

21世纪高等学校计算机类课程创新规划教材 · 微课版



Android 应用程序设计 (修订版)

◎ 张思民 编著

零基础
入门

项目式
教学

源码和
课件

案例演示
视频

基于
Eclipse + ADT

微课版

5 HOURS

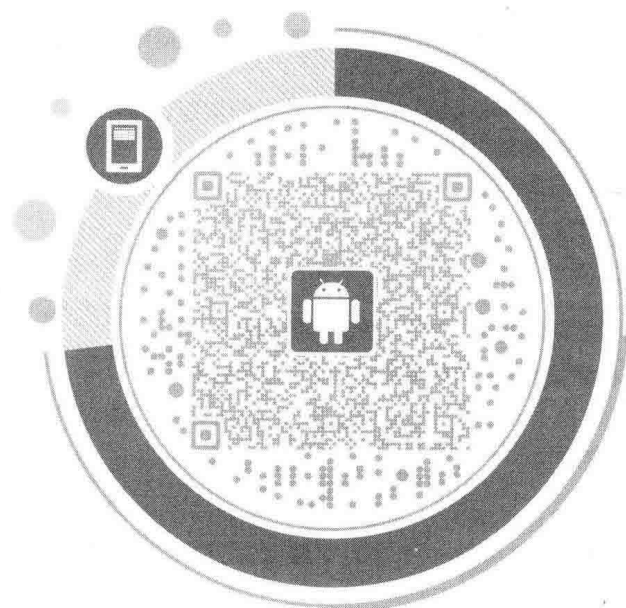
5小时
教学视频

清华大学出版社



21世纪高等学

教材 · 微课版



Android 应用程序设计 (修订版)

常州大学图书馆藏

◎ 张思民 编著

清华大学出版社
北京

内 容 简 介

本书是面向 Android 系统的初学者的入门教程，内容几乎涵盖了 Android 相关的所有技术。本书大致可以分成两个部分，第一部分（第 1~4 章）主要介绍 Android SDK 开发环境的安装、应用程序的结构、用户界面的组件及其设计方法，第二部分（第 5~10 章）主要介绍较高级的主题，内容包括异常处理与多线程、后台服务与系统服务技术、图形与多媒体处理技术、数据库技术及输入/输出流的处理技术、网络通信技术、地图服务及传感器检测技术等。

本书在叙述上浅显易懂，对每一个知识点都配了相应的例题。本书提供了所有例题的源代码、电子课件，以及本书大部分例题的视频教学演示（扫描二维码）。

本书可以作为高等院校及各类培训学校 Android 系统课程的教材，也可以作为学习 Android 程序设计的编程人员的自学用书。

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。

版权所有，侵权必究。侵权举报电话：010-62782989 13701121933

图书在版编目 (CIP) 数据

Android 应用程序设计 / 张思民编著. —修订版. —北京：清华大学出版社，2018
(21 世纪高等学校计算机类课程创新规划教材·微课版)
ISBN 978-7-302-49578-9

I. ①A… II. ①张… III. ①移动终端—应用程序—程序设计 IV. ①TN929.53

中国版本图书馆 CIP 数据核字 (2018) 第 027488 号

责任编辑：魏江江

封面设计：刘 键

责任校对：徐俊伟

责任印制：杨 艳

出版发行：清华大学出版社

网 址：<http://www.tup.com.cn>，<http://www.wqbook.com>

地 址：北京清华大学学研大厦 A 座 邮 编：100084

社总机：010-62770175 邮 购：010-62786544

投稿与读者服务：010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈：010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载：<http://www.tup.com.cn>, 010-62795954

印 刷 者：北京富博印刷有限公司

装 订 者：北京市密云县京文制本装订厂

经 销：全国新华书店

开 本：185mm×260mm 印 张：19.75 字 数：481 千字

版 次：2013 年 3 月第 1 版 2018 年 5 月第 2 版 印 次：2018 年 5 月第 1 次印刷

印 数：1~1500

定 价：39.50 元

产品编号：078298-01

前 言

Android 系统自 2007 年推出以来，应用越来越广泛。除了手机、平板电脑使用 Android 系统之外，其他嵌入式系统也大量使用 Android 系统来设计。例如，车载设备、医疗设备、VoIP 电话和智能电视等厂商纷纷推出 Android 系统产品。可以说，Android 系统如日中天，相信将来还会有更好的发展。

1. 本书特点

作为一本教材，本书有以下特点。

(1) 易学易懂。本书面向 Android 系统的初学者，在叙述方式上浅显易懂，摒弃枯燥的理论，尽可能使用图示加以说明。对每一个知识点，都配了相应的例题。所有例题均短小精悍，适合课堂教学讲授。读者学完每一章内容后都可以编写出相应功能的程序。

(2) 解释详细。对每一个例题，均进行了详细分析和解释，既可以帮助读者学习理解知识和概念，大大降低学习难度，又具有启发性。

(3) Java 语言零基础学习。为了帮助没有 Java 语言基础的读者学习 Android 系统，特别安排了一章介绍 Java 基础知识的内容。

(4) 配有视频教学演示。书中大部分例题均录制了教学视频，详细地记录了设计的操作过程，帮助读者更加轻松、迅速地理解和掌握本书内容。

2. 学习方法

学习 Android 程序设计，应该循序渐进、由浅入深，不能跳跃式地进行，前面的内容还没搞清楚，就急于学习后面的内容，这样只会事倍功半，欲速则不达。

应该说，学习任何一种编程技术都会有一定难度。因此，要强调动手实践，多编程、多练习，熟能生巧，从学习中体验到程序设计的乐趣和成功的喜悦，增强学习信心。

3. 本书内容

本书在内容结构上大致可以分成两个部分。

第一部分（第 1~4 章）主要介绍 Android SDK 开发环境的安装、应用程序的结构、用户界面的组件及其设计方法，该部分内容是学习 Android 程序设计的入门基础。

第 1 章主要讲解 Android SDK 开发环境的安装，并说明如何下载 Android SDK 和如何从头开始创建新的应用程序。第 2 章简要介绍 Java 语言基础知识，为不熟悉 Java 语言的读者提供帮助，对于已有 Java 语言基础的读者，可以跳过本章。第 3~4 章讲解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及多用户界面程序的开发。

第二部分（第 5~10 章）主要介绍较高级的主题，内容包括异常处理及多线程、图形与多媒体处理技术、后台服务与系统服务技术、数据库技术及输入/输出流的处理技术、网

络通信技术、地图服务及传感器检测技术等。

第 5 章讲解 Android 的异常处理方法以及多线程。第 6 章讲解图形与多媒体处理技术,介绍了绘制几何图形的的基本方法、处理触摸屏事件的方法,还详细讨论了音频播放和视频播放的设计,以及录音、照相和文本转换语音技术,最后详细讲解了如何处理图像的缩放、变形、颜色等数字图像处理技术。第 7 章讲解后台服务与系统服务,以及系统功能调用。第 8 章讲解数据存储技术,介绍了 SQLite 数据库存储方式、文件存储方式和 XML 文件的 SharedPreferences 存储方式。第 9 章讲解网络通信,介绍了 Socket 套接字编程、基于 Web 编程和与 JavaScript 脚本交互的编程技术,以及无线网络通信技术 WiFi 的程序设计方法。第 10 章讲解地图服务及传感器检测技术,地图服务主要介绍地图查询和贴图的方法,传感器检测主要介绍重力加速度的应用。

书中所有例题均已在 Eclipse + ADT 环境下运行通过。本书提供了所有例题的源代码、电子课件。

参加本书编写、校对及程序测试工作的还有梁维娜、张静文、杨军民、颜敏敏等,在此表示感谢。

由于编者水平有限,书中难免有不足之处,敬请读者批评指正。

编 者
2018 年 1 月

图书资源支持

感谢您一直以来对清华版图书的支持和爱护。为了配合本书的使用,本书提供配套的资源,有需求的读者请扫描下方的“书圈”微信公众号二维码,在图书专区下载,也可以拨打电话或发送电子邮件咨询。

如果您在使用本书的过程中遇到了什么问题,或者有相关图书出版计划,也请您发邮件告诉我们,以便我们更好地为您服务。

我们的联系方式:

地址:北京海淀区双清路学研大厦 A 座 707

邮编:100084

电话:010-62770175-4604

资源下载:<http://www.tup.com.cn>

电子邮件:weijj@tup.tsinghua.edu.cn

QQ: 883604(请写明您的单位和姓名)

用微信扫一扫右边的二维码,即可关注清华大学出版社公众号“书圈”。

资源下载、样书申请



书圈

目 录

第 1 章 Android 系统及其开发过程	1
1.1 Android 系统概述	1
1.2 安装 Android SDK 开发环境	2
1.2.1 安装 Android SDK 前必要的准备	2
1.2.2 安装 Android SDK 详解	3
1.2.3 设置环境变量	7
1.3 Android API 和在线帮助文档	7
1.4 Android 应用程序的开发过程	8
1.4.1 开发 Android 应用程序的一般过程	8
1.4.2 生成 Android 应用程序框架	9
1.4.3 编写 MainActivity.java	10
1.4.4 配置应用程序的运行参数	11
1.4.5 在模拟器中运行应用程序	12
1.5 Android 应用程序结构	12
1.5.1 目录结构	12
1.5.2 Android 应用程序架构分析	18
1.6 Android 应用程序设计示例	19
习题 1	21
第 2 章 Java 语法概述	22
2.1 语法基础	22
2.1.1 数据类型	22
2.1.2 常量与变量	23
2.1.3 对变量赋值	24
2.1.4 关键字	24
2.1.5 转义符	24
2.2 基本数据类型应用示例	25
2.2.1 整型与浮点型	25
2.2.2 字符型	27
2.2.3 布尔型	28
2.2.4 数据类型的转换	29
2.3 程序控制语句	30
2.3.1 语句的分类	30

2.3.2	顺序控制语句	31
2.3.3	if 语句	32
2.3.4	switch 语句	34
2.3.5	循环语句	35
2.3.6	转语句	40
2.4	类与对象	42
2.4.1	类的定义	42
2.4.2	对象	44
2.4.3	接口	46
2.4.4	包	47
2.5	XML 语法简介	47
	习题 2	51
第 3 章	Android 用户界面设计	53
3.1	用户界面组件包 widget 和 View 类	53
3.2	文本标签与按钮	54
3.2.1	文本标签	54
3.2.2	按钮	56
3.3	文本编辑框	61
3.4	Android 布局管理	64
3.4.1	布局文件的规范与重要属性	64
3.4.2	常见的布局方式	65
3.5	进度条和选项按钮	72
3.5.1	进度条	72
3.5.2	选项按钮	74
3.6	图像显示与画廊组件	80
3.6.1	图像显示 ImageView 类	80
3.6.2	画廊组件 Gallery 与图片切换器 ImageSwitcher	84
3.7	消息提示	88
3.8	列表组件	91
3.8.1	列表组件 ListView 类	91
3.8.2	列表组件 ListActivity 类	94
3.9	滑动抽屉组件	96
	习题 3	100
第 4 章	多个用户界面的程序设计	102
4.1	页面切换与传递参数值	102
4.1.1	传递参数组件 Intent	102
4.1.2	Activity 页面切换	102
4.1.3	应用 Intent 在 Activity 页面之间传递数据	106
4.2	菜单	110

4.2.1	选项菜单	110
4.2.2	上下文菜单	112
4.3	对话框	114
4.3.1	消息对话框	114
4.3.2	其他几种常用对话框	120
习题 4		123
第 5 章	异常处理与多线程	124
5.1	异常处理	124
5.2	多线程	126
5.2.1	线程与多线程	126
5.2.2	线程的生命周期	127
5.2.3	线程的数据通信	128
5.2.4	创建线程	130
习题 5		137
第 6 章	图形与多媒体处理	138
6.1	绘制几何图形	138
6.1.1	几何图形绘制类	138
6.1.2	几何图形的绘制过程	139
6.2	触摸屏事件处理	144
6.2.1	简单触摸屏事件	144
6.2.2	手势识别事件	150
6.3	音频播放	153
6.3.1	多媒体处理包	153
6.3.2	媒体处理播放器	154
6.3.3	播放音频文件	155
6.4	视频播放	161
6.4.1	应用媒体播放器播放视频	161
6.4.2	应用视频视图播放视频	164
6.5	录音与拍照	166
6.5.1	用于录音、录像的 MediaRecorder 类	166
6.5.2	录音示例	167
6.5.3	拍照	170
6.6	将文本转换成语音	176
6.7	图像处理技术	178
6.7.1	处理图像的颜色矩阵	178
6.7.2	处理图像的坐标变换矩阵	184
习题 6		192
第 7 章	后台服务与系统服务技术	193
7.1	后台服务 Service	193

7.2	信息广播机制 Broadcast	197
7.3	系统服务	206
7.3.1	Android 的系统服务	206
7.3.2	系统通知服务 Notification	206
7.3.3	系统定时服务 AlarmManager	209
7.3.4	系统功能的调用	212
	习题 7	215
第 8 章	数据存储	216
8.1	SQLite 数据库	216
8.1.1	SQLite 数据库简介	216
8.1.2	管理和操作 SQLite 数据库的对象	217
8.1.3	SQLite 数据库的操作命令	218
8.2	文件处理	230
8.2.1	输入流和输出流	230
8.2.2	处理文件流	231
8.3	轻量级存储 SharedPreferences	237
	习题 8	240
第 9 章	网络通信	241
9.1	网络编程的基础知识	241
9.1.1	IP 地址和端口号	241
9.1.2	套接字	244
9.2	基于 TCP 的网络程序设计	246
9.3	基于 HTTP 的网络程序设计	251
9.4	Web 视图	255
9.4.1	浏览器引擎 WebKit	255
9.4.2	Web 视图对象	255
9.4.3	调用 JavaScript	258
9.5	无线网络通信技术 WiFi	266
	习题 9	272
第 10 章	地图服务及传感器检测技术	273
10.1	Google 地图	273
10.1.1	Google Maps 包	273
10.1.2	导入 Google 地图 API 的 Maps 包	274
10.1.3	显示地图 MapView 类	274
10.1.4	添加 Google 地图的贴图	279
10.2	位置服务	282
10.3	传感器检测技术	286
10.3.1	传感器简介	286

10.3.2 加速度传感器的应用示例	289
习题 10	297
附录 A JavaSDK 及 Eclipse 的安装与配置	298
附录 B Android 的调试工具	300
附录 C Map API Key 的申请过程	303

1.1 Android 系统概述

2007 年 11 月 5 日, Google 公司推出了基于 Linux 操作系统的智能手机平台 Android 系统。Android 系统由操作系统、中间件、用户界面程序和应用软件等组成。为了推动 Android 系统发展, Google 与手机制造商、电信运营商、半导体公司、软件公司等 65 家企业联手组成了一个商业联盟组织——OHA (Open Handset Alliance, 开放手机联盟), 制定了基于 Android 的移动设备生产和开发标准。

Android 的出现绝非偶然, 它是由传统的移动电话系统开发模式演变而来的一种符合时代潮流的新型移动开发模式的产物。Android 的出现为移动开发者带来了新的机遇与挑战。

移动电话的开发经历了传统移动电话的开发、半开放式移动电话的开发、全开放式移动电话的开发 3 个发展阶段。

(1) 传统移动电话的开发: 移动电话厂商制作移动电话出售, 厂商有自己的研发机构, 也依靠其他公司提供的解决方案来完成移动电话的开发工作。通俗点说, 就是买了移动电话, 里面的功能已经固定, 没有扩展的机会。

(2) 半开放式移动电话的开发: 随着自定义需求的增加, 移动开发走向了半开放模式。在这种模式下, 厂商制造移动电话出售, 预置了部分基本软件功能, 但是支持增加第三方应用程序, 用户可以根据自己的需要选择下载安装。在这种模式下, 第三方应用程序开发接口是开放的, 但是系统本身是不开放的, 因此只能称为半开放模式。

(3) 全开放式移动电话的开发: Android 的出现, 是全开放开发模式的缩影, 不仅第三方应用程序接口开放, Android 系统本身也是完全开放的。各个厂商在统一的平台上开发移动电话, 第三方开发移动应用。如果系统不能满足需求, 则可以在系统中增加新的功能, 这就是全开放的优势。

移动电话经过 20 年的发展, 已经不仅仅是一个移动的通信工具, 随着 3G 技术的发展, 移动电话正向着智能化的方向迈进, 移动电话正逐渐成为多种应用工具的功能载体, 如通信工具、网络工具、媒体播放器、媒体获取设备、多类型的连接设备、信息感知终端、视频电话、个性化定制平台等。

Android 系统诞生在开放时代的背景下, 其全开放的智能移动平台、多硬件平台的支持、使用众多标准化的技术、核心技术完整、完善的 SDK 和文档、完善的辅助开发工具等特点与智能手机的发展方向紧密相连, 它将代表并引领新时代的技术潮流。

对于开发者而言, Android 开发分为两大类:

(1) 移植开发移动电话系统。移植开发是为了将 Android 系统运行在手持式移动设备上, 在具体的硬件系统上构建 Android 软件系统。这种类型的开发在 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。

(2) Android 应用程序开发。应用程序开发可以基于硬件设备(用于测试的实体手机), 也可以基于 Android 模拟器。应用开发处于 Android 系统的顶层, 使用 Android 系统提供的 Java 框架(API) 进行开发设计工作, 是大多数开发者从事的开发工作。本书所介绍的 Android 应用程序设计, 都是在这个层次上进行的。



视频演示

1.2 安装 Android SDK 开发环境

1.2.1 安装 Android SDK 前必要的准备

1. Android 系统开发的操作平台与软件环境要求

目前, 对 Android 系统开发支持的操作系统有:

- Windows 7/8/10 (32 位或 64 位)
- Mac OS X10 或更高版本
- Linux (GNOME 或 KDE 桌面)

本书讲解主要以 Windows 系统为例, 其他操作系统下的 Android 开发请参考 Android 的开发文档。

对于 Android 系统开发的软件环境, 在这里主要介绍 Eclipse+ADT (Android Development Tools 插件)。因此, 需要安装 Java 8 以上和 Eclipse 3.3 以上版本的环境。有关 Java SDK 以及 Eclipse 的安装与配置请参考附录。

2. 下载最新版本的 Android SDK 软件

这里以 2017 年 4 月 Google 发布的移动操作系统平台的版本 Android 8.0 为例, 其面向开发人员的 SDK 版本 (android-sdk_r26.0.2) 也同步推出 (截止到本书校对书稿时, Google 发布的最新版本为 android-sdk_r26.0.2)。

可以到 Android 官方网站 <http://developer.android.com/sdk/index.html> 下载最新的系统软件, 见表 1-1 (以版本 Android 8.0 为例)。

表 1-1 下载 Android 系统安装包

安 装 平 台	系统安装包	Size
Windows	android-sdk_r26.0.2-windows.zip	90 379 469 bytes
	installer_r26.0.2-windows.exe (推荐)	70 495 456 bytes
Mac	android-sdk_r26.0.2-macosx.zip	58 218 455 bytes
Linux	android-sdk_r26.0.2-linux.tgz	82 616 305 bytes

1.2.2 安装 Android SDK 详解

1. 运行 Android SDK 安装文件

解压安装包文件 sdk-tools-windows-r26.zip, 得到如图 1.1 所示的系统安装框架, 然后运行其中的 tools\bin\ sdkmanager 来安装 Android SDK。

2. 运行 SDK Manager.exe 文件

在命令行窗口, 进入 tools\bin 目录, 输入安装命令:

```
sdmanager "platform-tools" "platforms;android-26"
```

也可以在 Eclipse 中运行 SDK Manager 项, 弹出如图 1.2 所示的 SDK 管理窗口, 系统自动搜索所有版本的系统安装包。单击位于窗口下方的 Install Packages 按钮, 系统自动下载并安装这些选中的包, 该过程需要比较长的时间。



图 1.1 Android SDK 系统安装框架

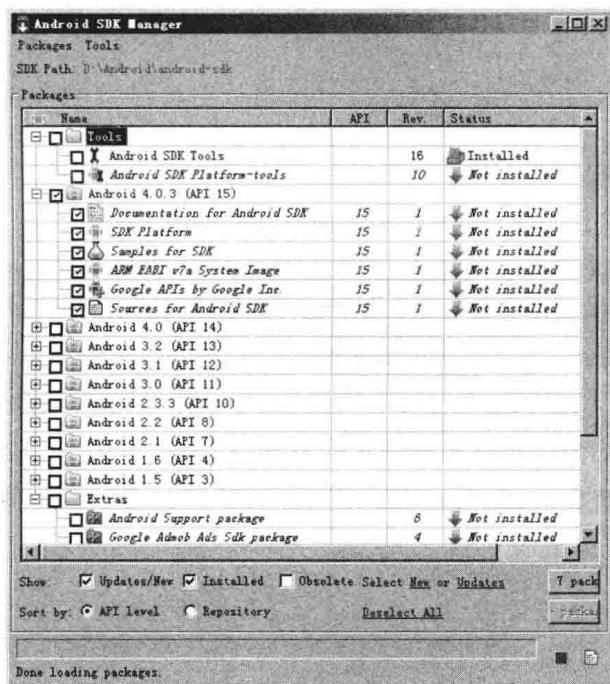


图 1.2 Android SDK 的管理窗口

Android SDK 系统在安装完之后会提示是否安装 ADB (Android Debug Bridge)。ADB 是开发 Android 应用项目的调试工具, 这里要确认安装。在 Android SDK 所有系统文件安装完之后, 打开安装目录, 其目录结构如图 1.3 所示。

目录结构中主要文件夹的作用如下。

- add-ons: 放置 Google 提供的 API 包, 包括 Google 地图 API 等。
- docs: 放置 Android 系统的帮助文档和说明文档。
- platforms: 针对每个 SDK 版本提供与其相对应的 API 包。
- tools 和 platform-tools: 放置通用的工具文件, 如 Android 模拟器 AVD、SQLite 数

据库、调试工具 ADB、创建模拟的 SD 卡工具 mkshcard 等。为了方便地使用这些工具，通常将它们设置为系统环境变量。

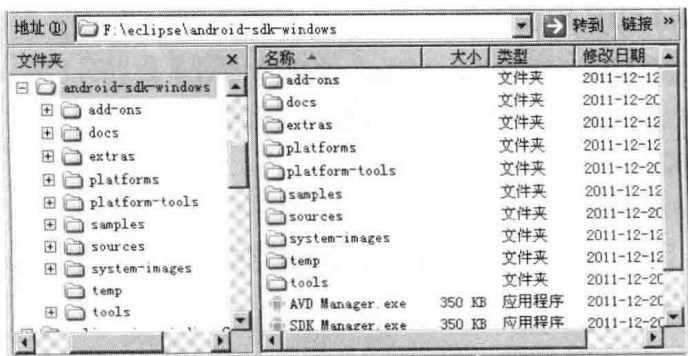


图 1.3 安装 Android SDK 系统完毕后的目录结构

- samples: 放置每个 SDK 版本提供的示例程序。
- system-images: 由于 Android 是基于 Linux 的系统，该文件夹中放置了不同版本的 img 系统映像文件。

3. 安装 Android 开发工具 ADT

Android 为 Eclipse 开发环境提供了一个外挂插件 ADT (Android Development Tools)，该插件为用户提供了一个强大的综合环境用于开发 Android 应用程序。ADT 扩展了 Eclipse 的功能，可以让用户快速地建立 Android 项目，创建应用程序界面，在基于 Android 框架 API 的基础上添加组件，以及用 SDK 工具集调试应用程序。

下面详细介绍安装和配置 ADT 的基本方法和步骤。

(1) 打开 Eclipse 设置工作目录。在安装 Android 插件 ADT 之前必须已经安装好 Eclipse 集成开发环境，如果尚未安装，可参照附录 A 自行安装。启动 Eclipse，第一次启动时会要求用户设置工作目录，如图 1.4 所示。作者建立了一个 D:\AdroidTest\android-SDK 工作目录，读者可根据自己的需要设置相应的工作目录。

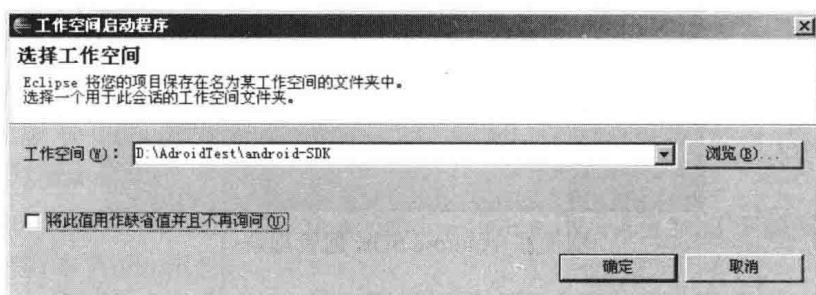


图 1.4 设置 Android 工作目录

(2) 安装 ADT 插件。在 Eclipse 中，选择“帮助”(Help) → “安装新软件”(Install New Software) 命令，在弹出的 Install 对话框中单击 Add 按钮，然后在 Add Repository 对话框的 Location 文本框中输入 <http://dl-ssl.google.com/android/eclipse/>，单击“确定”按钮，如图 1.5 所示。

(3) 设置 ADT 的首选项。顺利安装 ADT 后，在 Eclipse 中，选择“窗口”(Window) → “首选项”(Preferences) 命令，弹出“首选项”对话框，在 SDK Location 文本框中设置

安装 Android SDK 的绝对路径, 如图 1.6 所示。

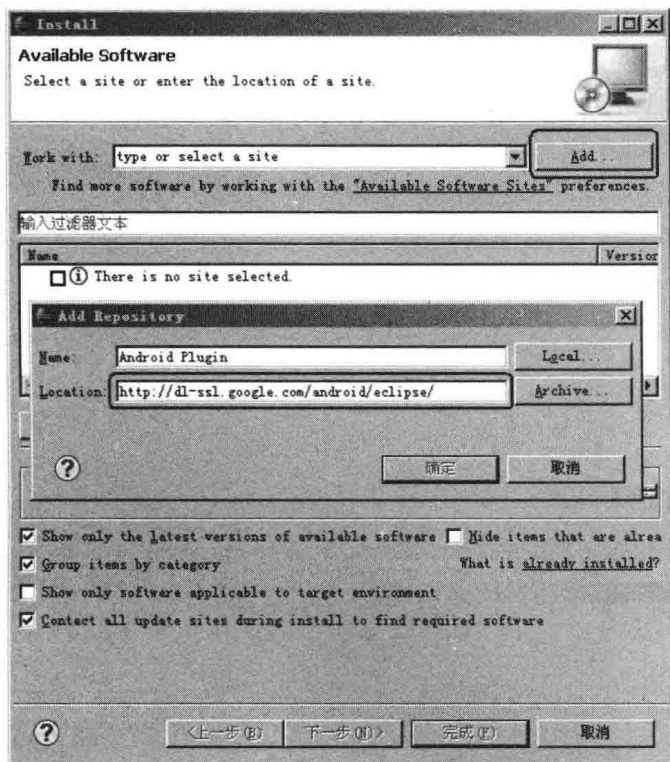


图 1.5 在 Eclipse 中安装 ADT 插件

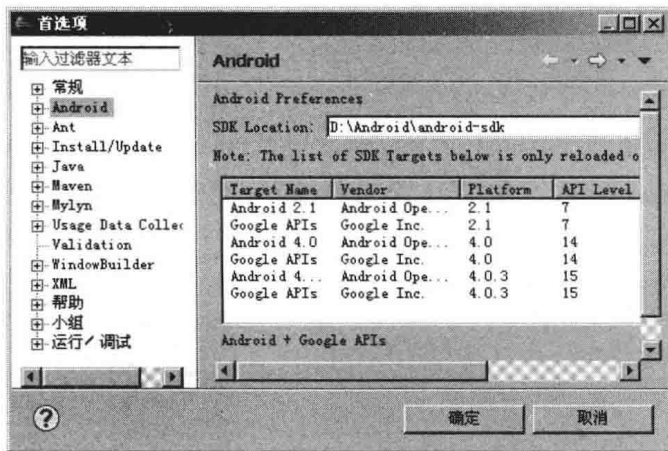


图 1.6 设置 ADT 的首选项

4. 创建 Android 虚拟设备 AVD

Android 应用程序可以在实体手机上执行, 也可以创建一个 Android 虚拟设备 AVD (Android Virtual Device) 来测试。每一个 Android 虚拟设备 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 卡等。

下面介绍如何创建一个 Android 虚拟设备 AVD。

Eclipse 集成开发环境提供了 Android SDK and AVD Manager 功能, 读者可以用它来创建 Android 虚拟设备 AVD。

(1) 选择 Eclipse 中的“窗口”(Window) → AVD Manager 命令, 在弹出的 Android Virtual

Device Manager 对话框中可以看到已创建的 AVD。单击右边的 New 按钮创建一个新的 AVD, 如图 1.7 所示。

(2) 在弹出的 Create new Android Virtual Device (AVD) 对话框中, 输入或选择如图 1.8 所示的各项内容, 单击 Create AVD 按钮, 创建一个新的 AVD。

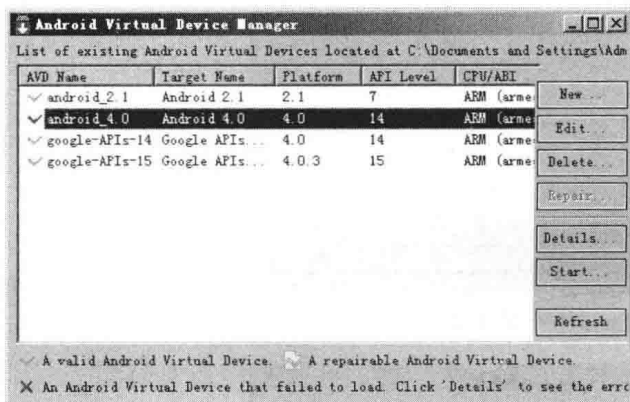


图 1.7 Android Virtual Device Manager 对话框

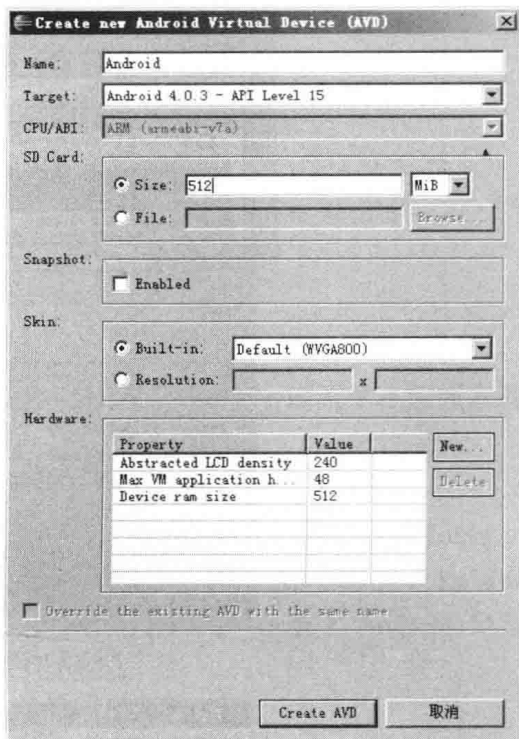


图 1.8 创建新的 AVD

(3) 运行 AVD 模拟器。在如图 1.7 所示的 Android Virtual Device Manager 对话框中选择已经建立的 AVD, 单击 Start 按钮, 可以启动 AVD 模拟器。启动 AVD 模拟器的过程时间很长, 建议打开后不要关闭, 可以在该模拟器上测试 Android 应用程序。启动的 AVD 模拟器如图 1.9 所示。



图 1.9 Android 的 AVD 模拟器