TRY IT OUT Using the Contacts Content Provider  
codefile Provider.zip available for download at Wrox.com1. Using Eclipse, create a new Android project and name it Provider.  
2. Add the following statements in bold to the main.xml file:  
<?xml version=”1.0” encoding=”utf-8”?>  
<LinearLayout xmlns:android=”http://schemas.android.com/apk/res/android”  
android:layout\_width=”fill\_parent”  
android:layout\_height=”fill\_parent”  
android:orientation=”vertical” >  
<ListViewandroid:id=”@+id/android:list”android:layout\_width=”fill\_parent”android:layout\_height=”wrap\_content”android:layout\_weight=”1”android:stackFromBottom=”false”android:transcriptMode=”normal” /><TextViewandroid:id=”@+id/contactName”android:textStyle=”bold”android:layout\_width=”wrap\_content”android:layout\_height=”wrap\_content” /><TextViewandroid:id=”@+id/contactID”android:layout\_width=”fill\_parent”android:layout\_height=”wrap\_content” /></LinearLayout>  
3. In the ProviderActivity.java class, code the following:  
package net.learn2develop.Provider;  
import android.app.ListActivity;import android.content.CursorLoader;import android.database.Cursor;import android.net.Uri;import android.os.Bundle;  
import android.provider.ContactsContract;import android.widget.CursorAdapter;import android.widget.SimpleCursorAdapter;public class ProviderActivity extends ListActivity {  
/\*\* Called when the activity is first created. \*/  
@Override  
public void onCreate(Bundle savedInstanceState) {  
super.onCreate(savedInstanceState);

setContentView(R.layout.main);  
**Uri allContacts = Uri.*parse*(“content://contacts/people”);  
Cursor c;  
if (android.os.Build.VERSION.SDK\_INT <11) {  
//---before Honeycomb---  
c = managedQuery(allContacts, null, null, null, null);  
} else {  
//---Honeycomb and later---  
CursorLoader cursorLoader = new CursorLoader(  
this,  
allContacts,  
null,  
null,  
null ,  
null);  
c = cursorLoader.loadInBackground();  
}  
String[] columns = new String[] {  
ContactsContract.Contacts.DISPLAY\_NAME,  
ContactsContract.Contacts.\_ID};  
int[] views = new int[] {R.id.contactName, R.id.contactID};  
SimpleCursorAdapter adapter;  
if (android.os.Build.VERSION.SDK\_INT <11) {  
//---before Honeycomb---  
adapter = new SimpleCursorAdapter(  
this, R.layout.main, c, columns, views);  
} else {  
//---Honeycomb and later---  
adapter = new SimpleCursorAdapter(  
this, R.layout.main, c, columns, views,  
CursorAdapter.FLAG\_REGISTER\_CONTENT\_OBSERVER);  
}  
this.setListAdapter(adapter); }**}  
**4.** Add the following statements in bold to the AndroidManifest.xml file:  
<?xml version=“1.0“ encoding=“utf-8“?>  
<manifest xmlns:android=“http://schemas.android.com/apk/res/android“  
package=”net.learn2develop.Provider”  
android:versionCode=”1”  
android:versionName=”1.0” >  
<uses-sdk android:minSdkVersion=”14” />  
**<uses-permission android:name=”android.permission.READ\_CONTACTS”/>**<application

android:icon=”@drawable/ic\_launcher”  
android:label=”@string/app\_name” >  
<activity  
android:label=”@string/app\_name”  
android:name=”.ProviderActivity” >  
<intent-filter >  
<action android:name=”android.intent.action.MAIN” />  
<category android:name=”android.intent.category.LAUNCHER” />  
</intent-filter>  
</activity>  
</application>  
</manifest>  
**5.** Launch an AVD and create a few contacts in the Android Emulator. To add a contact, go to the  
Phone application and click the Star icon at the top (see Figure 7-1). Click the MENU button on  
the emulator and click the New contact menu item. You will be warned about backing up your  
contacts. Click the Keep Local button and enter the name, phone number, and e-mail address of a  
few people.  
**6.** Press F11 to debug the application on the Android emulator. Figure 7-2 shows the activity  
displaying the list of contacts you just created.  
**FIGURE 7-1 FIGURE 7-2**