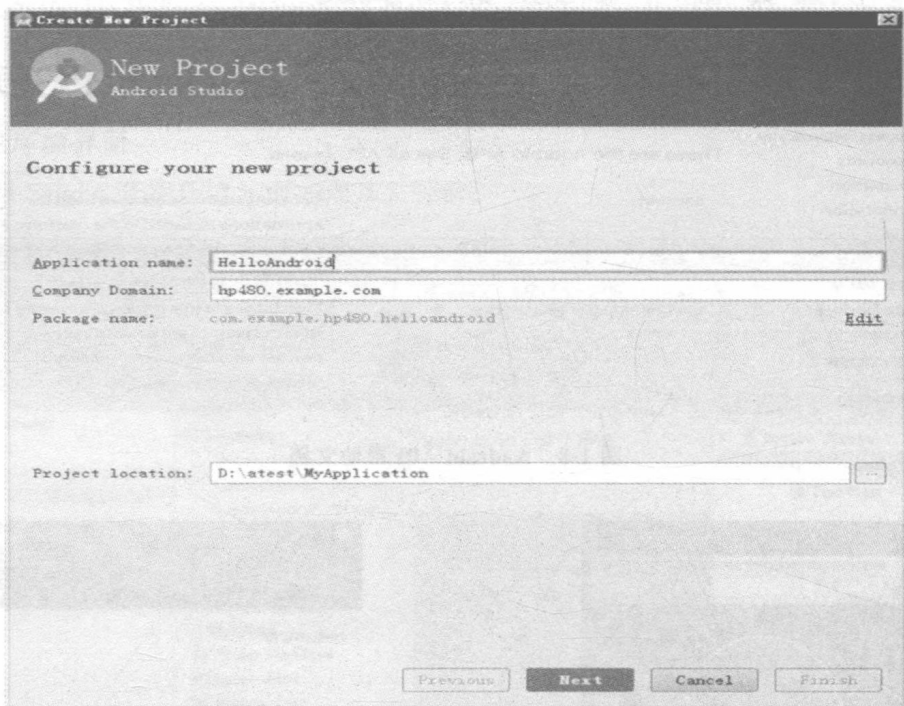


图 1-11 输入 Android 项目名称

在“New Android Application”对话框中输入应用程序名称、项目名称、包名等参数，并选择 Android SDK 的版本，如图 1-12 所示。

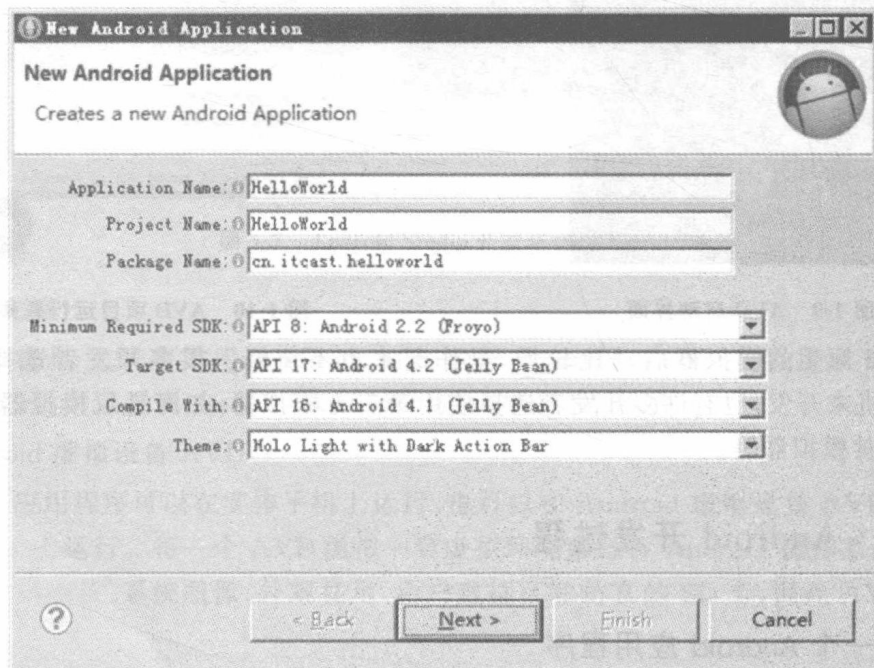


图 1-12 选择 Android SDK 版本

2. 向 Android 项目添加 Activity

在添加 Activity 窗口时,选择 Empty Activity,如图 1-13 所示,然后在 Activity 设置窗口中设置 Activity 名称、布局名称和标题等。

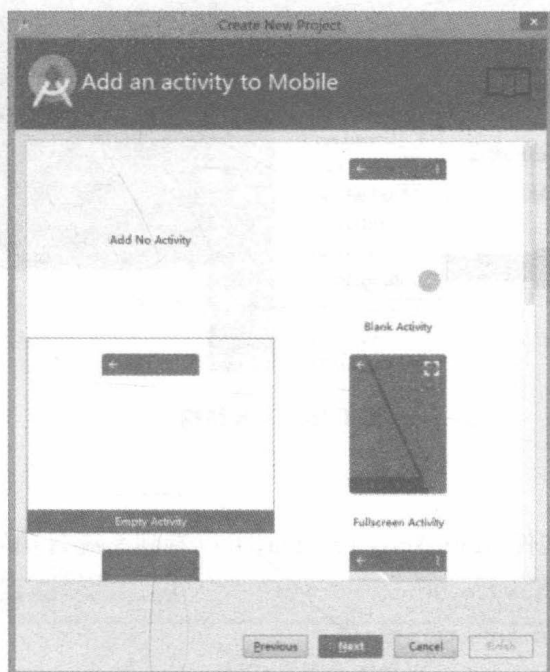


图 1-13 添加 Activity

3. 系统自动生成应用项目框架

最后,系统自动生成一个 Android 应用项目框架,如图 1-14 所示。

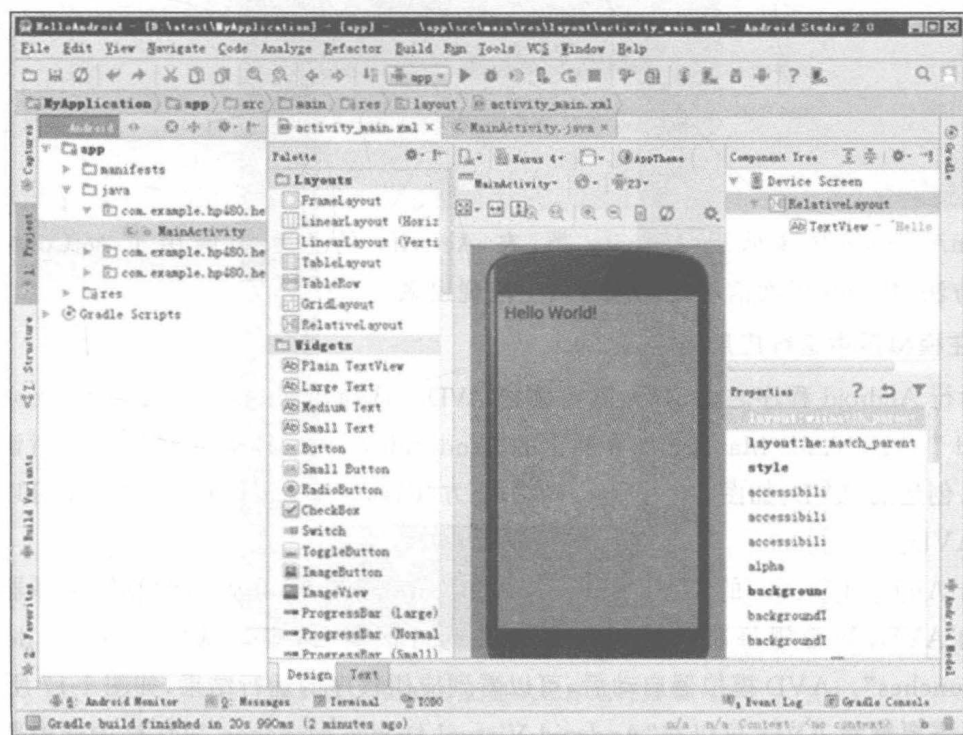


图 1-14 系统自动生成一个 Android 应用项目框架