Developing apps for Android devices is popular these days. Perhaps you would like to learn how to develop your own Android apps with Java 7 (although you cannot use APIs and language features newer than Java 5).

Chapter 12 presents a rapid introduction to app development. You first learn about Android architecture and the architecture of an Android app. You then learn how to install the Android SDK and a platform so that you have the tools and an environment to begin app development. Because the SDK provides an emulator to emulate Android devices, you next learn how to create and start an Android Virtual Device (AVD), which you can use to test your apps in lieu of an actual Android device. Finally, you’re introduced to a simple app, learn how to create this app via the SDK, and learn how to install and run the app on an AVD.

**Note** If you want to learn more about Android after reading this chapter, check out *Beginning Android 3* by Mark Murphy (Apress, 2011; ISBN: 978-1-4302-3297-1). You might also want to check out *Android Recipes* by Dave Smith and Jeff Friesen (Apress, 2011; ISBN: 978-1-4302-3413-5). *Android Recipes* teaches you additional Android app architecture fundamentals, shows you how to install the Eclipse IDE and develop an app with that IDE, presents solutions to various app development problems, introduces you to various third-party development tools and the Android NDK, shows you how to create your own libraries and use third-party libraries, and presents app design guidelines.

**Exploring Android and Android App Architectures**

The *Android Developer’s Guide* (http://developer.android.com/guide/index.html) defines *Android* as a software stack (a set of software subsystems needed to deliver a fully functional solution) for mobile devices. This stack includes an operating system (a modified version of the Linux kernel), middleware (software that connects the low-level operating system to high-level apps), and key apps (written in Java) such as a web browser (known as Browser) and a contact manager (known as Contacts).

Android offers the following features:

- Application framework enabling reuse and replacement of app components
- Bluetooth, EDGE, 3G, and WiFi support (hardware dependent)
• Camera, GPS, compass, and accelerometer support (hardware dependent)
• Dalvik virtual machine optimized for mobile devices
• GSM Telephony support (hardware dependent)
• Integrated browser based on the open source WebKit engine
• Media support for common audio, video, and still image formats (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
• Optimized graphics powered by a custom 2D graphics library; 3D graphics based on the OpenGL ES 1.0 specification (hardware acceleration optional)
• SQLite for structured data storage

Note Although not part of an Android device’s software stack, Android’s rich development environment (including a device emulator and a plugin for the Eclipse IDE) could also be considered an Android feature.

Android apps are written in Java and can access only the Java APIs described in the API reference at http://developer.android.com/reference/packages.html (as well as Android-oriented third-party APIs). They cannot access Java APIs beyond Java 5. This restriction affects Java 7’s try-with-resources statement, which is based on the new java.lang.AutoCloseable interface and API support for suppressed exceptions. You cannot use try-with-resources in your Android source code.

Note Not all Java 5 (and previous version) APIs are supported by Android. For example, Android doesn’t support the Abstract Window Toolkit (AWT) or Swing. Instead, it offers a smaller set of user-interface APIs.

Android Architecture

The Android software stack consists of apps at the top, middleware (consisting of an application framework, libraries, and the Android runtime) in the middle, and a Linux kernel with various drivers at the bottom. Figure 12-1 shows this layered architecture.