

整合Android Studio和Eclipse两种开发环境，提供全书所有源代码，涉及访问网络摄像头、语音识别、媒体动画、网络开发、近距离无线通信、访问智能电表或智能插座等

新平台 + 新版本 + 新结构 + 新内容 + 新项目 + 免费资源



新编

Android

应用开发从入门到精通

何福贵 / 等编著

ANDROID

面向系统集成的新发展
提供完整可运行源代码

Android Studio 新环境
覆盖安卓开发各个环节



多媒体动画、界面设计
权限、系统类及数据库

WiFi、蓝牙和NFC技术
应用程序的托管及发布



机械工业出版社
CHINA MACHINE PRESS

本书作者

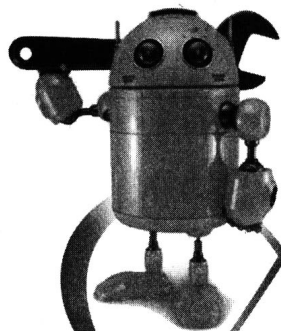


何福贵：博士后、博导、北京市教学名师、优秀专业创新团队带头人。2010年至今指导学生参加全国职业技能竞赛获得一等奖4项、二等奖5项，2017年入选“高创计划”教学名师。曾出版专著十余本，撰写发表论文几十篇，涉及EI检索论文3篇、国际会议论文5篇、核心期刊4篇，以及教育改革十余篇等。

本书特色

本书基于最新Android Studio开发平台和最新的Android SDK，以Android项目开发的视角，循序渐进地讲解了Android项目开发过程的主要内容，依次介绍了开发环境的搭建、项目设计、界面设计、应用程序构成设计、高级界面设计、数据持久化方案、多媒体应用开发、网络开发、无线通信、开源库和开源项目，以及应用程序的托管和发布等内容。在讲解每一个知识时，都遵循了理论联系实际的教学方式，配以实战演练，彻底剖析了Android项目开发的完整实现流程。





新编

Android

应用开发从入门到精通

何福贵 / 等编著

ANDROID



机械工业出版社
CHINA MACHINE PRESS

前言

本书基于当前最新的 Android Studio 2.3 版本和 Android SDK，从 Android 发展的前沿角度出发，展示了 Android 开发的最新相关知识内容。通过本书的学习，您将掌握实用的移动终端开发基础知识和应用技能，精通 Android 项目开发技术，从而能够胜任应用程序的实际开发任务，为培养综合应用能力铺平了道路。

全书共 12 章，以 Android 项目开发的视角，循序渐进地讲解并展示了 Android 项目开发过程的主要流程，具体如下。

第 1 章介绍了 Android 的开发环境，包括两种环境的搭建方法，以及 Eclipse 项目到 Android Studio 项目的转化方法，完成开发前的准备工作。

第 2 章介绍了 Android 软件项目开发的整体流程及 Android 开发过程中的代码规范，让读者对 Android 项目开发形成整体的了解。

第 3 章介绍了 Android 界面设计，包括布局、控件和 Activity，以及新的设计方法。

第 4 章对 Android 应用程序的各组成部分进行了深入讲解，包括事件处理机制、Android 多线程、Android 广播组件、后台服务 Service、AsyncTask、Handler 等。

第 5 章针对 Android 界面的设计，介绍了一些更复杂和高级的界面设计方法，包括 Android 的一些新控件的使用方法。通过本章的学习，读者将能够设计出更美观的界面。

第 6 章对 Android 常用的数据持久化方案进行了详细讲解，包括 SharedPreferences 存储、SQLite 数据库操作和最新的 LitePal 数据库操作等。

第 7 章介绍了与 Android 相关的动画技术，包括绘图动画、Drawable 动画、矢量动画等基本的图形类和二维动画，以及 Open GL ES 三维动画。

第 8 章介绍了 Android 音视频的操作方法，包括 Android 系统类的实现方法，并介绍了被 Android 开发者广泛应用的基于 FFmpeg 开发并开源的轻量级视频播放器 Ijkplayer。

第 9 章介绍了 Android 的权限机制，讲解了 JSON 格式数据的构造和解析方法。

第 10 章介绍了 Android 目前应用最广泛的无线通信技术，包括 WiFi、蓝牙和 NFC。不光展示了这三种技术的应用方法，还提供了对应的实际项目。

第 11 章介绍了 Android 的开源库和开源项目，包括一些典型 Android 开源库的获取和使用方法，一些典型 Android 开源项目的功能，以及获取 Android 开源资源的方法。

第 12 章介绍了应用程序的托管和发布方法。

总体来说，本书具有如下特点。

(1) 面向项目。按照实际项目的特点进行编写，以项目为主线进行内容讲解。

(2) 面向前沿。立足于 Android 发展的前沿角度，使用最新的开发环境。

(3) 有序分类。对知识进行了科学编排，使每一章既具有独立性，整体上又具有完整性。

本书由何福贵主要编写，其他参与编写人员还包括闫秀珍、何小波等。由于编写时间仓促，作者水平有限，书中疏漏和错误之处在所难免，望广大专家、读者提出宝贵意见。

目 录

前 言

第一章 Android 开发环境

1.1 Android 开发环境简介	1	1.3.4 安装 Android Studio 新插件	26
1.2 基于 Eclipse 的开发环境	1	1.3.5 详解项目中的资源	30
1.2.1 开发环境的搭建	2	1.3.6 详解 build.gradle 文件	33
1.2.2 项目结构	10	1.3.7 项目运行	36
1.2.3 使用第三方库	12	1.3.8 导入 Eclipse 项目	43
1.2.4 项目的运行和调试	13	1.3.9 导入 JAR 文件	45
1.3 基于 Android Studio 的开发环境	15	1.3.10 调试	47
1.3.1 Android Studio 的特点	15	1.4 两种开发环境的比较和应用程序 转化	55
1.3.2 搭建 Android Studio 应用开发 环境	16	1.5 本章小结	57
1.3.3 Android Studio 2.3 的新特性	23		

第二章 Android 开发基础知识

2.1 总体流程	58	2.3.3 变量和常量命名方法	61
2.2 各阶段描述	58	2.3.4 方法的命名方法	61
2.3 Android 开发代码规范	60	2.3.5 注释规范	61
2.3.1 项目和包命名规范	60	2.4 本章小结	63
2.3.2 类和接口命名方法	60		

第三章 应用程序用户接口——界面设计

3.1 用户界面设计基础	64	3.3 布局内部构成——界面控件	82
3.2 界面最外层设计——布局	66	3.4 界面设计助手——辅助设计工具	85
3.2.1 简单布局——常用布局	66	3.5 Android 新控件	86
3.2.2 百分比布局	72	3.6 界面背后的劳动者——Activity	89
3.2.3 复杂布局——布局嵌套	74	3.6.1 Activity 简介	89
3.2.4 Android 新布局 ConstraintLayout	79	3.6.2 创建 Activity 和加载布局	90



3.6.3 Activity 的生命周期·····	92	3.7.1 什么是 Material Design ·····	97
3.6.4 使用 Intent 在 Activity 之间穿梭·····	93	3.7.2 Material Design 内容 ·····	98
3.6.5 Intent 调用常见系统组件 ·····	95	3.8 实例: WebView 实现监控界面 ·····	98
3.7 界面设计新体验——Material Design ···	97	3.9 本章小结 ·····	102

第四章 应用程序的构成部件

4.1 应用程序架构介绍 ·····	103	4.4.2 服务的实现 ·····	116
4.2 应用程序并行机制——线程和 线程池 ·····	105	4.4.3 实现 Service 和 Activity 之间 通信 ·····	118
4.2.1 线程的实现方法 ·····	105	4.5 应用程序的消息处理机制— Handler ·····	122
4.2.2 Android 的线程池 ·····	106	4.5.1 Handler 类 ·····	122
4.3 应用程序互动机制——事件机制 ·····	109	4.5.2 实例: 获取当前时间 ·····	124
4.3.1 事件处理机制 1——基于监听器的 事件处理 ·····	109	4.6 应用程序轻量级并行——AsyncTask 机制 ·····	126
4.3.2 事件处理机制 2——基于回调的事件 处理 ·····	110	4.6.1 AsyncTask 抽象类 ·····	126
4.3.3 事件响应的实现 ·····	110	4.6.2 实例: 实现定时器 ·····	127
4.3.4 实例: 获取触点坐标 ·····	111	4.7 AsyncTask 和 Handler 两种异步方式 比较 ·····	129
4.4 应用程序后台劳动者——Service ·····	114	4.8 本章小结 ·····	130
4.4.1 服务的创建 ·····	114		

第五章 界面设计更进一步——UI 高级设计

5.1 自定义控件 ·····	131	实现拍照 ·····	158
5.1.1 自定义 View 类控件 ·····	131	5.5.1 SurfaceView 简介 ·····	158
5.1.2 实例: 自定义控件——走动的 钟表 ·····	133	5.5.2 实例: Camera2 + SurfaceView——实现 拍照 ·····	159
5.2 Android 适配器——BaseAdapter ·····	139	5.6 艺术般的控件: RecyclerView 和 CardView——实现新闻卡片·····	166
5.3 复杂控件 ListView——实现场景对象 选择 ·····	141	5.6.1 RecyclerView 和 CardView 简介·····	166
5.3.1 ListView 控件的简单应用 ·····	141	5.6.2 实例: RecyclerView 和 CardView—— 实现新闻卡片 ·····	167
5.3.2 ListView 控件的高级应用 ·····	141	5.7 Android 7.0 新工具类: DiffUtil ·····	172
5.3.3 实例: ListView 实现场景对象 选择 ·····	144	5.8 更炫的控件: DrawerLayout——实现侧 滑菜单效果 ·····	175
5.4 高级 ListView: ExpandableListView—— 实现商品列表折叠 ·····	149	5.9 对话框 ·····	182
5.4.1 ExpandableAdapter 简介·····	149	5.9.1 常用对话框 ·····	182
5.4.2 实例: ExpandableListView 实现商品 列表折叠 ·····	151	5.9.2 MDDialog ·····	184
5.5 高级控件 Camera2 + SurfaceView——		5.10 本章小结 ·····	187

第六章 数据持久化方案

- | | | | |
|--|-----|--------------------------------|-----|
| 6.1 轻量级存储: SharedPreferences——
实现“记住密码”功能 | 188 | 联系人信息 | 206 |
| 6.2 结构化数据存储——SQLite | 193 | 6.5 最新对象数据库操作——LitePal | 208 |
| 6.2.1 SQLite 简介 | 193 | 6.5.1 LitePal 简介 | 208 |
| 6.2.2 创建 SQLite 数据库 | 194 | 6.5.2 配置 LitePal | 209 |
| 6.2.3 操作数据库 | 196 | 6.5.3 数据库创建和升级 | 210 |
| 6.3 实例: SQLite——实现会员功能 | 198 | 6.5.4 数据库操作 | 214 |
| 6.4 数据共享: ContentProvider——获得 | | 6.5.5 LitePal 1.5.0 的新特性 | 218 |
| | | 6.6 本章小结 | 220 |

第七章 让界面动起来——Android 动画

- | | | | |
|---------------------------------------|-----|---|-----|
| 7.1 绘图动画——绘制仪表盘 | 221 | 7.4 Android 5.0 新动画——AnimatedVector-
Drawable 矢量动画 | 234 |
| 7.2 帧动画 Drawable——模拟电扇
转动 | 226 | 7.5 三维动画: Open GL ES——书本翻页
动画 | 238 |
| 7.3 SurfaceView 实现下雨的天气动画
效果 | 229 | 7.6 本章小结 | 249 |

第八章 更丰富的应用——Android 多媒体

- | | | | |
|--|-----|--------------------------------------|-----|
| 8.1 视频播放器 1——MediaController +
VideoView 播放视频 | 250 | 朗读 | 274 |
| 8.2 视频播放器 2——MediaPlayer +
SurfaceView 播放视频 | 253 | 8.5.1 Text-To-Speech 开发流程 | 275 |
| 8.3 实现按住说话录音 | 258 | 8.5.2 Text-To-Speech 实现文字朗读 | 276 |
| 8.4 实现二维码识别 | 267 | 8.6 Android 语音识别——多种语言语音
识别 | 278 |
| 8.5 Android TTS 文字识别——实现文字 | | 8.7 基于 Ijkplayer 的视频播放器 | 282 |
| | | 8.8 本章小结 | 287 |

第九章 连接到远方——Android 网络开发

- | | | | |
|----------------------------------|-----|--|-----|
| 9.1 Android 应用程序的权限 | 288 | 图片 | 305 |
| 9.1.1 Android 权限机制详解 | 288 | 9.5 使用 Volley 加载网络图片 | 309 |
| 9.1.2 Android 6.0 网络权限管理 | 291 | 9.5.1 使用 ImageRequest 对象加载
图片 | 310 |
| 9.2 解析 JSON 格式数据 | 295 | 9.5.2 使用 ImageLoader 对象加载
图片 | 311 |
| 9.2.1 使用 JSONObject | 296 | 9.6 使用 xUtils 实现网络文件下载 | 313 |
| 9.2.2 使用 GSON | 297 | 9.7 本章小结 | 316 |
| 9.3 使用 OkHttp3 请求天气预报 | 300 | | |
| 9.4 使用 Universal-Image-Loader 加载 | | | |



第十章 更方便的通信——Android 无线通信

10.1 Android Wifi 应用——获取 Wifi 列表	317	10.3 实例：蓝牙控制智能小车	327
10.2 Android 蓝牙——查找蓝牙设备	320	10.4 AndroidNFC——通过 NFC 读取 MifareClassic 卡信息	341
10.2.1 Android 蓝牙开发步骤	322	10.5 本章小结	348
10.2.2 Android 查找蓝牙设备	324		

第十一章 Android 的开源库和开源项目

11.1 Android 的开源库	349	11.2.3 Timber	362
11.1.1 Android View Animations	349	11.2.4 OmniNotes	362
11.1.2 图表库	350	11.2.5 Super Clean Master	363
11.1.3 CameraFilter	355	11.2.6 Pedometer	363
11.1.4 Lottie	355	11.2.7 Traval Mate	364
11.1.5 StyleableToast	357	11.2.8 Music-Player	364
11.1.6 CameraFragment	358	11.2.9 PLDroidPlayer	365
11.2 Android 开源项目	360	11.3 Android 开源网站	367
11.2.1 Easy Sound Recorder	360	11.4 本章小结	369
11.2.2 MLManager	361		

第十二章 Android 应用程序托管和发布

12.1 Git 版本控制工具	370	12.3 将应用程序发布到 360 应用商店	381
12.1.1 安装 Git	371	12.3.1 生成正式签名的 APK 文件	381
12.1.2 创建代码仓库	372	12.3.2 申请 360 开发账号	383
12.1.3 提交本地代码	374	12.3.3 发布应用程序	385
12.2 GitHub	374	12.3.4 嵌入广告	387
12.2.1 在 GitHub 中注册创建版本库	375	12.4 本章小结	397
12.2.2 将代码托管到 GitHub	378		

第一章

Android开发环境

Android 系统是目前市场占有率最高的移动操作系统。由于 Google 的开放政策，任何手机厂商和个人都能免费获取 Android 操作系统的源码，并且可以自由地使用和定制。如今，不管在哪里，都可以看到 Android 手机。今天的 Android 世界可谓欣欣向荣，让我们一起走进 Android 的世界吧。

1.1 Android 开发环境简介

工欲善其事，必先利其器，开发每一种应用程序，必须先准备集成开发环境（Integrated Development Environment，IDE）。目前 Android 应用程序的开发环境有两类：（1）Eclipse + ADT + Android SDK + JDK；（2）Android Studio + JDK。随着 Android Studio 2.2 的发布，Android 应用程序的开发环境已从 Eclipse 转向了 Android Studio。目前已经发布了 Android Studio 2.4 Preview 版和 Android Studio 2.3 正式版。Android 开发环境可以搭建在目前任何一种主流系统（Mac、Windows、Linux）上。

1.2 基于 Eclipse 的开发环境

Eclipse 是比较常见的开发 Android 应用程序的环境之一，到目前为止，大部分 Android 应用程序是基于 Eclipse 开发的。

Android 开发可以使用 Windows XP、Windows Vista、Mac OS、Linux 等操作系统平台。Android 开发所需要的工具为：JDK + Eclipse + Android SDK + ADT，在下载这些工具时，要依据不同的操作系统下载不同的版本。下面列出了 Android 开发工具版本的匹配关系，如表 1-1 所示。

表 1-1 Android 开发工具的版本对应关系

Android SDK 版本	JDK 版本	Eclipse 版本	ADT 版本
Android 6.0	Java 1.8 or higher	EclipseJuno (Version 4.2.1) or higher	ADT 23.0.7
Android4.2.2 (SDK Tools r21.1.)	Java 1.6 or higher	Eclipse Helios (Version 3.6.2) or higher	ADT 21.1.0 ADT 21.0.1
Android4.2 (SDK Tools r20.0.3)	Java 1.6 or higher	Eclipse Helios (Version 3.6.2) or higher	ADT 20.0.3
Android4.1 (SDK Tools r20.0.1)	Java 1.6 or higher	Eclipse Helios (Version 3.6.2) or higher	ADT 20.0.2

(续)

Android SDK 版本	JDK 版本	Eclipse 版本	ADT 版本
Android4.0.3 (SDK Tools r18)	Java 1.6 or higher	Eclipse Helios (Version 3.6.2) or higher	ADT 18.0.0
Android4.0.3 (SDK Tools r17)	Java 1.6 or higher	Eclipse Helios (Version 3.6.2) or higher	ADT 17.0.0
Android4.0.3 (SDK Tools r16)	Java 1.6 or higher	Eclipse Helios (Version 3.6) or higher	ADT 16.0.0
Android4.0 (SDK Tools r15)	Java 1.6 or higher	Eclipse3.3 或者 3.4	ADT 15.0.1
Android3.0 (SDK Tools r10)	Java 1.6 or higher	Eclipse3.3 或者 3.4	ADT 10.0.0
Android2.3 (SDK Tools r8)	Java 1.6 or higher	Eclipse3.3 或者 3.4	ADT 8.0.0

1.2.1 开发环境的搭建

本节介绍如何搭建开发环境。

1. JDK 的安装

JDK 即 Java Development Kit，简单来说，JDK 是面向开发人员的 Java SDK，它提供了 Java 的开发环境和运行环境。因为 Android 应用程序是面向 Java 的应用程序，Android 开发语言使用的是 Java，所以要安装 JDK 开发工具包（JDK 的下载地址读者可自行查询，此处不再赘述），该工具包的下载界面如图 1-1 所示。

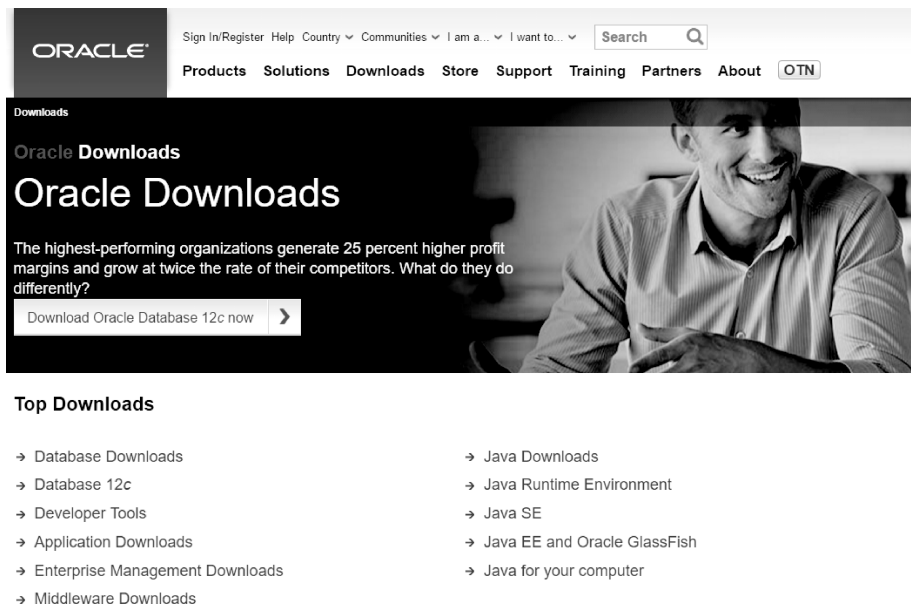


图 1-1 Java JDK 的下载界面

Java 有三个类型版本，便于软件开发人员、服务提供商和设备生产商针对特定的市场进行开发。

(1) Java SE (Java Platform, Standard Edition)。Java SE 以前称为进行开发和部署 J2SE。它允许对在桌面、服务器、嵌入式环境和实时环境中使用的 Java 应用程序。Java SE 包含了支持 Java Web 服务开发的类，并为 Java Platform, Enterprise Edition (Java EE) 提供基础。

(2) Java EE (Java Platform, Enterprise Edition)。这个版本以前称为 J2EE。企业版本帮助开发和部署可移植、健壮、可伸缩且安全的服务器端 Java 应用程序。

(3) Java ME (Java Platform, Micro Edition)。这个版本以前称为 J2ME。Java ME 为在移动设备和嵌入式设备(比如手机、PDA、电视机顶盒和打印机)上运行的应用程序提供健壮且灵活的环境。

目前最新的版本为 JDK 8u121, 在下载之前要选择 JDK 运行的平台, 目前 JDK 可运行的平台有: Linux、Mac OS、Solaris SPARC、Window 四种, 而 Windows 平台又分为 32 位和 64 位两种, 本书选择 64 位 SE 版本, 对应的文件为 `jdk-8u111-windows-x64.exe`, 在下载完成之后, 安装过程如图 1-2 所示。

2. Eclipse

Eclipse 是一个开放源代码的、基于 Java 的可扩展开发平台。就其本身而言, 它只是一个框架和一组服务, 用于通过插件组件构建开发环境。Eclipse 的初始下载界面如图 1-3 所示。之后单击 `Download Packages` 链接, 进入下一界面。



图 1-2 JDK 的安装过程

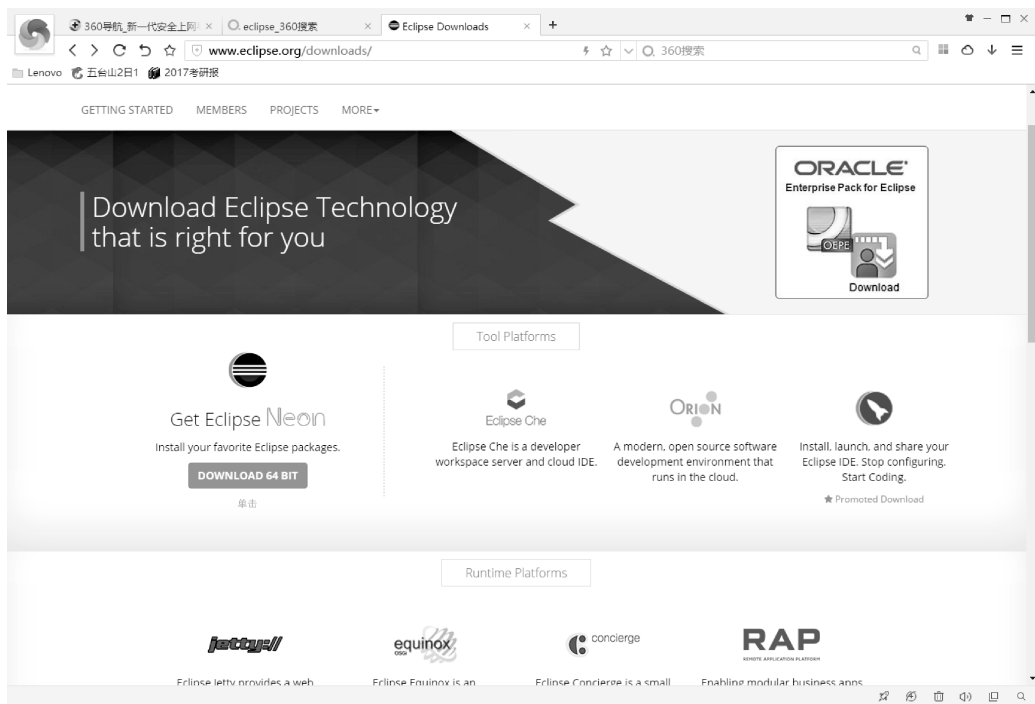


图 1-3 Eclipse 的初始下载界面

在如图 1-4 所示的界面中选择 Eclipse IDE for Java EE Developers, 选择 64 bit 进行下载,

下载的文件名为“eclipse-jee-neon-2-win32-x86_64.zip”。

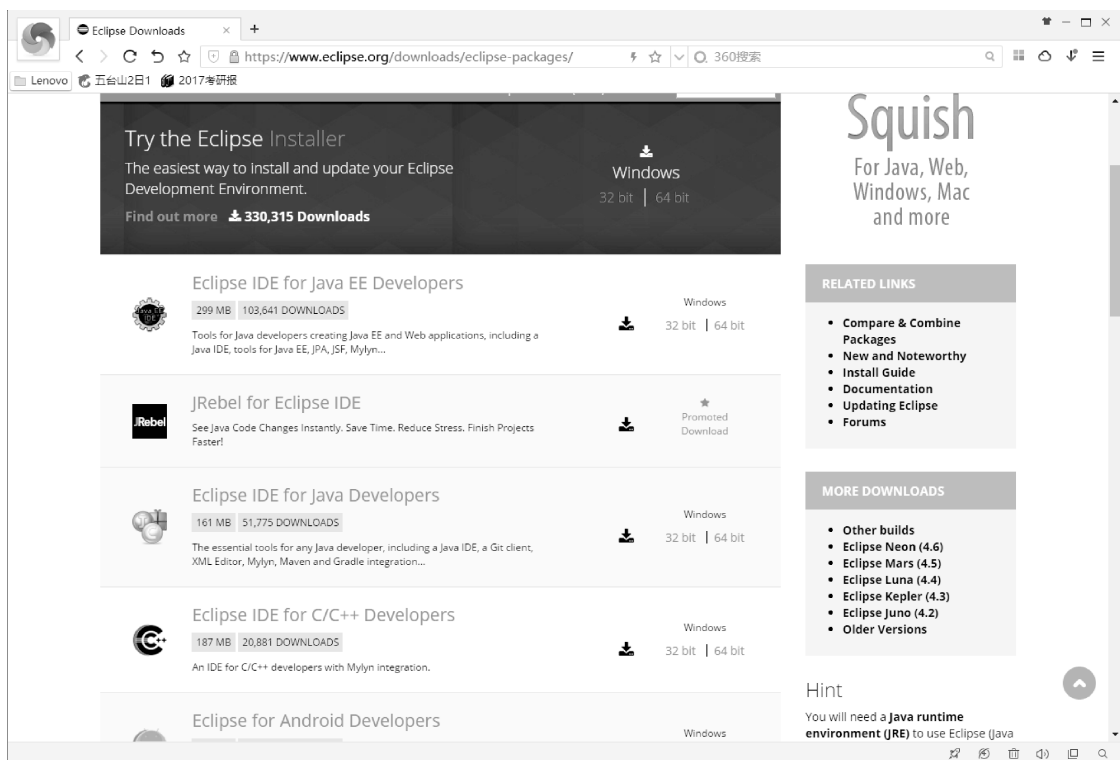


图 1-4 Eclipse 下载界面

3. 安装 Android ADT 插件

用 Eclipse 集成环境进行 Android 开发时，需要安装 ADT（Android Development Tools）。ADT 是 Eclipse 开发 Android 应用程序的插件。

安装方法如下。

(1) 启动 Eclipse，然后选择菜单栏中 Help > Install New Software 命令，如图 1-5 所示。

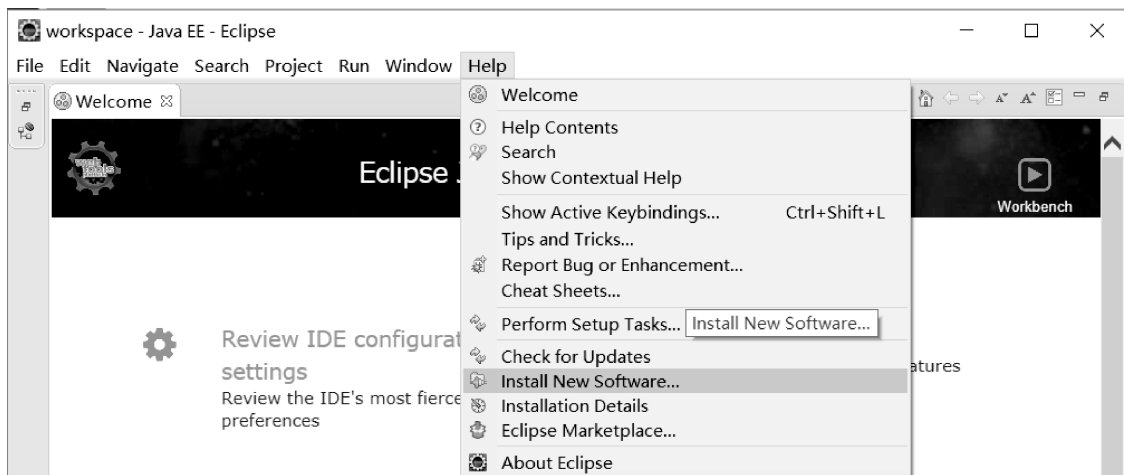


图 1-5 选择菜单命令

(2) 在弹出的窗口中，单击 Add 按钮，在弹出的对话框中添加存储库，输入 ADT 的名称，以及位置：<https://dl-ssl.google.com/android/eclipse/>，单击 OK 按钮，如图 1-6 所示。

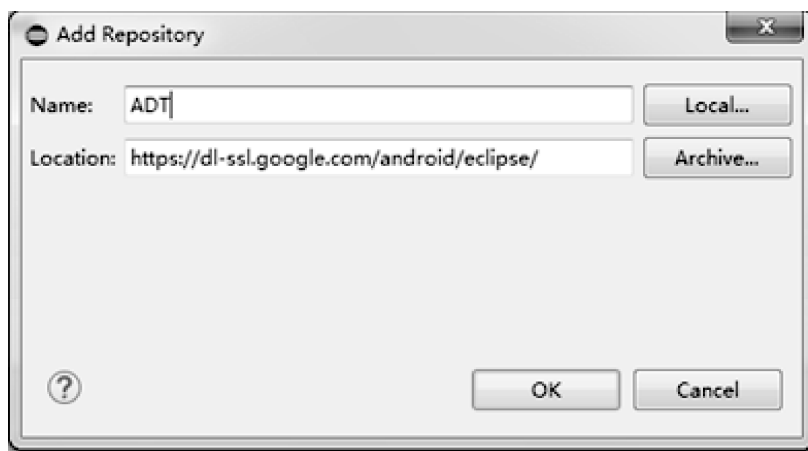


图 1-6 安装 ADT 插件

下载完 ADT 软件后，选择菜单栏中 Help > Install New Software 命令，在弹出的窗口中，单击 Add 按钮，然后单击 Archive 按钮，选中 ADT 文件，如图 1-7 所示。

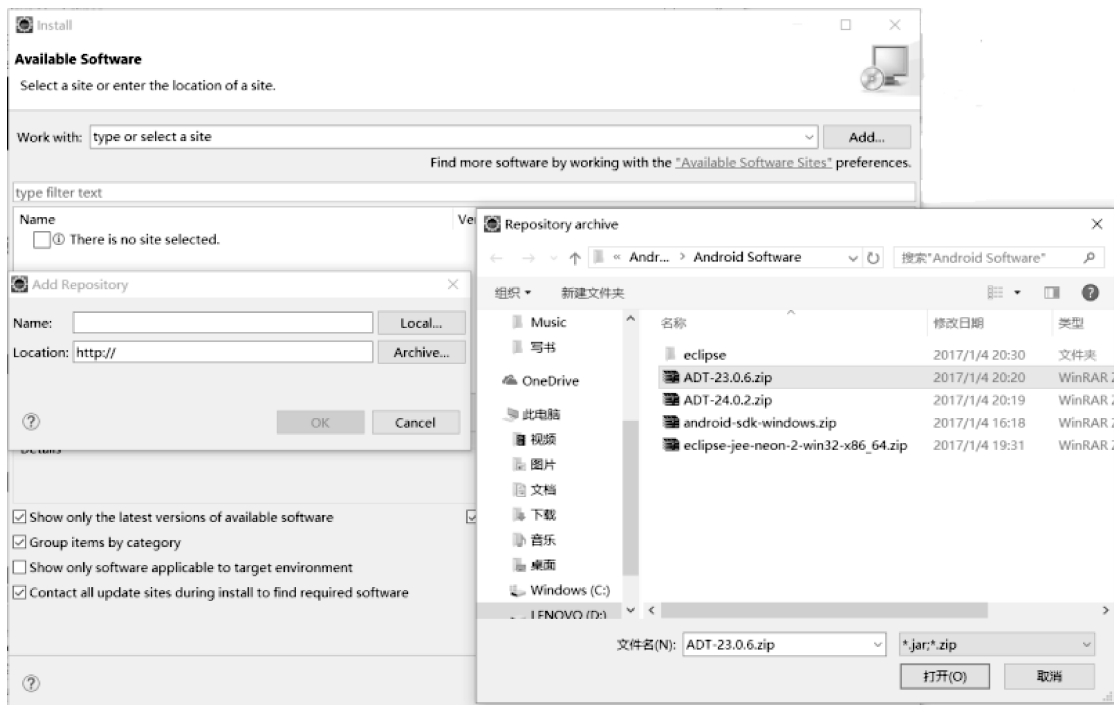


图 1-7 加载下载的 ADT 插件

(3) 打开 ADT 插件后，选择 Developer Tools 选项，如图 1-8 所示。

(4) 单击 Next 按钮，开始安装 ADT-23.0.6，直到安装完成，如图 1-9 所示。

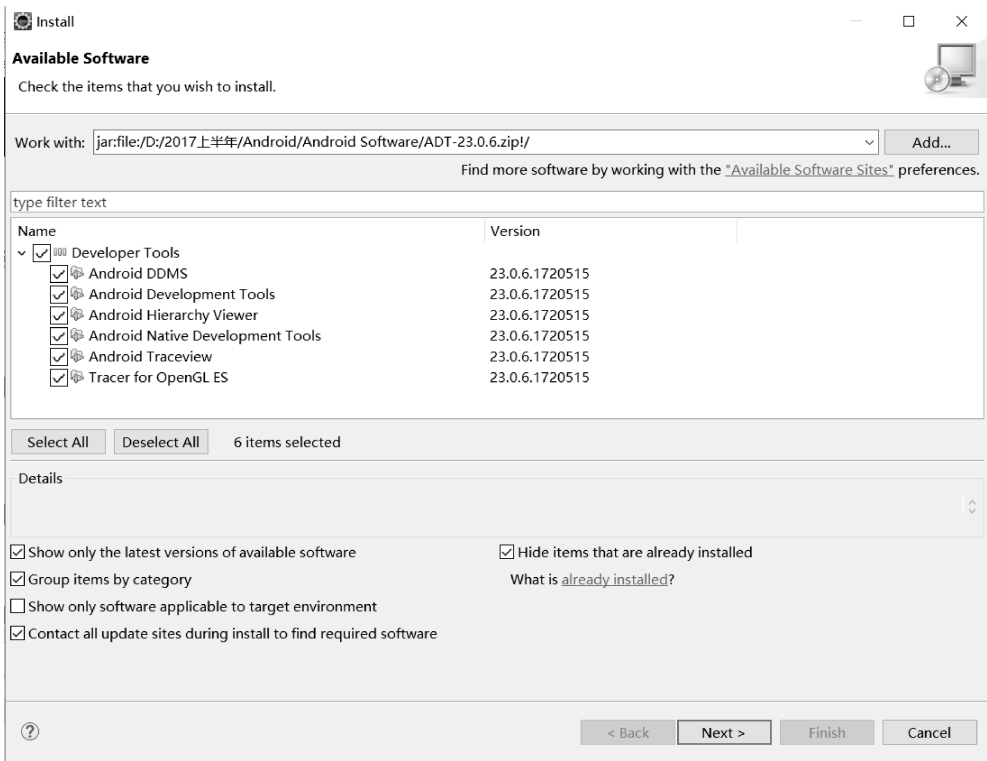


图 1-8 选择 Developer Tools 选项

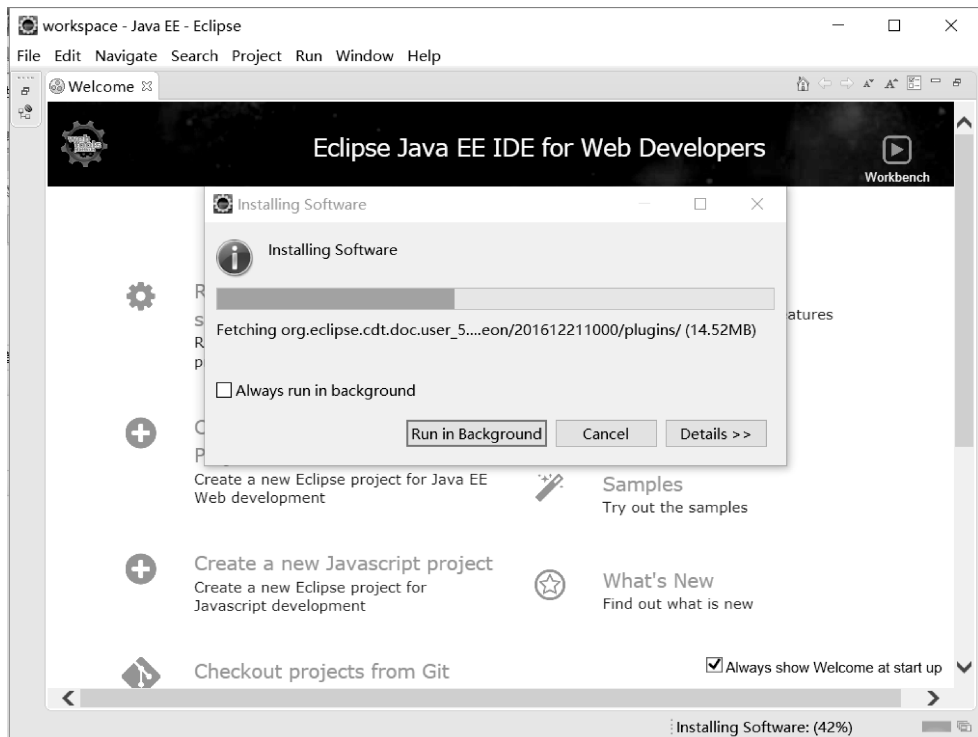


图 1-9 安装 ADT-23.0.6

4. Android SDK

Android SDK 可从 Android 的官网下载，选择的 Android SDK 版本应与 ADT 的版本对应，前面安装的 ADT 为“ADT-23.0.6.zip”，那么对应的版本也要是 23 以上的 Android SDK，本书选择的文件为“android-sdk-windows-r24-updated.7z”，将压缩文件“android-sdk-windows-r24-updated.7z”解压，可以看到下载的文件和文件夹，如图 1-10 所示。

Android SDK 目录中的内容如下。

(1) add-ons：里面保存一些附加的库，第三方公司为 Android 平台开发的附加功能系统，比如 GoogleMaps 等（一开始此包内容为空）。

(2) build-tools：里面保存了构建项目时用到的工具，当新建 Android 项目时，会用到这个包。如果没有此包，新建的项目会报错。其中还包括一些编译的工具。总之这个包不能缺少。

(3) docs：其中保存了 SDK 文档，包括对各种控件、类的官方说明，可以在里面找到所有的开发文档。

(4) extras：该文件夹中存放了 Google 提供的 USB 驱动、Intel 提供的硬件加速等附件工具包。

(5) platform-tools：该文件夹中存放了 Android 平台的相关工具，比如 adb.exe、sqlite3.exe。

(6) tools：里面存放了大量的 Android 开发、调试的工具和 platform-tools 有些重复，都是与开发有关的工具。

(7) platforms：platforms 是 SDK 中最重要的文件夹，其中可以有許多不同版本的 SDK。

(8) system-images：这里面存放的是创建 Android 虚拟机时的镜像文件。

(9) sources：里面存放的是源码。

5. 在 Eclipse 中配置 Android SDK

Android SDK 下载完成以后，启动 Eclipse，单击菜单栏中 Window > Preferences 命令，设置 Android SDK 目录，本书中将目录设为“D:\2017\Android\Android Software\android-sdk-windows-r24-updated”。完成设置后，可以看到此下载包仅包含一个 Android 5.1.1 版本，如图 1-11 所示。

若要增加版本，则运行图 1-10 中 Android SDK 目录中的 SDK Manager.exe 文件，如图 1-12 所示，选中没有安装的项，也就是 Status 标识为 Not installed 的项目，单击下面的安装按钮，即可安装。

名称	修改日期	类型
add-ons	2015/2/28 5:56	文件夹
build-tools	2015/4/5 12:29	文件夹
docs	2015/4/5 12:33	文件夹
extras	2015/4/5 12:35	文件夹
platforms	2015/4/5 12:33	文件夹
platform-tools	2015/4/5 12:29	文件夹
samples	2015/4/5 12:35	文件夹
sources	2015/4/5 12:35	文件夹
system-images	2015/4/5 12:37	文件夹
temp	2015/4/7 10:36	文件夹
tools	2015/2/27 16:35	文件夹
AVD Manager.exe	2015/2/28 5:56	应用程序
SDK Manager.exe	2015/2/28 5:56	应用程序
SDK Readme.txt	2015/2/28 5:56	文本文档

图 1-10 Android 的 SDK 目录

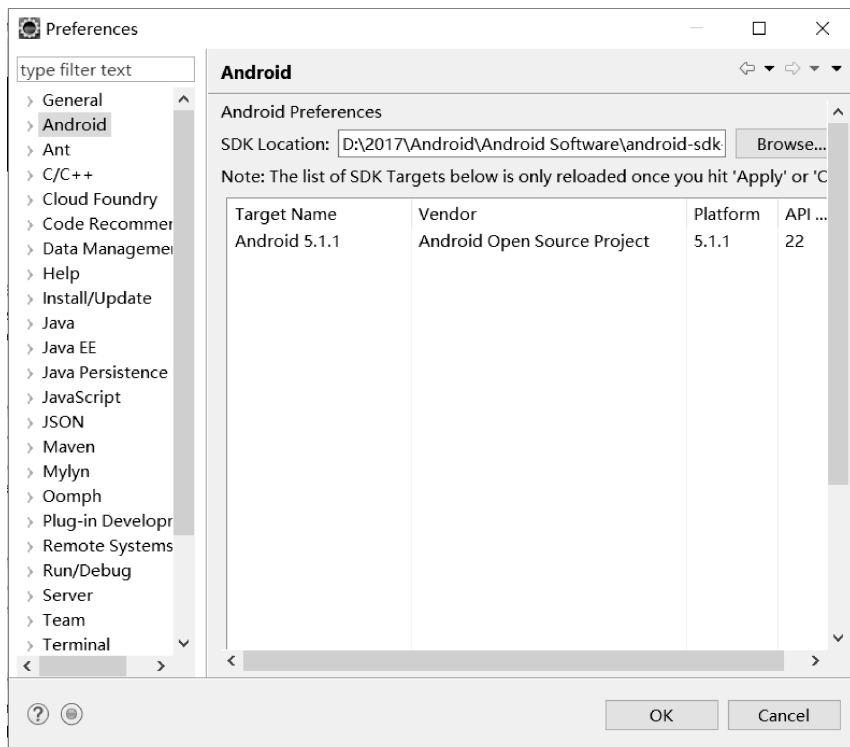


图 1-11 在 Eclipse 中配置 Android SDK

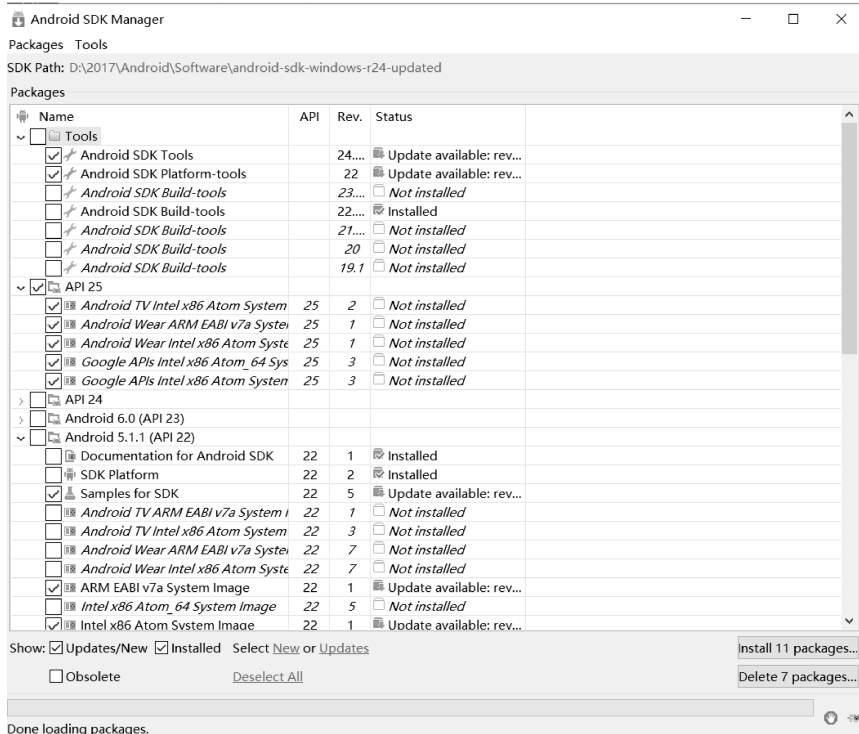


图 1-12 SDK Manager.exe 运行界面

6. 创建 Android 模拟器

Android SDK 自带一个移动模拟器，它是一个可以运行在计算机上的虚拟设备。该模拟器可以在不使用物理设备的情况下预览、开发和测试 Android 应用程序，一般开发程序先在模拟器运行，调试完成后下载到真机运行。

(1) 建立 Android 虚拟机，在 Eclipse 主界面中单击菜单栏中的 Window > Android Virtual Device Manager 命令，如图 1-13 所示。

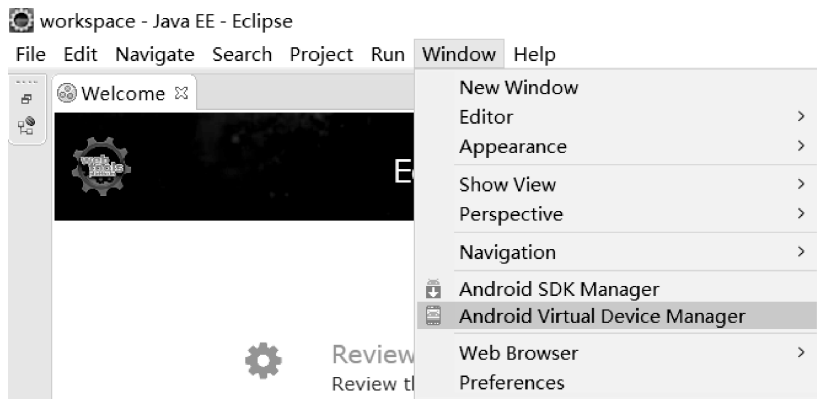


图 1-13 创建 Android 虚拟机

(2) 弹出的窗口如图 1-14 所示。

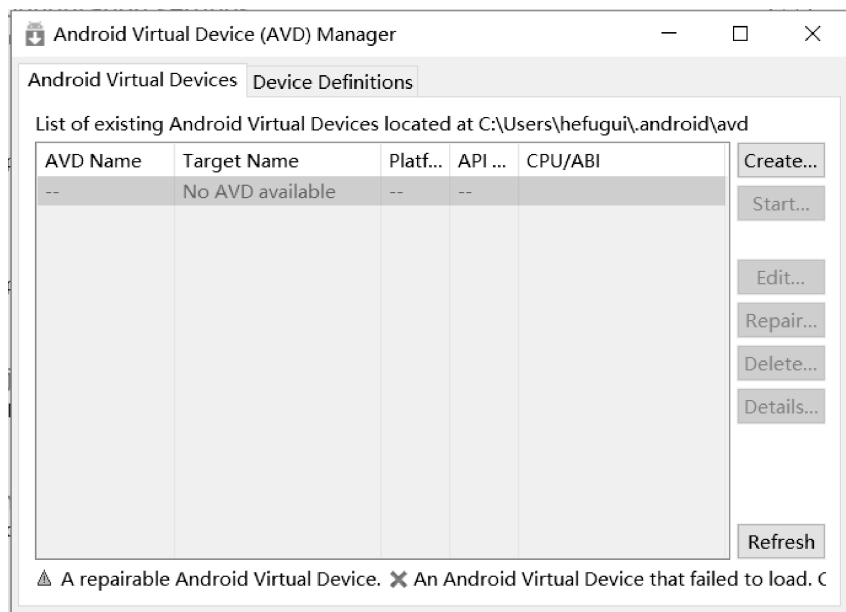


图 1-14 Android Virtual Device (AVD) Manager 窗口

(3) 单击 Create 按钮，弹出虚拟机编辑窗口，进行配置，如图 1-15 所示。

(4) 单击 OK 按钮，回到 Android Virtual Device (AVD) Manager 窗口，可以看到已经建立了一个虚拟机，如图 1-16 所示。

(5) 单击 Start 按钮，启动虚拟机，如图 1-17 所示。