



Android移动应用开发 基础教程

讲授：葛新



第1章 Android开发起步

本章主要内容：

- Android简介
- 搭建Android开发环境
- 创建Android项目
- Android编程的日志工具



1.1 Android简介

本节主要内容：

1. Android平台特点
2. Android体系架构
3. Android版本



1.1 Android简介

- Android本义为 “机器人”
- Android是基于Linux内核、应用Java开发的轻量级的移动操作系统。
- Google为Android内置了诸多常用应用：电话、短信、个人管理、多媒体播放、网页浏览等等
- 2003年10月，Andy Rubin等人创建了Android公司，组建了Android开发团队。
- 2005年8月，Google收购了Android公司及其开发团队，并有Andy Rubin继续负责Android项目。
- 2007年11月，Google正式发布Android平台，Android平台也不在局限于手机，还逐渐扩展到平板电脑和及其智能设备领域。
- 2011年一举超过称霸移动领域多年的诺基亚Symbian系统，成为全球市场份额占有率第一的智能设备平台。



1.1.1 Android平台特点

1、开放性

Android平台的源代码开放。开发人员可任意访问其核心代码，设计出丰富多彩的应用。Android的开放性也使更多的智能设备厂商加入到Android联盟中来。

2、不再受营运商限制

早期的手机，其上的应用、网络接入方式等等，全部由营运商说了算。Android打破了这种束缚，用户可以根据自己的喜好来定制手机应用。

3、丰富的硬件选择

Android的开发性，也使硬件生产商可以设计出功能各异的多种产品，例如Android手机、平板、眼镜、电视、车载设备以及穿戴设备等，为用户提供更多的选择。

4、开发不受限制

Android平台为开发人员提供了更加宽泛、自由的开发环境，使得各种优秀的应用不断出现。同时，这也使一些不健康、恶意的应用出现，如何遏制不良应用也成为Android的一个难题。

5、与Google应用无缝结合

Android平台可与Google的地图、邮件、搜索等优秀服务无缝结合，在手机、平板电脑以及其他智能设备上可以轻松使用这些服务。



1.1.2 Android体系架构





1. Linux内核层

Android系统运行于Linux内核之上，主要包括电源管理和各种启动模块，如显示驱动、键盘驱动、摄像头驱动、WiFi驱动、USB驱动等。

2. 硬件抽象层HAL

硬件抽象层包含多个库模块，为上层的Java API提供标准的设备硬件功能支持。开发人员通过框架API访问设备硬件时，Android系统为硬件加载相应的库模块。

3. 系统运行库层

系统运行库层包含了一系列原生C/C++库，它们通过Android应用框架API为开发者提供各种服务。例如，Webkit库提供浏览器支持、OpenGL ES库提供2D/3D绘画支持等。

4. Java API框架层

Java API框架层通过API提供Android系统的全部功能

5. 系统应用层

系统应用层包含了Android系统自带的一套核心应用，包括电子邮件、短信、日历、联系人等。



1.1.3 Android版本

版本号	系统代号	API级别	市场占有率
2.2	Froyo	8	0.1%
2.3.3 - 2.3.7	Gingerbread	10	1.7%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	1.6%
4.1.x	Jelly Bean	16	6.0%
4.2.x	Jelly Bean	17	8.3%
4.3	Jelly Bean	18	2.4%
4.4	KitKat	19	29.2%
5.0	Lollipop	21	14.1%
5.1	Lollipop	22	21.4%
6.0	Marshmallow	23	15.2%
7.0	Nougat	24	<0.1%



Android 7.0新增的主要功能和特性

1、多窗口支持

- 多窗口支持使用户可在运行Android 7.0系统的设备（手机、平板或TV）上一次打开两个应用。在Android 7.0手机和平板中，用户可以并排运行两个应用，或者在分屏模式下一个应用在另一个之上。用户可拖动两个应用之间的分隔线调整应用。在Android 7.0 TV中，同时运行的两个应用实现画中画模式，从而在看电视的同时允许用户浏览或使用其他应用。
- 多窗口支持也允许在两个应用之间执行拖放操作，进一步增强用户体验。



Android 7.0新增的主要功能和特性

2、通知功能增强

Android 7.0重新设计了通知，使其速度更快，也更易于使用。主要改变包括：

- **模板进行了更新**：通知模板更新，使开发人员只需修改少量代码即可实现通知。
- **允许更多的自定义消息传递样式**：使用MessagingStyle类的通知时，可自定义更多的与通知有关的用户界面标签，可配置消息、会话标题和内容视图等内容。
- **捆绑通知**：系统可将消息组合成组显示，用户可适当地进行拒绝或归档操作。
- **直接回复**：在实时通信应用中支持内联回复，方便用户在通知界面中快速回复短信。
- **自定义视图**：新的API允许在通知中使用自定义视图时充分利用系统装饰元素。



Android 7.0新增的主要功能和特性

3.及时编译（JIT）和预编译（AOT）

- Android 7.0添加了JIT编译器，对ART进行代码分析，提升应用性能。JIT编译器对AOT（Ahead of Time）编译器进行了补充，有助于提高运行性能，节约存储空间，加快应用和系统的更新速度。
- 通过配置文件，可让Android运行组件根据应用运行的实际情况管理每个应用的AOT/JIT编译。配置文件还可进一步指导便宜减少内存占用，这对低内存设备尤其重要。通过配置文件的知道，还可在设备处于空闲或充电状态时才进行编译，从而节约时间和省电。



Android 7.0新增的主要功能和特性

4. 随时随地的低功耗模式

- Android 6.0推出了低功耗模式，当设备未连接电源、处于静止状态且屏幕关闭时，设备进入低功耗模式，系统通过推迟应用的CPU和网络活动来实现省电目的。
- Android 7.0进一步完善了低功耗模式。只要屏幕关闭且未连接电源，但不一定要处于静止状态（例如用户将手机放于口袋中）时低功耗模式就会启动，限制CPU和网络活动。



Android 7.0新增的主要功能和特性

5.流量节省程序

- 相信读者都为不得不使用移动流量带来的昂贵资费烦恼过。Android 7.0推出了流量节省模式，允许用户在设置中启用流量节省程序，当设备使用按流量计费的网络时，系统可屏蔽后台流量，同时指示前台应用尽可能少用流量。例如，限制流媒体服务的比特率、降低图像质量、延迟最佳的预缓冲等。用户还可将应用加入白名单，从而允许其在启用了流量节省程序后再后台的流量消耗。



Android 7.0新增的主要功能和特性

6.号码屏蔽

- Android 7.0增加了号码屏蔽功能，允许默认短信应用、默认手机应用和运营商应用通过框架API访问屏蔽的号码列表，其他应用无法访问此列表。利用平台标准的号码屏蔽功能，还可以屏蔽已屏蔽号码发出的短信，可通过备份/还原在重置或跨设备保留屏蔽的号码，可在多个应用中使用相同的屏蔽号码列表。Android设备的运营商可通过读取用户设备中的屏蔽号码列表，执行服务器端的屏蔽，阻止已屏蔽号码的来电和短信到达用户。



1.2 搭建Android开发环境

本节主要内容：

1. 需要哪些工具
2. JDK下载安装
3. Android Studio简介
4. Android Studio的下载安装



1.2.1 需要哪些工具

Android开发需要的工具如下。

- **JDK**: Android程序都使用Java语言进行编写, JDK是Java语言开发工具包, 它包含了Java运行环境、工具、基础类库等。目前, Android支持Java 7全部功能和Java 8的部分功能。
- **Android SDK**: 这是Google提供的Android开发工具包, 开发Android应用时, 需要在IDE中引入该包。
- **Android Studio**: 这是Google推出的Android开发IDE。早期的Android开发大多使用Eclipse, 在其中安装Google提供的Android开发插件ADT, 来进行Android开发。随着Android Studio的不断完善和功能增强, 以成为Android开发的理想选择。Google也不再维护和更新ADT插件。



1.2.2 JDK下载安装

Oracle Technology Network - Java - Java SE - Downloads

Java SE
Java EE
Java ME
Java SE Support
Java SE Advanced & Suite
Java Embedded
Java DB
Web Tier
Java Card
Java FX
New to Java
Community
Java Magazine

Overview Downloads Documentation Community Technologies Training

Java SE Development Kit 8 Downloads

Thank you for downloading this release of the Java™ Platform, Standard Edition Development Kit (JDK™). The JDK is a development environment for building applications, applets, and components using the Java programming language.

The JDK includes tools useful for developing and testing programs written in the Java programming language and running on the Java platform.

See also:

- Java Developer Newsletter: From your Oracle account, select Subscriptions, expand Technology, and subscribe to Java.
- Java Developer Day hands-on workshops (free) and other events
- Java Magazine

JDK 8u111 Checksum
JDK 8u112 Checksum

Java SE Development Kit 8u111

You must accept the Oracle Binary Code License Agreement for Java SE to download this software.

☒ Accept License Agreement ☐ Decline License Agreement

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	71.78 MB	jdk-8u111-linux-arm32-hf.tar.gz
Linux ARM 64 Hard Float ABI	74.73 MB	jdk-8u111-linux-arm64-hf.tar.gz
Linux x86	160.30 MB	jdk-8u111-linux-i386.rpm
Linux x86	175.04 MB	jdk-8u111-linux-i386.tar.gz
Linux x64	158.25 MB	jdk-8u111-linux-x64.rpm
Linux x64	173.04 MB	jdk-8u111-linux-x64.tar.gz
Mac OS X	227.38 MB	jdk-8u111-macosx-x64.dmg
Solaris SPARC 64-bit	131.92 MB	jdk-8u111-solaris-sparc64.tar.gz
Solaris SPARC 64-bit	93.02 MB	jdk-8u111-solaris-sparc64.tar.gz
Solaris x64	140.58 MB	jdk-8u111-solaris-x64.tar.gz
Solaris x64	95.82 MB	jdk-8u111-solaris-x64.tar.gz
Windows x86	180.22 MB	jdk-8u111-windows-i386.exe
Windows x64	184.54 MB	jdk-8u111-windows-x64.exe

Java SE Development Kit 8u111



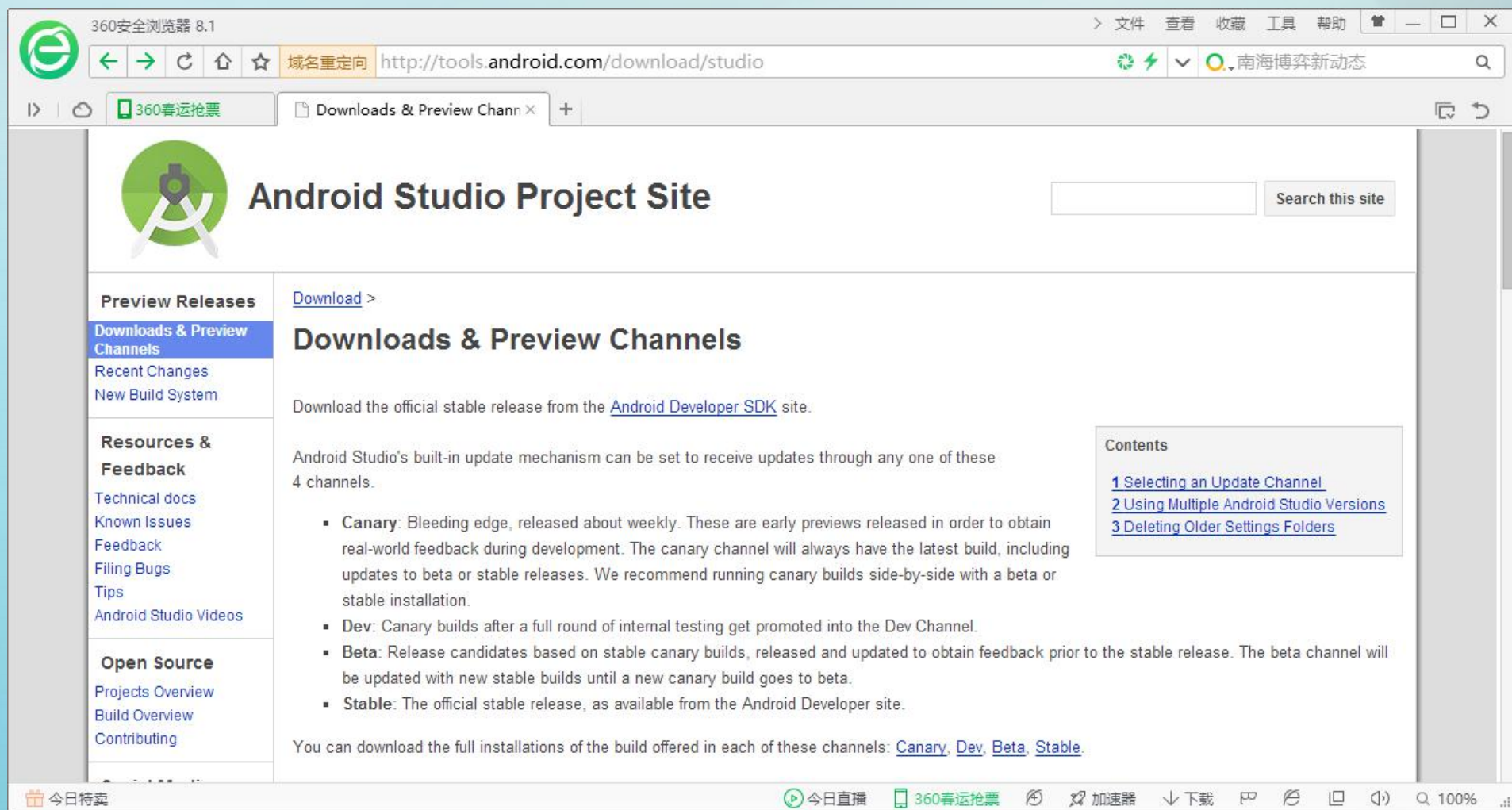
1.2.3 Android Studio简介

Android Studio主要的特点如下。

- Instant Run
- 智能代码编辑器
- 快速、功能丰富的模拟器
- 强大灵活的构建系统
- 专门为Android设备开发
- 代码模板和GitHub集成



1.2.4 Android Studio的下载安装





1.3 创建第一个Android项目

本节主要内容：

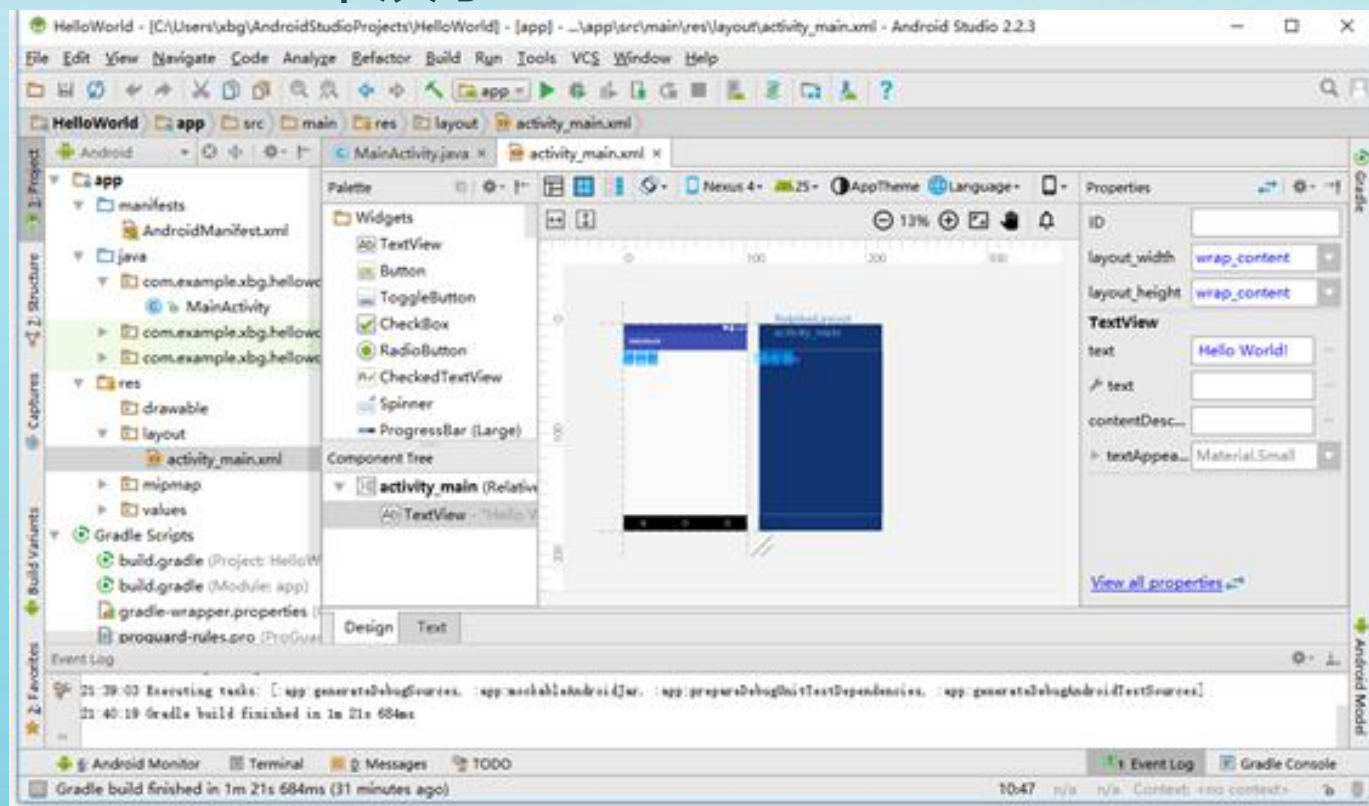
1. 创建Hello World项目
2. 创建模拟器
3. 运行项目
4. 了解Android项目组成



1.3.1 创建Hello World项目

Hello World项目在屏幕上显示“Hello World”字符串，通过创建该项目了解Android Studio如何创建一个新的项目。

在Android Studio中演示





1.3.2 创建模拟器

在创建项目时，Android Studio自动创建很多东西，现在不需要修改任何代码即可运行前面创建的HelloWorld项目。不过在运行之前，需要创建一个模拟器作为项目运行设备。当然，也可连接一个物理设备（例如一台Android手机）来测试运行项目。

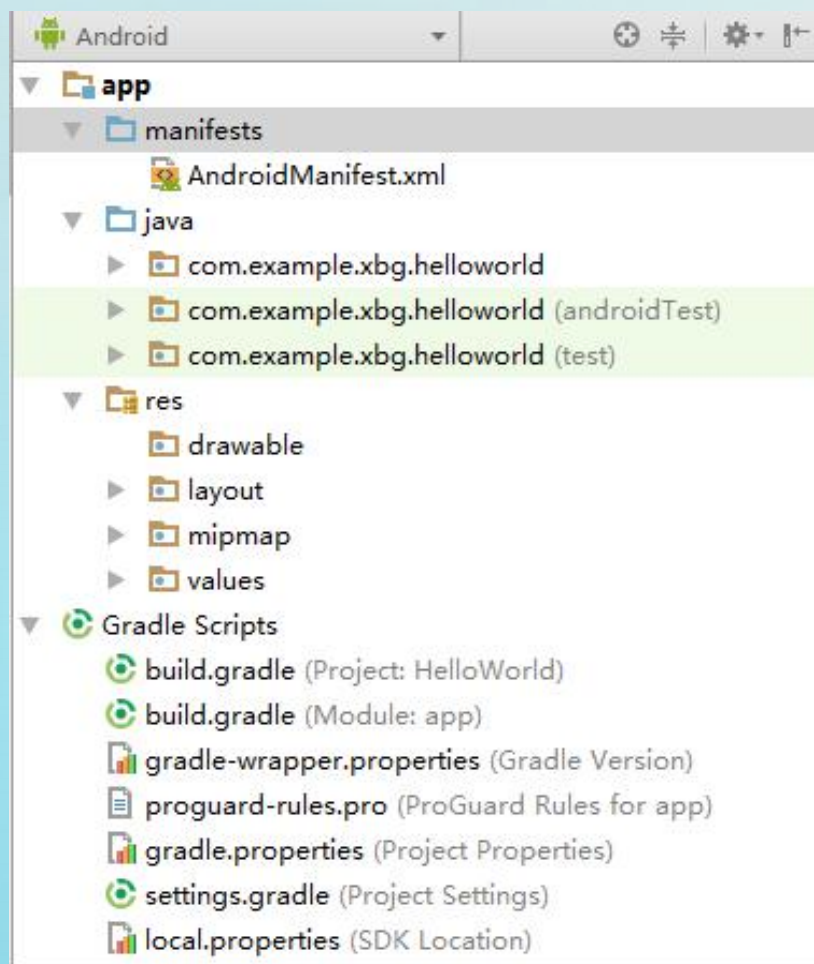


1.3.3 运行项目

- 在Android Studio中演示



1.3.4 了解Android项目组成





1.4 Android编程小工具：日志

本节主要内容：

1. 使用日志API输出调试信息
2. 日志分类与日志过滤器



1.4.1 使用日志API输出调试信息

可使用下面的多种方法在程序中输出调试信息，这些信息统称为日志，具有不同的级别。

- `System.out.println()`：输出的日志级别为Info，即普通信息。
- `System.err.println()`：输出的日志级别为Warn，即警告信息。
- `Log.v()`：输出的日志级别为Verbose，即冗余信息。
- `Log.d()`：输出的日志级别为Debug，即调试信息。
- `Log.i()`：输出的日志级别为info，即普通信息。
- `Log.w()`：输出的日志级别为Warn，即警告信息。
- `Log.e()`：输出的日志级别为Error，即错误信息。



1.4.2 日志分类与日志过滤器

Android应用程序日志可分为Verbose、Debug、Info、Warn和Error等5个级别，依次从低到高。

