1

Introduction to Android

Objectives

In this chapter you'll be introduced to:

- The history of Android and the Android SDK.
- Google Play Store for downloading apps.
- The Android packages used in this book to help you create Android apps.
- A quick refresher of object-technology concepts.
- Key software for Android app development, including the Android SDK, the Java SDK and the Android Studio Integrated Development Environment (IDE).
- Important Android documentation.
- Test-driving an Android tip-calculator app in Android Studio.
- Characteristics of great Android apps.

Self-Review Exercises

1.1 Fill in the blanks in each of the following statements:

a) In 2007, the __________ was formed to develop, maintain and evolve Android, driving innovation in mobile technology and improving the user experience while reducing costs.

ANS: Open Handset Alliance.

b) The __________ IDE allows you to create, run and debug Android apps.

ANS: Android Development Tools (ADT).

c) Multitouch screens allow you to control your Android device with __________ involving one touch or multiple simultaneous touches.

ANS: gestures.

d) With web services, you can create __________, which enable you to rapidly develop apps by quickly combining complementary web services, often from different organizations and possibly other forms of information feeds.

ANS: mashups.

e) Android uses a collection of __________, which are named groups of related, predefined classes.

ANS: packages.

f) The __________, included in the Android SDK, allows you to run Android apps in a simulated environment within Windows, Mac OS X or Linux.

ANS: Android emulator.

g) Almost any noun can be reasonably represented as a software object in terms of __________ (e.g., name, color and size) and behaviors (e.g., calculating, moving and communicating).

ANS: attributes.

h) A program unit called a(n) __________ houses the methods that perform its tasks.

ANS: class.

i) You send messages to an object. Each message is a(n) __________ that tells a method of the object to perform its task.

ANS: method call.

1.2 State whether each of the following is *true* or *false*. If *false*, explain why.

a) One benefit of developing Android apps is that the operating system is proprietary to Google.

ANS: False. The operating system is open source and free.

b) The openness of the Android platform discourages innovation.

ANS: False. The openness of the platform spurs rapid innovation.

c) You can reuse a class many times to build many objects. Reuse of existing classes when building new classes and programs saves time and effort.

ANS: True.

d) Attributes are specified by the class's methods.

ANS: False. Attributes are specified by the class's instance variables.

e) Objects may communicate with one another, but they're normally not allowed to know how other objects are implemented—implementation details are hidden within the objects themselves.

ANS: True.

1.3 Fill in the blanks in each of the following statements (based on Section 1.8):

a) Objects have the property of ________—although objects communicate with one another, they're normally not allowed to know how other objects are implemented.

ANS: information hiding.

b) The ________ that objects come from are essentially reusable software components; they include attributes and behaviors.

ANS: classes.

c) The process of analyzing and designing a system from an object-oriented point of view is called ________.

ANS: object-oriented analysis and design (OOAD).

d) With ________, new classes of objects are derived by absorbing characteristics of existing classes, then adding unique characteristics of their own.

ANS: inheritance.

e) The size, shape, color and weight of an object are considered ________ of the object's class.

ANS: attributes.

f) A class that represents a bank account might contain one ________ to deposit money to an account, another to withdraw money from an account and a third to inquire what the account's current balance is.

ANS: method.

g) You must build an object of a class before a program can perform the tasks that the class's methods define—this process is called ________.

ANS: instantiation.

h) The balance of a bank account class is an example of a(n) ________ of that class.

ANS: attribute.

i) Your project's requirements define what the system is supposed to do and your design specifies ________ the system should do it.

ANS: how.

Exercises

1.4 Fill in the blanks in each of the following statements:

a) Android apps are developed with __________—one of the world's most widely used programming language, a logical choice because it's powerful, free and open source.

ANS: Java.

b) GUI programming in Java is __________ driven, so you'll write apps that respond to various user interactions such as screen touches and keystrokes.

ANS: event.

c) Touching the screen and holding your finger in position is called a(n) __________.

ANS: long press.

d) Touching the screen, then moving your finger in one direction and releasing it is called a(n) __________.

ANS: fling.

e) Before running an app in the emulator, you'll need to create a(n) __________, which defines the characteristics of the device on which you want to test, including the hardware, system image, screen size, data storage and more.

ANS: Android Virtual Device.

f) Performing a task in a program requires a(n) __________ which houses the program statements that actually perform its tasks.

ANS: method.

g) You must build an object of a class before a program can perform the tasks that the class's methods define. The process of doing this is called __________.

ANS: instantiation.

h) __________ helps you build more reliable and effective systems, because existing classes and components often have gone through extensive testing, debugging and performance tuning.

ANS: Reuse.

i) Classes __________ (i.e., wrap) attributes and methods into objects—an object's attributes and methods are intimately related.

ANS: encapsulate.

j) A new class of objects can be created quickly and conveniently by __________—the new class absorbs the characteristics of an existing one, possibly customizing them and adding unique characteristics of its own.

ANS: inheritance.

k) Unlike actual buttons on a device, __________ buttons appear on the device's touch screen.

ANS: soft.

l) __________ was the complete user-interface redesign that Google introduced with Android 5.0.

ANS: Material Design.

m) Among its many new features, Android 6.0 introduced a new __________ to make apps easier to install.

ANS: Permissions model.

1.5 State whether each of the following is *true* or *false*. If *false*, explain why.

a) The vast majority of Android development is done in C++.

ANS: False. The vast majority of Android development is done in Java.

b) You can reproduce on the emulator most of the Android gestures and controls using your computer's keyboard and mouse.

ANS: True.

c) Objects, or more precisely the classes objects come from, are essentially reusable software components.

ANS: True.

1.6 One of the most common objects is a wrist watch. Discuss how each of the following terms and concepts applies to the notion of a watch: object, attributes, behaviors, class, inheritance (consider, for example, an alarm clock), messages, encapsulation and information hiding.

ANS: The entire watch is an object that is composed of many other objects (such as the moving parts, the band, the face, etc.) Watch attributes are time, color, band, style (digital or analog), etc. The behaviors of the watch include setting the time and getting the time. A watch can be considered a specific type of clock (as can an alarm clock). With that in mind, it is possible that a class called `Clock` could exist from which other classes such as watch and alarm clock could inherit the basic features in the clock. The watch is an abstraction of the mechanics needed to keep track of the time. The user of the watch does not need to know the mechanics of the watch in order to use it; the user only needs to know that the watch keeps the proper time. In this sense, the mechanics of the watch are encapsulated (hidden) inside the watch. The interface to the watch (its face and controls for setting the time) allows the user to set and get the time. The user is not allowed to directly touch the internal mechanics of the watch. All interaction with the internal mechanics is controlled by the interface to the watch. The data members stored in the watch are hidden inside the watch and the member functions (looking at the face to get the time and setting the time) provide the interface to the data.