



配备
源码、电子课件

普通高等职业教育 计算机系列规划教材

Android Studio 移动应用开发基础



◆ 吴绍根 罗 佳 主编

MOBILE
INTERNET



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

普通高等职业教育计算机系列规划教材

Android Studio

移动应用开发基础

吴绍根 罗佳 主编

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书是 Android Studio 移动应用开发系列教材中的基础篇，书中不仅讲解了 Android 的基本理论知识，还列举了大量示例来帮助读者理解和掌握知识点。主要内容包括 Android 概述、Android 界面开发、Android 数据存储、Android 多媒体开发及网络开发等。本书针对各个章节涉及的知识点，安排多个示例引导读者学习，由易到难，循序渐进。笔者通过一步一步的操作这些案例来介绍知识点的应用情况，同时，针对每个案例设计了对应的练习题，让读者在完成知识点学习之后能够有一个相对应的实践过程。

本书内容翔实，案例经典，实践性强，既可作为高职、高专移动互联应用技术专业课程的教材和教学参考书，也可供从事 Android 移动编程开发的用户学习和参考。无论是拥有丰富 Java 开发经验的程序员，还是只有 Java 基础的初学者，本书都将是十分有价值的学习资料。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有，侵权必究。

图书在版编目 (CIP) 数据

Android Studio 移动应用开发基础 / 吴绍根，罗佳主编. —北京：电子工业出版社，2019.8

普通高等职业教育计算机系列规划教材

ISBN 978-7-121-36993-3

I. ①A… II. ①吴… ②罗… III. ①移动终端—应用程序—程序设计—高等职业教育—教材 IV. ①TN929.53

中国版本图书馆 CIP 数据核字 (2019) 第 131962 号

责任编辑：徐建军 特约编辑：田学清

印 刷：北京七彩京通数码快印有限公司

装 订：北京七彩京通数码快印有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱

邮编：100036

开 本：787×1092 1/16 印张：16

字数：461 千字

版 次：2019 年 8 月第 1 版

印 次：2019 年 8 月第 1 次印刷

定 价：46.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：(010) 88254888，88258888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：(010) 88254570。

本书是 Android Studio 移动应用开发系列教材中的基础篇，是一本介绍 Android 应用开发基础的实用教材，全面介绍了 Android 平台上应用开发各方面的基础知识。本书的主要风格是：语言通俗易懂、操作步骤详细、编程思路清晰。本书知识点的组织由浅入深，循序渐进，读者只要具备基本的 Java 基础，阅读本书就不会有太大问题。

本书中的示例都是针对知识点精心设计的，并且有相对应的练习题来供读者练习。在学习时，只要按照书中解决问题的步骤一步一步做下去，读者就会对所学知识点有一个清楚的认知，然后把书中对应的示例独立练习一遍，读者将会对知识点有进一步的巩固，从而做到“学中做，做中学”。

本书还具有如下几个特点。

1. 基础知识全面

本书深入阐述了 Android 应用开发的核心基础组件，并详细介绍了 Android 全部图形界面组件的功能和用法、Android 各种资源的管理和用法、事件处理、Android 输入和输出处理、音频和视频等多媒体开发、网络通信等内容，其涉及的内容是所有 Android 开发人员必备的知识。

2. 案例驱动，实用性强

本书没有枯燥的理论介绍，而是采用了“案例驱动”的方式来讲授知识点，每个知识点都可以找到相对应的参考示例，同时笔者还针对性地设计了对应的练习题供读者独立实践，使读者在学习之后能够通过实践再一次巩固知识点。

3. 通俗易懂，讲解详细

只要读者具备一定的 Java 编程基础，阅读本书后就可以很轻松地上手进行 Android 应用开发。

笔者具有多年从事 Java 及 Android 等移动应用开发的教学经验。本书由广东轻工职业技术学院的吴绍根和罗佳组织编写，最后由罗佳负责统稿并审校。在编写过程中，企业工程师吴边等提供了大量真实的案例和许多宝贵的建议，在此，编者一并表示衷心的感谢！

为了方便教师教学，本书配有电子教学课件及相关资源，请有此需要的教师登录华信教育资源网 (www.hxedu.com.cn) 注册后免费进行下载，如有问题可在网站留言板留言或与电子工业出版社联系 (E-mail:hxedu@phei.com.cn)。

由于编者水平有限，编写时间仓促，书中难免存在疏漏和不足，恳请同行专家和读者给予批评和指正。

目 录

第 1 章	Android 概述	1
1.1	Android 是什么	1
1.2	Android 应用程序的组成	3
1.3	Android 的发展历史	4
1.4	Android 应用开发环境概述	4
第 2 章	建立 Android 应用开发环境	5
2.1	下载和安装 Android Studio	5
2.2	开发第一个 Android 应用程序	5
2.2.1	创建 Hello World 程序工程	5
2.2.2	运行 Hello World 程序	9
2.3	Android 应用程序的结构	14
2.4	同步练习	15
第 3 章	剖析 Android 应用程序	16
3.1	AndroidManifest.xml	16
3.2	MainActivity.java——Activity 介绍	19
3.3	Android 程序资源	21
3.3.1	字符串资源	21
3.3.2	布局资源	23
3.3.3	ID 资源	24
3.3.4	图片资源	27
3.3.5	Android 的其他资源	28
3.3.6	引用资源	28
3.4	同步练习	30
第 4 章	深入分析 Activity	31
4.1	Activity 的生命周期	31
4.2	Activity 生命周期示例	33
4.3	使用 Log 类输出程序调试信息	36
4.4	Android 常见 Activity	38
4.5	同步练习	39

第 5 章	Android 常用 UI 组件	40
5.1	使用基于 XML 的布局	40
5.2	Android 基本组件	44
5.2.1	Button	44
5.2.2	TextView	47
5.2.3	ImageView	49
5.2.4	EditText	49
5.2.5	CheckBox	50
5.2.6	RadioButton	50
5.3	同步练习一	50
5.4	Android 容器组件	50
5.4.1	LinearLayout	50
5.4.2	RelativeLayout	55
5.4.3	FrameLayout	57
5.4.4	ScrollView	60
5.4.5	CoordinatorLayout	63
5.5	同步练习二	64
5.6	AdapterView	64
5.6.1	AdapterView 入门	64
5.6.2	Adapter	65
5.6.3	ListView	66
5.6.4	Spinner	77
5.6.5	GridView	83
5.7	同步练习三	87
5.8	Android 其他常用组件	87
5.9	同步练习四	87
第 6 章	样式和主题	88
6.1	样式入门	88
6.2	定义样式	92
6.2.1	定义样式的一般方法	92
6.2.2	样式定义中的可用属性	93
6.3	应用样式	94
6.3.1	将样式应用到某个组件	95
6.3.2	将样式应用到某个 Activity 或整个 Application	95
6.4	使用 Android 平台已定义的样式和主题	96
6.5	Android 应用程序的主题样式结构分析	97
6.6	同步练习	98

第 7 章 理解和使用 Intent	99
7.1 Intent 应用入门案例	99
7.2 同步练习一	103
7.3 细说 Intent	103
7.3.1 Intent 的 action	106
7.3.2 Intent 的 data	106
7.3.3 Intent 的 category	108
7.3.4 Intent 的 extra	109
7.4 Intent 解析	109
7.5 获得 Activity 返回的结果	109
7.6 Intent 的综合应用举例	115
7.6.1 运行效果	115
7.6.2 程序代码	117
7.7 同步练习二	121
7.8 广播消息和广播接收器	121
7.8.1 发送和接收普通消息	122
7.8.2 接收 Android 平台的广播消息	127
7.9 同步练习三	127
第 8 章 菜单和 Toolbar	128
8.1 菜单	128
8.2 ActionBar 和 Toolbar	132
8.3 同步练习	135
第 9 章 动画	136
9.1 View 动画之补间动画基础	136
9.1.1 补间动画举例	136
9.1.2 补间动画类型	139
9.1.3 使用动画监听器	142
9.2 View 动画之帧动画	144
9.3 同步练习	147
第 10 章 多媒体播放	148
10.1 使用 MediaPlayer 播放音频	148
10.1.1 播放简短的音频	148
10.1.2 使用 MediaPlayer 自制一个音频播放器	151
10.2 同步练习一	165
10.3 播放视频	165
10.4 同步练习二	167

第 11 章 保存程序数据	168
11.1 使用 SharedPreferences 保存程序数据	168
11.2 同步练习一	172
11.3 设置程序首选项	172
11.4 同步练习二	180
11.5 在程序目录下存储程序数据	181
11.6 同步练习三	181
11.7 访问外部存储器	181
11.7.1 检查 SD 卡状态	182
11.7.2 获得 SD 卡上特定子目录的 File 对象	182
11.8 使用 SQLite 数据库保存程序数据	183
11.8.1 SQLite 数据库介绍	183
11.8.2 在 Android 中使用 SQLite 数据库	183
第 12 章 使用后台任务	194
12.1 使用 Java 线程执行后台任务	194
12.2 同步练习一	198
12.3 使用 AsyncTask 执行后台任务	198
12.4 使用 Service 完成后台任务	203
12.5 同步练习二	211
第 13 章 使用网络	212
13.1 使用 ConnectivityManager 管理网络状态	212
13.2 使用 HttpURLConnection 访问网络	214
13.2.1 使用 HttpURLConnection 的 GET 方法获取图片	215
13.2.2 使用 HttpURLConnection 的 POST 方法获取图片	220
13.3 同步练习一	223
13.4 使用 OkHttp 访问网络	223
13.4.1 使用 Get 方法进行服务请求	223
13.4.2 使用 Post 方法进行服务请求	224
13.4.3 设置请求头及提取响应头	226
13.4.4 配置 OkHttp 超时	226
13.5 OkHttp Get 实现示例	227
13.6 OkHttp Post 实现示例	231
13.7 同步练习二	235
13.8 使用 Multipart 传递请求数据到服务器端程序	235
13.9 同步练习三	241
13.10 使用 JSON 格式的数据与服务器端通信	241

13.10.1	JSON 基础	241
13.10.2	在 JavaScript 中使用 JSON 数据	242
13.10.3	在 Java 中使用 JSON 数据	242
13.10.4	使用 POST 方法及 JSON 数据格式发送请求	243

Android 概述

说到智能手机，大家可能马上就能联想到 Android（中文翻译为“安卓”）、iOS 和 Windows Phone 等手机操作系统。确实，在当今，智能手机是如此的流行，以至于上至老年人，下至几岁孩童都在使用智能手机。市场研究公司 IDC 称，在 2018 年，Android 手机销量在全球市场上的占比达到 85.2%。在未来五年内，Android 手机的销量有望达到 2.4% 的复合年均增长率，到 2022 年将达到 14.1 亿部。

可以说 Android 占据了绝对优势。因此，作为应用开发人员，开发基于 Android 的手机应用是一个重要的方向。

1.1 Android 是什么

Android 是如此的流行，那么，Android 到底是什么？这个问题有点抽象，但是，作为即将踏入 Android 开发阵营的读者需要了解这个问题的答案。注意，只需要了解就可以了，即使在阅读这段文字时对有些概念不太理解，这也是正常的，不会影响将来开发出高水平的 Android 应用。

Android 是什么？Android 是一个平台，它包括基础系统、开发工具和完整的文档。Android 平台是一个通用的计算平台，它采用 Linux 为其支撑操作系统，以 Java 作为其开发环境，通过编程实现完整的电话、视频、网络、界面设计等基础功能。Android 平台的体系结构如图 1-1 所示。

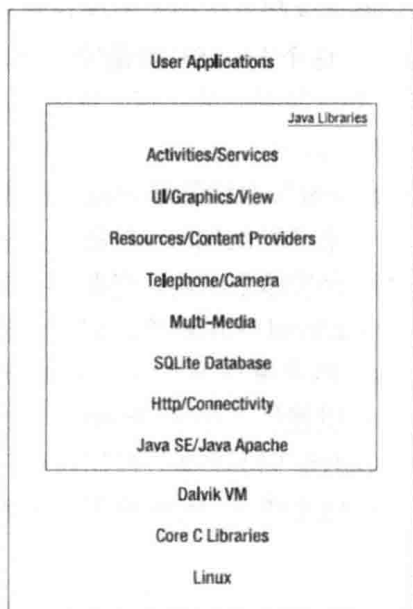


图 1-1 Android 平台的体系结构

我们采用从下往上的方式来介绍 Android 平台的体系结构的组成部分。

从图 1-1 可以看到，处在 Android 平台的底层的是 Linux 操作系统，在 Android 的平台体系结构下，Linux 提供了基础的支撑功能，包括设备管理、进程管理、文件管理等功能。按理说，Google 完全可以开发一套自己的操作系统而不采用 Linux 操作系统作为 Android 平台的支撑，可是，Linux 是如此成熟，应用又是如此广泛，并且还是开源的系统，为什么不用呢？为什么还要自己开发别人已经做得很好并且可以开源和自由使用的系统呢？这可能是 Google 使用 Linux 作为 Android 平台支撑的原因吧。

在图 1-1 中，从下往上，可以看到“Core C Libraries”，翻译为中文就是“核心 C 语言程序库”。为什么是这样呢？如前所述，Linux 是一个基础的支撑系统，可是在编写应用程序时如何使用 Linux 提供的功能呢？Linux 为了可以让应用程序使用它提供的功能而专门提供了一系列的 C 语言函数给应用程序调用，进而应用程序可以充分利用 Linux 操作系统提供的功能来达到应用程序的业务目标。作为 Android 应用开发者，除非将来要开发特定功能的应用程序，在一般情况下，读者不会直接使用到这些 C 语言函数库。当某一天要使用这个“Core C Libraries”开发 Android 应用程序时，恭喜你，你是 Android 应用开发的行家了。

图 1-1 再往上就是“Dalvik VM”，什么是 Dalvik VM？它没有那么神秘，Dalvik VM 其实就是一个变形的 Java 虚拟机（Java VM），也就是一个与普通 Java 虚拟机有一点区别的 Java 虚拟机而已。可是，Google 为什么要实现一个变形的 Java 虚拟机呢？原因很简单，因为普通的 Java 虚拟机在普通的桌面计算机上可以运行得很好，可是在小型的设备，如手机上就不一定了，为此，Google 实现了一个变形的 Java 虚拟机，并命名为 Dalvik VM。还有一个问题需要回答：为什么 Dalvik VM 需要出现在这里呢？Android 应用开发是基于 Java 语言进行的，Dalvik VM 就像我们开发桌面 Java 应用程序一样，我们需要一个 Java 虚拟机来运行开发的 Java 程序。

图 1-1 再往上就是一组用一个框框起来的“Java Libraries”，也就是一组 Java 类库。我们先来问第一个为什么：为什么 Java Libraries 会出现在这里？我们已经知道，Android 应用程序是基于 Java 语言开发的，为了开发 Java 应用程序，我们需要使用 Java 类库来完成某些基本的功能，因此，这里的“Java Libraries”就是在编写 Android 应用程序时可以使用的一组 Java 类库。第二个问题：这个 Java Libraries 中都包含了哪些 Java 类库？Java Libraries 中很多类库读者可能都不认识，也应该不认识，因为这些不认识的类库正是本书要介绍的内容，可是其中的一个读者一定认识：Java SE。是的，这个 Java SE 就是读者学习 Java 语言时的那个 Java SE，这个类库中包含像 String、Integer、File 等基础的类库，读者可以像使用 Java SE 类库中的类一样来使用其他类库中的类。

图 1-1 再往上就是“User Applications”，也就是用户应用程序，我们将来开发的所有 Android 应用程序都归属于这里。那么，如何来开发 Android 应用程序呢？这就是本书的目标。

我们从下往上介绍了 Android 平台的体系结构，现在，我们再从上往下将以上介绍的内容串起来，然后来看看如何开发一个 Android 应用程序，以及开发的 Android 应用程序是如何在 Android 设备上运行的：采用 Java 语言来开发一个 Android 应用程序（也就是一个 User Application），在开发应用程序时，可以使用“Java Libraries”中提供的 Java 类库来实现所要求的功能，当开发完成以后，可以在 Dalvik VM 上运行应用程序，Dalvik VM 会解释 Java 代码并在 Linux 操作系统上来执行代码，进而完成 Android 应用所要完成的业务功能。

1.2 Android 应用程序的组成

我们对 Android 体系有一个初步了解后，作为一个开发者，读者一定想知道 Android 是由哪些部分组成的。如下这些概念对于一个 Android 应用开发初学者来说，有些难于理解，但不要太纠结于这些概念，知道这些概念就好。我们会尽量采用通俗易懂的语言来介绍这些概念，使读者对 Android 程序的组成有一个初步的了解。

任何一个应用程序都会包括如下一些基本内容：应用程序的界面、业务功能的处理、部件之间的数据交互、数据存储。如此而已，Android 应用程序也不例外，只是被赋予了不同的名称而已。具体来说，一个 Android 应用程序包括如下的基本组成部分。

(1) Activity（窗体）。

在 Android 应用程序中，一个界面就是一个 Activity。这个名字有点与众不同，Google 经常会创造一些新名称，不是吗？既然 Activity 就是一个界面，因此，对每个 Activity，在进行设计时，都包括对界面的布局（Android 提供了丰富的 UI 组件来实现绚丽的界面）、对界面组件的点击会进行相应的事件处理等程序设计工作。可以将 Android 的 Activity 界面类比为 Internet 网页的一个页面。

(2) View（窗体组件）。

View 就是构建应用程序界面的基本组件，也就是说，Activity 界面是由一个或多个 View 构成的，例如，Button、Label、Text Field 等都是 View，View 是构建 Activity 的基本元素。

(3) Intent（窗体间或应用之间的通信组件）。

一般来说，一个 Android 应用程序会包括多个界面，用户在进行不同的操作时可能会进行不同界面之间的切换，就像在 Internet 的页面之间，当用户点击不同的页面链接时会进行不同页面之间的切换一样。在 Internet 页面之间的切换是通过链接来完成的，在 Android 的 Activity 之间实现不同的 Activity 界面之间的切换是通过称为 Intent 的对象来完成的。因此，可以这么说，Intent 是 Android 应用程序界面之间及功能部件之间实现信息交互的桥梁。

(4) Content Provider（应用之间数据交互方式）。

Content Provider，也就是内容提供者，是 Android 建议的应用程序之间进行数据交互的方式。举例来说，如果一个应用程序希望将自己的数据提供给其他应用程序使用，则该应用程序需要实现 Content Provider 接口，这样其他的应用程序便可以通过这个接口访问这个程序所提供的数据。一个典型的实现了 Android 的 Content Provider 接口的程序是通讯录程序，任何需要使用通讯录数据的程序都可以通过该接口从电话通讯录程序中获得通讯录数据。

(5) Service（无窗口的在后台默默运行的程序）。

所谓 Service 就是运行于后台的程序。一般来说，Service 程序没有用户界面，它们运行于后台并为运行在前端的程序提供服务。Android 的 Service 程序在运行方式上类似于 Windows Phone 操作系统中的后台进程：它们在安静的运行，并在需要的时候为其他程序提供服务。

(6) 广播接收器（信息广播方式）。

广播接收器，即 Broadcast Receiver。Android 平台中的程序在运行时会发生任何可能的事件，某个应用程序在运行时可能会将它的事件广播出来，其他的程序可以监听这样的事件，并对发生的事件进行必要的处理。举例来说，Android 的电池电量监视程序（这是 Android 的一个 Service）在随时监视着电池的电量，当电池的电量低于某个门槛值时，该程序会广播一个消息，而其他的应用程序可以监听这个消息，并针对这个事件做出必要的处理，例如，一个正

在进行高耗电运算的程序监听到这个消息时，应该停止进行高耗电的运算，以便减少对电量的消耗。

(7) `AndroidManifest.xml` 文件（应用程序描述文件）。

`AndroidManifest.xml` 文件是 Android 应用程序的配置文件：它将构成 Android 应用程序的各个组件有效地装配起来从而构成一个完整的 Android 应用程序。每个应用程序一定会包含一个且只能包含一个配置文件。

1.3 Android 的发展历史

手机的常用操作系统有：Symbian OS、Microsoft Windows Mobile、Mobile Linux、iOS 等，这其中没有任何一个操作系统是事实上的标准，也没有任何一个操作系统是开源的。为此，Google 创立了 Android 移动平台。

2007 年，Google 牵头建立了开放手机联盟（Open Handset Alliance），到 2009 年，这个联盟成员包括 Sprint Nextel、T-Mobile、Motorola、Samsung、Sony Ericsson、Toshiba、Vodafone、Google、Intel、Texas Instruments 等 IT 巨头，到 2011 年，成员已近 80 家，Android 已经成为移动设备事实上的行业标准。到 2018 年 8 月，Android 平台的版本已从 1.0 发展到了 9.0。Android 是进化得如此之快，因此，读者在学习 Android 开发时，必须要学会使用 Android 的在线帮助文档。

1.4 Android 应用开发环境概述

Android 平台采用 Java 语言作为应用程序开发语言。Android 开发环境包括如下基本内容：Java 基本包、Android 基础组件、Android UI 组件、Android 服务组件、Android 电话和媒体服务组件、Android 仿真器（Android Virtual Device, AVD）、Android 调试器等。

笔者将在后续章节中，对这些内容进行详细介绍。首先从建立 Android 开发环境开始讲解。

建立 Android 应用开发环境

由于 Android 应用是基于 Java 语言进行开发的，因此，首先需要安装 Java SE 环境，笔者建议安装 Java JDK 8 或以上版本。基于 Java JDK，Android 平台提供了两种建立开发环境的方式。

- (1) 使用基于 Eclipse 和 ADT 的应用开发环境；
- (2) 使用基于 Android Studio 的应用开发环境。

由于第一种方式使用 Eclipse IDE 作为开发环境，因此，对于熟悉 Eclipse 的开发者来说使用第一种方式会更为合适，第二种方式，也就是基于 Android Studio 的方式，是 Android 提供的非 Eclipse IDE 的开发环境。

由于 Google 将逐渐取消对 Eclipse 开发环境的支持，因此，我们将使用 Android Studio 作为本书的开发环境。为了测试 Android 应用程序，我们可以使用 Android SDK 自带的模拟器，也可以使用第三方的模拟器如 Genny Motion 来测试应用程序。在本书中，我们使用 Android SDK 自带的模拟器来测试程序。

2.1 下载和安装 Android Studio

在下载和安装 Android Studio 之前，需要先下载和安装 Java SDK。从 Oracle 网站下载并安装就可以了。在安装 Java JDK 之后，从 Android 的开发者网站下载 Android Studio，运行下载得到的文件即可安装 Android Studio 开发环境。

2.2 开发第一个 Android 应用程序

2.2.1 创建 Hello World 程序工程

我们已经建立了 Android 应用开发环境，现在就可以基于这个环境开发 Android 应用程序了。启动 Android Studio，即可显示如图 2-1 所示的界面。



图 2-1 Android Studio 启动界面

按照提示安装 Android Studio 开发环境即可。安装完毕后，运行 Android Studio 开发环境，单击第一个功能选项“Start a new Android Studio project”，即可创建一个新的 Android 应用程序工程，如图 2-2 所示。

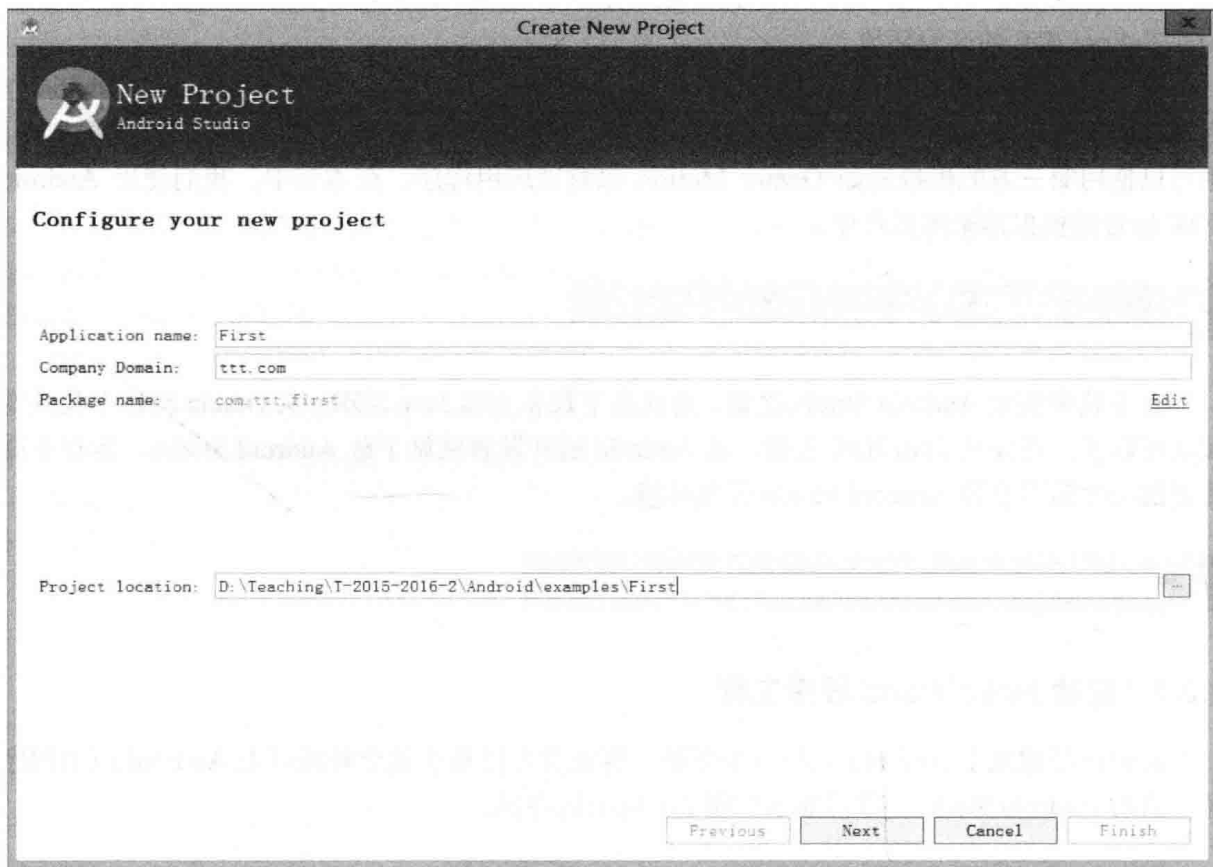


图 2-2 新建 Android 应用程序工程界面

在如图 2-2 所示的界面中，在“Application name”中，输入应用程序的名字，在“Company Domain”中输入公司的域名（可以是假想的域名），在“Project location”中输入应用程序文件存放的位置，然后单击“Next”按钮，即可显示如图 2-3 所示的界面。

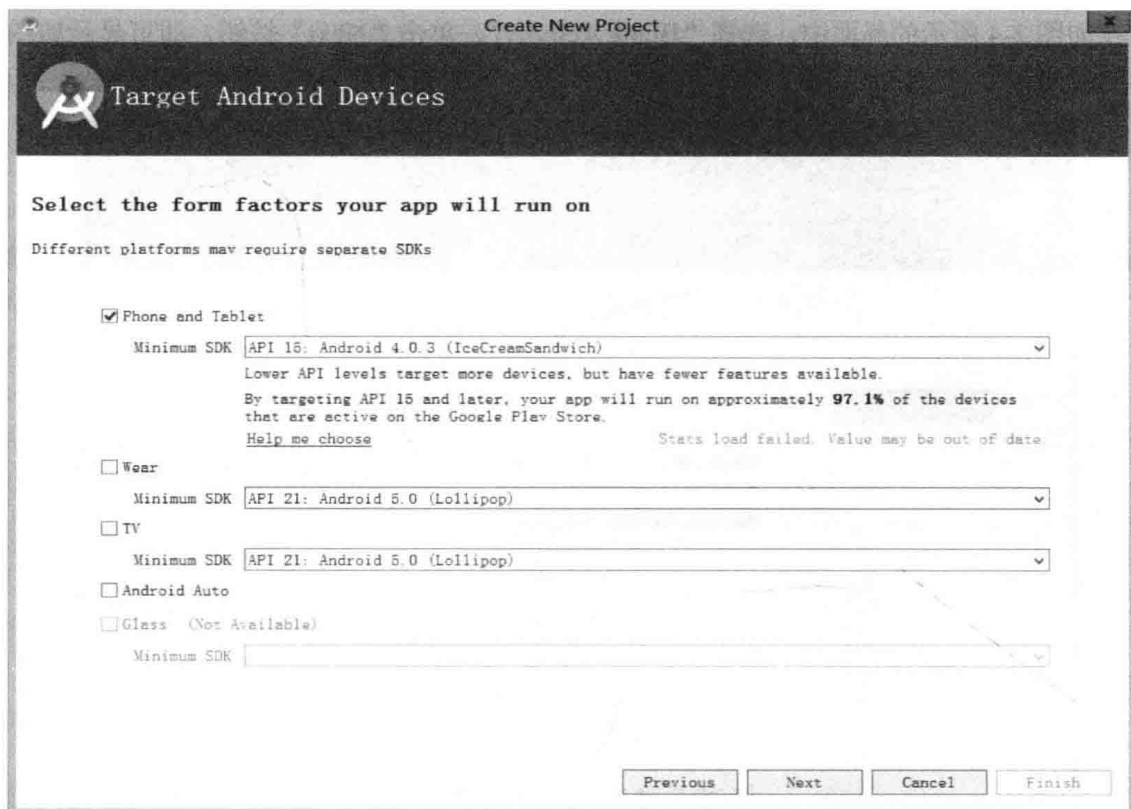


图 2-3 应用程序运行目标机设置

在如图 2-3 所示的界面中，选择应用程序运行的目标机，直接单击“Next”按钮，即可显示如图 2-4 所示的界面。



图 2-4 选择应用界面框架

在如图 2-4 所示的界面中，选择“Blank Activity”，单击“Next”按钮，即可显示如图 2-5 所示的界面。

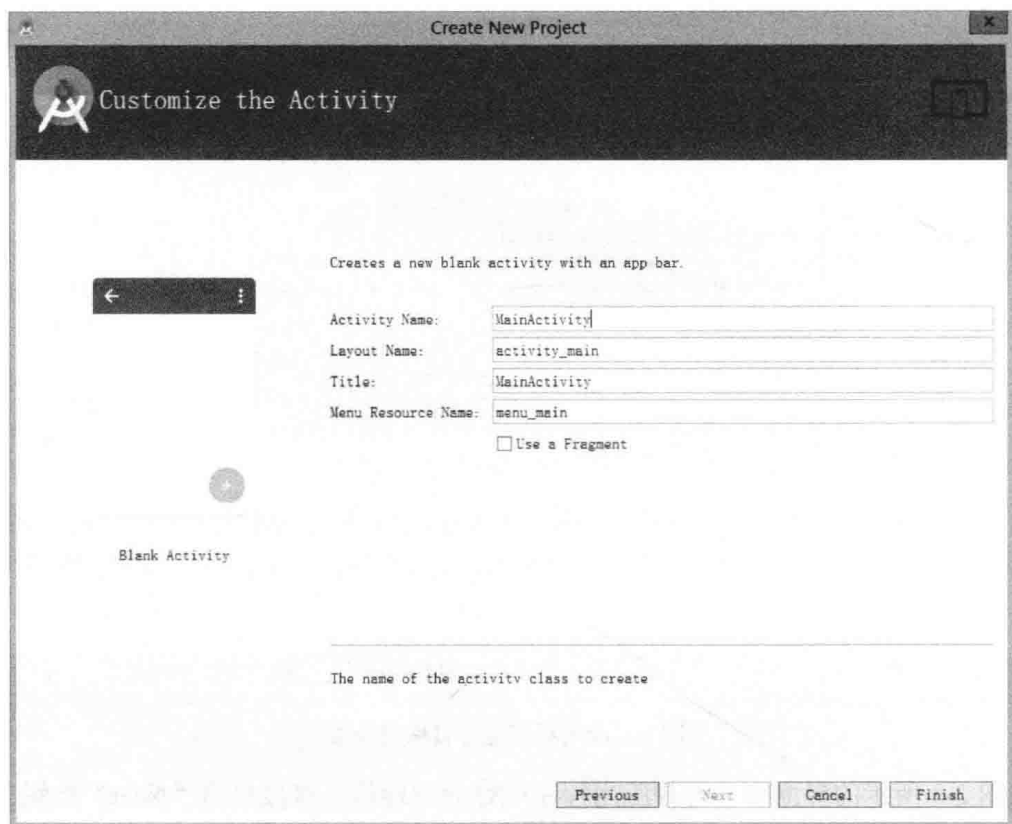


图 2-5 Android 工程概览

在如图 2-5 所示的界面中，单击“Finish”按钮，Android Studio 将创建一个新的 Android App 工程，Android Studio 创建工程文件需要一些时间，创建成功后，即可显示如图 2-6 所示的界面。

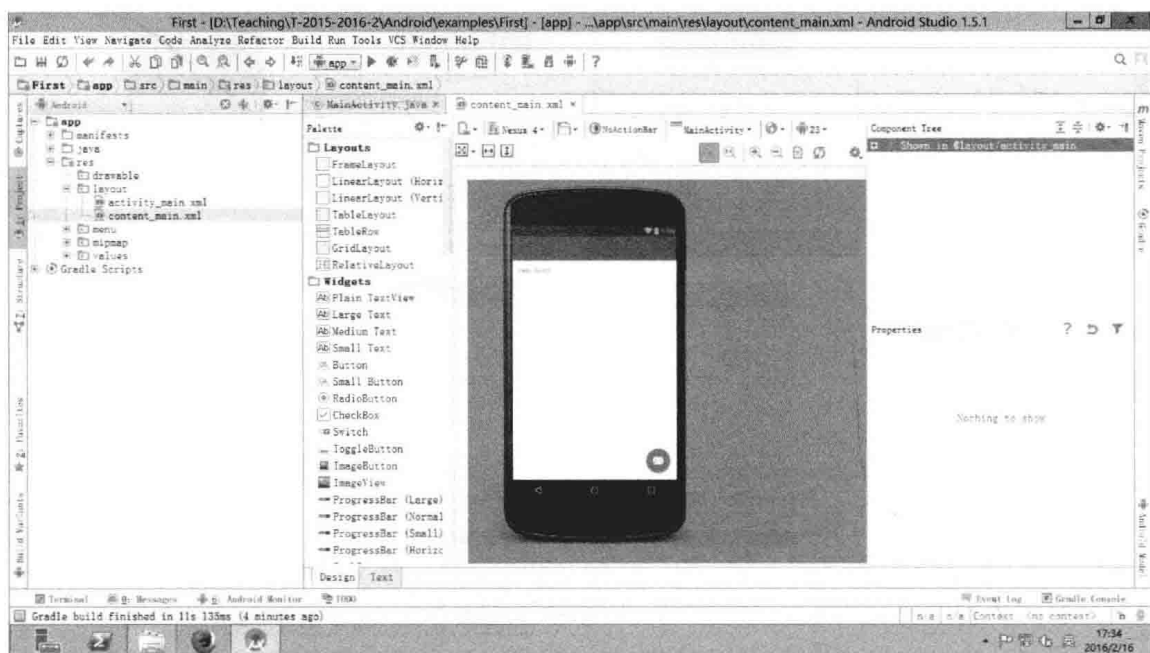


图 2-6 成功创建 Android App 工程的界面