

# Introduction to Android

# 1

## 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.
- Basic object-technology concepts.
- Key software for Android app development, including the Android SDK, the Java SDK, the Eclipse integrated development environment (IDE) and Android Studio.
- Important Android documentation.
- Test-driving an Android drawing app in Eclipse (in the print book) and in Android Studio (online).
- Characteristics of great Android apps.



## Self-Review Exercises

- 1.1** Fill in the blanks in each of the following statements:
- a) App developers can send data from their servers to their apps installed on Android devices even if the apps are *not* running currently using \_\_\_\_\_.  
ANS: Android C2DM.
  - b) \_\_\_\_\_ is a short-range wireless connectivity standard that allows communication between two devices within a few centimeters.  
ANS: near-field communication (NFC).
  - c) \_\_\_\_\_ describe portions of an app's user interface, which can be combined into one screen or used across multiple screens.  
ANS: fragments.
  - 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, pre-defined 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) Using NFC \_\_\_\_\_ allows you to touch two Android devices to share content.  
ANS: Android Beam.
  - 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) Android 2.2 introduced external storage, which allows one to store apps on an external memory device.  
ANS: True.
  - b) Cloud computing allows one to use software and data stored in the local machines.  
ANS: False. It allows you to use software and data stored in the "cloud".
  - c) Java is neither object oriented nor has access to extensive class libraries that help you develop powerful apps quickly.  
ANS: False. Java is object-oriented and has access to extensive class libraries.
  - 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) \_\_\_\_\_ are software components stored on one computer that can be accessed by app (or other software component) on another computer over the Internet.

ANS: web services.

c) Android version 2.3 is also known as \_\_\_\_\_.

ANS: Gingerbread.

d) Touching the screen, moving your finger in a direction and releasing it generates a \_\_\_\_\_ gesture.

ANS: swipe.

e) Before running an app in the emulator, you'll need to create an \_\_\_\_\_, 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 \_\_\_\_\_ 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.

## 4 Chapter 1 Introduction to Android

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) Colors are defined using the RGBA color scheme in which the red, green, blue and \_\_\_\_\_ components are specified by integers in the range 0–255.

**ANS:** alpha.

**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) Microsoft Visual Studio is the recommended integrated development environment for Android development, though developers may also use a text editor and command-line tools to create Android apps.

**ANS:** False. Eclipse or Android Studio is the recommended integrated development environment for Android development, though developers may also use a text editor and command-line tools to create Android apps.

c) Reuse helps you build more reliable systems as existing classes and components have often gone through extensive testing, debugging and performance tuning.

**ANS:** True.

d) An object has attributes that it carries along as it's used in a program. These attributes are specified as part of the object's class.

**ANS:** True.

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

**ANS:** Car is an object that is composed of many other objects (such as the steering type, body type, etc.) Attributes of the car are make, model, color, transmission type (manual or auto), etc. The behaviors of the car include move forward and move backward. A car can be considered a specific engine type (as petrol or diesel). With that in mind, it is possible that a class called Car could exist from which other classes such as petrol engine car and diesel engine car could inherit the basic features in the car. The car is an abstraction of the mechanics, move forward and move backward. The driver of the car does not need to know the mechanics of the car in order to drive it; the driver only needs to know that the car moves forward and backward. In this sense, the mechanics of the car are encapsulated (hidden) inside the Car class. The interface to the car allows the developer to set and get the steering type, body type, etc. The developer is not allowed to directly touch the internal mechanics of the car. All interaction with the internal mechanics is controlled by the interface to the car. The data members stored in the car are hidden and the member functions (to set and get the steering type, body type, etc.) provide the interface to the data.