

<<疯狂Android讲义>>

图书基本信息

书名：<<疯狂Android讲义>>

13位ISBN编号：9787121194856

10位ISBN编号：7121194856

出版时间：2013-3

出版时间：电子工业出版社

作者：李刚

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<疯狂Android讲义>>

内容概要

《疯狂Android讲义(第2版)》内容简介：移动互联网已经成为当今世界发展最快、市场潜力最大、前景最诱人的业务，而Android则是移动互联网上市场占有率最高的平台（已远超iOS，最新统计数据：Android占53.7%，iOS占35%）；与此同时，Android应用选择了Java作为其开发语言，这对于Java来说也是一次极好的机会。

本书是《疯狂Android讲义》的第2版。

本书基于最新的Android 4.2，Android SDK、ADT都基于Android 4.2，书中每个案例、每个截图都全面升级到Android 4.2。

本书全面地介绍了Android应用开发的相关知识，全书内容覆盖了Android用户界面编程、Android四大组件、Android资源访问、图形/图像处理、事件处理机制、Android输入/输出处理、音频/视频多媒体应用开发、OpenGL与3D应用开发、网络通信编程、Android平台的Web Service、传感器应用开发、GPS应用开发、Google Map服务等。

本书并不局限于介绍Android编程的各种理论知识，而是从“项目驱动”的角度来讲授理论。

全书一共包括近百个实例，这些示范性的实例既可帮读者更好地理解各知识点在实际开发中的应用，也可供读者在实际开发时作为参考、拿来就用。

本书最后还提供了两个实用的案例：疯狂连连看和电子拍卖系统Android客户端，具有极高的参考价值。

本书提供了配套的答疑网站，如果读者在阅读本书时遇到技术问题，可以登录疯狂Java联盟（<http://www.crazyit.org>）发帖，笔者将会及时予以解答。

本书适合有一定Java编程基础的读者。

如果读者已熟练掌握Java编程语法并具有一定图形界面编程经验，阅读本书将十分合适。

否则，阅读本书之前建议先认真阅读疯狂Java体系之《疯狂Java讲义》。

<<疯狂Android讲义>>

作者简介

李刚，从事10多年的Java EE应用开发，现任疯狂软件教育中心教学总监。

疯狂Java实训营创始人，疯狂Java体系图书作者。

曾任LITEON公司J2EE技术主管，负责该公司的企业信息化平台的架构设计。

51CTO专家门诊特邀嘉宾。

曾任广州电信、广东龙泉科技等公司的技术培训导师，广东技术师范学院计算机科学系的兼职副教授。

培训的学生已在华为、IBM、阿里软件、网易、电信盈科等名企就职。

国内著名高端IT技术作家，已出版《疯狂Java讲义》、《疯狂Android讲义》、《轻量级Java EE企业应用实战》、《疯狂Ajax讲义》、《疯狂XML讲义》、《经典Java EE企业应用实战》、《疯狂HTML 5/CSS 3/JavaScript讲义》、《Struts 2.x权威指南》等著作。

其中疯狂Java体系图书均已得到广泛的市场认同，多次重印成为超级畅销书，并被多所“985”“211”院校选作教材。

部分图书已被翻译成繁体中文版、授权到台湾地区。

书籍目录

第1章 Android应用与开发环境 1.1 Android的发展和历史 1.1.1 Android的发展和简介 1.1.2 Android平台架构及特性 1.2 搭建Android开发环境 1.2.1 下载和安装Android SDK 1.2.2 安装运行、调试环境 1.2.3 安装Eclipse和ADT插件 1.3 Android常用开发工具的用法 1.3.1 在命令行创建、删除和浏览AVD 1.3.2 使用Android模拟器 (Emulator) 1.3.3 使用DDMS进行调试 1.3.4 Android Debug Bridge (ADB) 的用法 1.3.5 使用DX编译Android应用 1.3.6 使用Android Asset Packaging Tool (AAPT) 打包资源 1.3.7 使用mksdcard管理虚拟SD卡 1.4 开始第一个Android应用 1.4.1 使用Eclipse开发第一个 Android应用 1.4.2 通过ADT运行Android应用 1.5 Android应用结构分析 1.5.1 创建一个Android应用 1.5.2 自动生成的R.java 1.5.3 res目录说明 1.5.4 Android应用的清单文件: AndroidManifest.xml 1.5.5 应用程序权限说明 1.6 Android应用的基本组件介绍 1.6.1 Activity和View 1.6.2Service 1.6.3 BroadcastReceiver 1.6.4 ContentProvider 1.6.5 Intent和IntentFilter 1.7 签名Android应用程序 1.7.1 在Eclipse中对Android 应用签名 1.7.2 使用命令对APK包签名 1.8 本章小结 第2章 Android应用的界面编程 2.1 界面编程与视图 (View) 组件 2.1.1 视图组件与容器组件 2.1.2 使用XML布局文件控制UI 界面 2.1.3 在代码中控制UI界面 2.1.4 使用XML布局文件和Java 代码混合控制UI界面 2.1.5 开发自定义View 2.2 第1组UI组件: 布局管理器 2.2.1 线性布局 2.2.2 表格布局 2.2.3 帧布局 2.2.4 相对布局 2.2.5 Android 4.0新增的网格布局 2.2.6 绝对布局 2.3 第2组UI组件: TextView 及其子类 2.3.1 文本框 (TextView) 与编辑框 (EditText) 的功能和用法 2.3.2 EditText的功能与用法 2.3.3 按钮 (Button) 组件的功能和用法 2.3.4 使用9Patch图片作为按钮背景 2.3.5 单选按钮 (RadioButton) 与复选框 (CheckBox) 的功能与用法 2.3.6 状态开关按钮 (ToggleButton) 与开关 (Switch) 的功能与用法 2.3.7 时钟 (AnalogClock和 DigitalClock) 的功能与用法 2.3.8 计时器 (Chronometer) 2.4 第3组UI组件: ImageView 及其子类 2.5 第4组UI组件: AdapterView 及子类 2.5.1 列表视图 (ListView) 和 ListActivity 2.5.2 Adapter接口及实现类 2.5.3 自动完成文本框 (AutoCompleteTextView) 的功能和用法 2.5.4 网格视图 (GridView) 功能和用法 2.5.5 可展开的列表组件 (ExpandableListView) 2.5.6 Spinner的功能和用法 2.5.7 画廊视图 (Gallery) 的功能和用法 2.5.8 AdapterViewFlipper的功能与用法 2.5.9 StackView的功能与用法 2.6 第5组UI组件: ProgressBar 及其子类 2.6.1 进度条 (ProgressBar) 的功能与用法 2.6.2 拖动条 (SeekBar) 的功能和用法 2.6.3 星级评分条 (RatingBar) 的功能和用法 2.7 第6组UI组件: ViewAnimator 及其子类 2.7.1 ViewSwitcher的功能与用法 2.7.2 图像切换器 (ImageSwitcher) 的功能与用法 2.7.3 文本切换器 (TextSwitcher) 的功能与用法 2.7.4 ViewFlipper的功能与用法 2.8 各种杂项组件 2.8.1 使用Toast显示提示信息框 2.8.2 日历视图 (CalendarView) 组件的功能和用法 2.8.3 日期、时间选择器 (DatePicker和 TimePicker) 的功能和用法 2.8.4 数值选择器 (NumberPicker) 的功能与用法 2.8.5 搜索框 (SearchView) 的功能与用法 2.8.6 选项卡 (TabHost) 的功能和用法 2.8.7 滚动视图 (ScrollView) 的功能和用法 2.8.8 Notification的功能与用法 2.9 对话框 2.9.1 使用AlertDialog创建对话框 2.9.2 对话框风格的窗口 2.9.3 使用PopupWindow 2.9.4 使用DatePickerDialog、TimePickerDialog 2.9.5 使用ProgressDialog创建 进度对话框 2.10 菜单 2.10.1 选项菜单和子菜单 (SubMenu) 2.10.2 使用监听器来监听菜单事件 2.10.3 创建复选菜单项和 单选菜单项 2.10.4 设置与菜单项关联的Activity 2.10.5 上下文菜单 2.10.6 使用XML文件定义菜单 2.10.7 使用PopupMenu创建弹出式菜单 2.11 使用活动条 (ActionBar) 2.11.1 启用ActionBar 2.11.2 使用ActionBar显示 选项菜单 2.11.3 启用程序图标导航 2.11.4 添加Action View 2.11.5 使用ActionBar实现Tab 导航 2.11.6 使用ActionBar实现下拉式导航 2.12 本章小结 第3章 Android的事件处理 3.1 Android事件处理概述 3.2 基于监听的事件处理 3.2.1 监听的模型 3.2.2 事件和事件监听器 3.2.3 内部类作为事件监听器类 3.2.4 外部类作为事件监听器类 3.2.5 Activity本身作为事件监听器 3.2.6 匿名内部类作为事件监听器类 3.2.7 直接绑定到标签 3.3 基于回调的事件处理 3.3.1 回调机制与监听机制 3.3.2 基于回调的事件传播 3.3.3 重写onTouchEvent方法 响应触摸屏事件 3.4 响应的系统设置的事件 3.4.1 Configuration类简介 3.4.2 重写onConfigurationChanged 响应系统设置更改 3.5 Handler消息传递机制 3.5.1 Handler类简介 3.5.2 Handler、Loop、MessageQueue的工作原理 3.6 异步任务 (AsyncTask) 3.7 本章小结 第4章 深入理解Activity与 Fragment 4.1 建立、配置和使用Activity 4.1.1 Activity 4.1.2 配置Activity 4.1.3 启动、关闭Activity 4.1.4 使用Bundle在Activity之间 交换数据 4.1.5 启动其他Activity并返回结果 4.2 Activity的回调机制 4.3 Activity的生命周期与加载模式 4.3.1 Activity

的生命周期演示 4.3.2 Activity与Servlet的相似性与 区别 4.3.3 Activity的4种加载模式 4.4 Fragment详解 4.4.1 Fragment概述及其设计哲学 4.4.2 创建Fragment 4.4.3 Fragment与Activity通信 4.4.4 Fragment管理与Fragment 事务 4.5 Fragment的生命周期 4.6 本章小结 第5章 使用Intent和IntentFilter 进行通信 5.1 Intent对象详解 5.1.1 使用Intent启动系统组件 5.2 Intent的属性及intent—filter配置 5.2.1 Component属性 5.2.2 Action、Category属性与intent—filter配置 5.2.3 指定Action、Category调用系统Activity 5.2.4 Data、Type属性与intent—filter 配置 5.2.5 Extra属性 5.2.6 Flag属性 5.3 使用Intent创建Tab页面 5.4 本章小结 第6章 Android应用的资源 6.1 资源的类型及存储方式 6.1.1 资源的类型以及存储方式 6.1.2 使用资源 6.2 使用字符串、颜色、尺寸资源 6.2.1 颜色值的定义 6.2.2 定义字符串、颜色、尺寸 资源文件 6.2.3 使用字符串、颜色、尺寸 资源 6.3 数组 (Array) 资源 6.4 使用 (Drawable) 资源 6.4.1 图片资源 6.4.2 StateListDrawable 资源 6.4.3 LayerDrawable资源 6.4.4 ShapeDrawable资源 6.4.5 ClipDrawable资源 6.4.6 AnimationDrawable资源 6.5 属性动画 (Property Animation) 资源 6.6 使用原始XML资源 6.6.1 定义原始XML资源 6.6.2 使用原始XML文件 6.7 使用布局 (Layout) 资源 6.8 使用菜单 (Menu) 资源 6.9 样式 (Style) 和主题 (Theme) 资源 6.9.1 样式资源 6.9.2 主题资源 6.10 属性 (Attribute) 资源 6.11 使用原始资源 6.12 国际化和资源自适应 6.12.1 Java国际化的思路 6.12.2 Java支持的语言和国家 6.12.3 完成程序国际化 6.12.4 为Android应用提供国际化资源 6.12.5 国际化Android应用 6.13 自适应不同屏幕的资源 6.14 本章小结 第7章 图形与图像处理 7.1 使用简单图片 7.1.1 使用Drawable对象 7.1.2 Bitmap和BitmapFactory 7.2 绘图 7.2.1 Android绘图基础: Canvas、Paint等 7.2.2 Path类 7.2.3 绘制游戏动画 7.3 图形特效处理 7.3.1 使用Matrix控制变换 7.3.2 使用drawBitmapMesh 扭曲图像 7.3.3 使用Shader填充图形 7.4 逐帧 (Frame) 动画 7.4.1 AnimationDrawable与逐帧动画 7.5 补间 (Tween) 动画 7.5.1 Tween动画与Interpolator 7.5.2 位置、大小、旋转度、透明度改变的补间动画 7.5.3 自定义补间动画 7.6 属性动画 7.6.1 属性动画的API 7.6.2 使用属性动画 7.7 使用SurfaceView实现动画 7.7.1 SurfaceView的绘图机制 7.8 本章小结 第8章 Android数据存储与IO 8.1 使用SharedPreferences 8.1.1 SharedPreferences与Editor简介 8.1.2 SharedPreferences的存储位置和格式 8.1.3 读、写其他应用SharedPreferences 8.2 File存储 8.2.1 openFileOutput和 openFileInput 8.2.2 读写SD卡上的文件 8.3 SQLite数据库 8.3.1 SQLiteDatabase简介 8.3.2 创建数据库和表 8.3.3 使用SQL语句操作SQLite数据库 8.3.4 使用sqlite3工具 8.3.5 使用特定方法操作SQLite数据库 8.3.6 事务 8.3.7 SQLiteOpenHelper类 8.4 手势 (Gesture) 8.4.1 手势检测 8.4.2 增加手势 8.4.3 识别用户的手势 8.5 自动朗读 (TTS) 8.6 本章小结 第9章 使用ContentProvider实现数据共享 第10章 Service与BroadcastReceiver 第11章 多媒体应用开发 第12章 OpenGL与3D应用开发 第13章 Android网络应用 第14章 管理Android手机桌面 第15章 传感器应用开发 第16章 GPS应用开发 第17章 使用Google Map服务 第18章 疯狂连连看 第19章 电子拍卖系统

章节摘录

版权页：插图：使用BroadcastReceiver组件接收广播消息比较简单，开发者只要实现自己的BroadcastReceiver子类，并重写onReceive（Context context，Intent intent）方法即可。

当其他组件通过sendBroadcast（）、sendStickyBroadcast（）或sendOrderedBroadcast（）方法发送广播消息时，如该BroadcastReceiver也对该消息“感兴趣”（通过IntentFilter配置），BroadcastReceiver的onReceive（Context context，Intent intent）方法将会被触发。

开发者实现了自己的BroadcastReceiver之后，通常有两种方式来注册这个系统级的“事件监听器”。

在Java代码中通过Context.registerReceiver（）方法注册BroadcastReceiver。

在AndroidManifest.xml文件中使用元素完成注册。

读者此处只要对BroadcastReceiver有一个大致的印象即可，本书后面的章节还会详细介绍如何开发、使用BroadcastReceiver组件。

1.6.4 ContentProvider 对于Android应用而言，它们必须相互独立，各自运行在自己的Dalvik虚拟机实例中，如果这些Android应用之间需要实现实时的数据交换。

例如我们开发了一个发送短信的程序，当发送短信时需要从联系人管理应用中读取指定联系人的数据——这就需要多个应用程序之间进行数据交换。

Android系统为这种跨应用的数据交换提供了一个标准：ContentProvider。

当用户实现自己的ContentProvider时，需要实现如下抽象方法。

insert（Uri，ContentValues）：向ContentProvider插入数据。

delete（Uri，ContentValues）：删除ContentProvider中指定数据。

update（Uri，ContentValues，String，String（））：更新ContentProvider中指定数据。

query（Uri，String（），String，String（），String）：从ContentProvider查询数据。

通常与ContentProvider结合使用的是ContentResolver，一个应用程序使用ContentProvider暴露自己的数据，而另一个应用程序则通过ContentResolver来访问数据。

1.6.5 Intent和IntentFilter 严格地说，Intent并不是Android应用的组件，但它对于Android应用的作用非常大——它是Android应用内不同组件之间通信的载体。

当Android运行时需要连接不同的组件时，通常就需要借助于Intent来实现。

Intent可以启动应用中另一个Activity，也可以启动一个Service组件，还可以发送一条广播消息来触发系统中的BroadcastReceiver。

也就是说，Activity、Service、BroadcastReceiver三种组件之间的通信都以Intent作为载体，只是不同组件使用Intent的机制略有区别而已。

当需要启动一个Activity时，可调用Context的startActivity（Intent intent）或startActivityForResult（Intent intent，int requestCode）方法，这两个方法中的Intent参数封装了需要启动的目标Activity的信息。

<<疯狂Android讲义>>

编辑推荐

《疯狂Android讲义(第2版)》第一版荣获“电子工业出版社最畅销图书奖”；累计印刷10次，销售码洋二百余万，是Android学习、开发人员必备之选。

1. 知识全面，覆盖面广本书深入阐述了Android应用开发的Activity、Service、BroadcastReceiver与ContentProvider四大组件，并详细介绍了Android全部图形界面组件的功能和用法，Android各种资源的管理与用法，Android图形、图像处理，事件处理，Android输入/输出处理，视频/视频等多媒体开发，OpenGL-ES开发，网络通信，传感器和GPS开发等内容，全面覆盖Android官方指南，在某些内容上更加具体、深入。

2. 内容实际，实用性强本书并不局限于枯燥的理论介绍，而是采用了“项目驱动”的方式来讲授知识点，全书近百个实例，几乎每个知识点都可找到对应的参考实例。

本书最后还提供了“疯狂连连看”、“电子拍卖系统Android客户端”两个应用，具有极高的参考价值。

3. 讲解详细，上手容易本书保持了“疯狂Java体系”的一贯风格：操作步骤详细、编程思路清晰，语言平实。

只要读者有一定的Java编程基础，阅读本书将可以很轻松地地上手Android应用开发；学习完本书最后的两个案例后，读者即可完全满足实际企业中Android应用开发的要求。

<<疯狂Android讲义>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>