14

Android
Persistency: Preferences

Victor Matos
Cleveland State University

Notes are based on:

The Busy Coder's Guide to Android Development
by Mark L. Murphy
Copyright © 2008-2009 CommonsWare, LLC.
ISBN: 978-0-9816780-0-9
&
Android Developers
http://developer.android.com/index.html

Android Data Storage

Android provides the following four mechanisms for storing and retrieving data:

1. Preferences,
2. Files,
3. Databases, and
4. Network.

http://developer.android.com/guide/topics/data/data-storage.html

Android Data Storage

A typical desktop operating system provides a common file system that any application can use to store files that *can be read by other applications* (perhaps with some access control settings).

Android uses a different system:

On Android, all application data (including files) are private to that application.

http://developer.android.com/guide/topics/data/data-storage.html

Android Data Storage

Android also provides a standard way for an application to expose its private data to other applications — through **content providers**.

Android supplies a number of content providers for standard data types, such as

image,
audio,
video files and
personal contact information.

Preferences

Preferences is an Android lightweight mechanism to store and retrieve *key-value* pairs of primitive data types (also called *Maps*, and *Associative Arrays.*

Typically used to keep state information and shared data among several activities of an application.

On each entry *<key-value>* the key is a string and the value must be a primitive data type.

Preferences are similar to Bundles however they are persistent while Bundles are not.

5

Preferences

Using Preferences API calls
You have three API choices to pick a Preference:

1. **getPreferences()** from within your Activity, to access activity specific preferences
2. **getSharedPreferences()** from within your Activity to access application-level preferences
3. **getDefaultSharedPreferences()**, on *PreferencesManager*, to get the shared preferences that work in concert with Android's overall preference framework

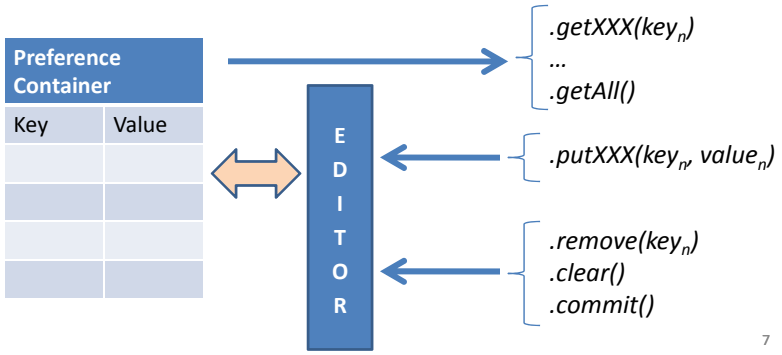
6

Preferences

Using Preferences API calls

All of the get… Preference methods return a Preference object whose contents can be manipulated by an *editor* that allows *putXXX… and getXXX…* commands to place data in and out of the Preference container.

XXX = { *Long, Int, Double, Boolean, String* }



7

Preferences

Example

1. In this example a persistent *SharedPreferences* object is created at the end of an activity lifecycle. It contains data (name, phone, credit, etc. of a fictional customer)

2. The process is interrupted using the "*Back Button*" and re-executed later.

3. Just before been killed, the state of the running application is saved in the designated *Preference* object.

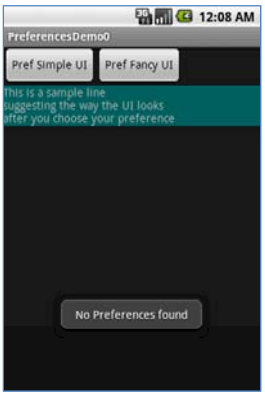
4. When re-executed, it finds the saved *Preference* and uses its persistent data.

Warning
Make sure you test from a 'fresh' configuration. If necessary use DDMS and *delete* existing Preferences held in the application's name-space.

8

Preferences

Example2: Saving/Retrieving a SharedPreference Object holding UI user choices.



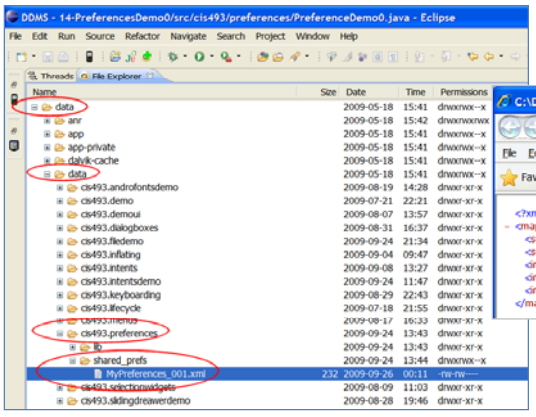
Initial UI with no choices Made/save yet.

Images of the choices made by the user regarding the looks of the UI. The 'green screen' corresponds to the fancy layout, the 'grey screen' is the simple choice. Data is saved into the SharedPreference object: *myPreferences_001.*

9

Preferences

Example2: Saving/Retrieving a SharedPreference Object



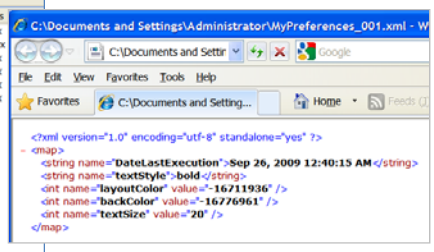


Image of the preference file (obtained by pulling a copy of the file out of the device).

Using DDMS to explore the Device's memory map. Observe the choices made by the user are saved in the *data/data/Shared_prefs/* folder as an XML file.

10

Preferences

Example2: Saving/Retrieving a SharedPreference Object

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
android:id="@+id/linLayout1Vertical"
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:orientation="vertical"
xmlns:android="http://schemas.android.com/apk/res/android" >

<LinearLayout
android:id="@+id/linLayout2Horizontal"
android:layout_width="fill_parent"
android:layout_height="wrap_content" >
      <Button
            android:id="@+id/btnPrefSimple"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Pref Simple UI"  />
      <Button
            android:id="@+id/btnPrefFancy"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Pref Fancy UI" />
</LinearLayout>

<TextView
      android:id="@+id/txtCaption1"
      android:layout_width="fill_parent"
      android:layout_height="wrap_content"
      android:background="#ff006666"
      android:text="This is some sample text " />

</LinearLayout>
```

11

Preferences

Example2: Saving/Retrieving a SharedPreference Object

```
package cis493.preferences;
import java.util.Date;
import android.app.Activity;
import android.content.SharedPreferences;
import android.graphics.Color;
import android.graphics.Typeface;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

public class PreferenceDemo0 extends Activity implements OnClickListener {
     Button btnSimplePref;
     Button btnFancyPref;
     TextView txtCaption1;
     Boolean fancyPrefChosen = false;
     View  myLayout1Vertical;
     final int mode = Activity.MODE_PRIVATE;
     final String MYPREFS = "MyPreferences_001";
     // create a reference to the shared preferences object
     SharedPreferences mySharedPreferences;
     // obtain an editor to add data to my SharedPreferences object
     SharedPreferences.Editor myEditor;
```

12

Preferences

Example2: Saving/Retrieving a SharedPreference Object

```
@Override
public void onCreate(Bundle savedInstanceState) {
     super.onCreate(savedInstanceState);
     setContentView(R.layout.main);
     myLayout1Vertical = (View)findViewById(R.id.linLayout1Vertical);
     txtCaption1 = (TextView) findViewById(R.id.txtCaption1);
     txtCaption1.setText("This is a sample line \n“
                         + "suggesting the way the UI looks \n"
                         + "after you choose your preference");
     // create a reference & editor for the shared preferences object
     mySharedPreferences = getSharedPreferences(MYPREFS, 0);
     myEditor = mySharedPreferences.edit();
     // has a Preferences file been already created?
     if (mySharedPreferences != null
        && mySharedPreferences.contains("backColor")) {
        // object and key found, show all saved values
            applySavedPreferences();
        } else {
            Toast.makeText(getApplicationContext(),
                           "No Preferences found", 1).show();
     }
     btnSimplePref = (Button) findViewById(R.id.btnPrefSimple);
     btnSimplePref.setOnClickListener(this);
     btnFancyPref = (Button) findViewById(R.id.btnPrefFancy);
     btnFancyPref.setOnClickListener(this);
}// onCreate
```

Preferences

Example2: Saving/Retrieving a SharedPreference Object

```
@Override
public void onClick(View v) {
     // clear all previous selections
     myEditor.clear();

     // what button has been clicked?
     if (v.getId() == btnSimplePref.getId()) {
          myEditor.putInt("backColor", Color.BLACK);// black background
          myEditor.putInt("textSize", 12);   // humble small font
     } else { // case btnFancyPref
          myEditor.putInt("backColor", Color.BLUE); // fancy blue
          myEditor.putInt("textSize", 20);   // fancy big
          myEditor.putString("textStyle", "bold");  // fancy bold
          myEditor.putInt("layoutColor", Color.GREEN);//fancy green
     }
     myEditor.commit();
     applySavedPreferences();
}
```

Preferences

Example2: Saving/Retrieving a SharedPreference Object

```
@Override
protected void onPause() {
    // warning: activity is on its last state of visibility!.
    // It's on the edge of been killed! Better save all current
    // state data into Preference object (be quick!)
    myEditor.putString("DateLastExecution", new Date().toLocaleString());
    myEditor.commit();
    super.onPause();
}
```

15

Preferences

Example2: Saving/Retrieving a SharedPreference Object

```
public void applySavedPreferences() {
    // extract the <key/value> pairs, use default param for missing data
    int backColor = mySharedPreferences.getInt("backColor",Color.BLACK);
    int textSize = mySharedPreferences.getInt("textSize", 12);
    String textStyle = mySharedPreferences.getString("textStyle", "normal");
    int layoutColor = mySharedPreferences.getInt("layoutColor",
    Color.DKGRAY);
    String msg = "color " + backColor + "\n"
              + "size "   + textSize    + "\n"
              + "style "  + textStyle;
    Toast.makeText(getApplicationContext(), msg, 1).show();

    txtCaption1.setBackgroundColor(backColor);
    txtCaption1.setTextSize(textSize);
    if (textStyle.compareTo("normal")==0){
       txtCaption1.setTypeface(Typeface.SERIF,Typeface.NORMAL);
    }
    else {
       txtCaption1.setTypeface(Typeface.SERIF,Typeface.BOLD);
    }
    myLayout1Vertical.setBackgroundColor(layoutColor);
}// applySavedPreferences

}//class
```

Preferences

Example3: Saving/Retrieving a SharedPreference Object containing 'business' data.

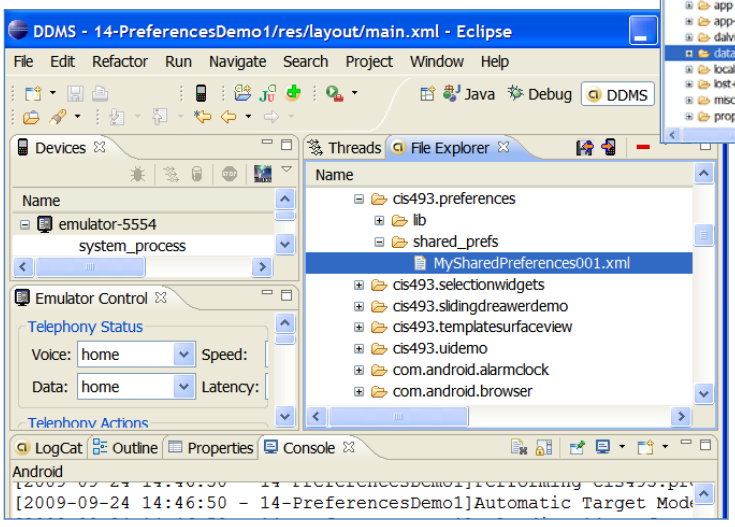
Image of the data held in the SharedPreferences object displayed the first time the Activity **Preferences1** is executed.

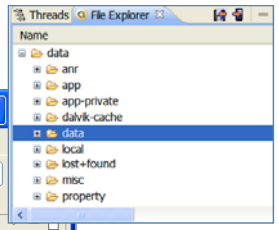
Image of the saved Preference data displayed the second time the Activity **Preferences1** is executed.

17

Preferences

Example: Saving/Retrieving a SharedPreference Object



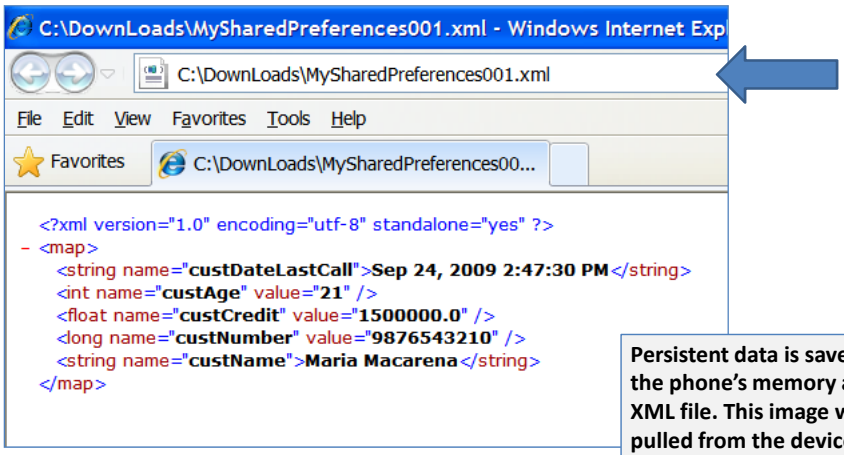


Use DDMS to see persistent data set

18

Preferences

Example: Saving/Retrieving a SharedPreference Object



Persistent data is saved in the phone's memory as an XML file. This image was pulled from the device using DDMS.

Preferences

Example: Saving/Retrieving a SharedPreference Object

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
android:id="@+id/linLayou1"
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:background="#ff0000ff"
android:orientation="vertical"
xmlns:android="http://schemas.android.com/apk/res/android"
>
<TextView
    android:id="@+id/captionBox"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="SharedPreferences Container: Customer Data"
    android:layout_margin="5px" android:textStyle="bold">
</TextView>
<EditText
    android:id="@+id/txtPref"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10px"
    >
</EditText>
</LinearLayout>
```

Preferences

Example: Saving/Retrieving a SharedPreference Object

```
package cis493.preferences;

import java.util.Date;

import android.app.Activity;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.widget.*;

public class Preference1 extends Activity {
    public static final String MYPREFS = "MySharedPreferences001";
    //this data values describe a typical customer record
    String custName = "n.a.";
    int    custAge = 0;
    float  custCredit = 0;
    long   custNumber = 0;
    String custDateLastCall;

    TextView captionBox;
    EditText txtPref;
    final int mode = Activity.MODE_PRIVATE;
```


Preferences

Example: Saving/Retrieving a SharedPreference Object

```
  @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        txtPref = (EditText)findViewById(R.id.txtPref);
        captionBox = (TextView) findViewById(R.id.captionBox);
        captionBox.setText("SharedPreference Container: \n\n"+
                "we are working on customer Macarena \n" +
                "fake an interruption, press 'Back Button' \n" +
                "re-execute the application.");

        //create a reference to the shared preferences object
        int mode = Activity.MODE_PRIVATE;
        SharedPreferences mySharedPreferences = getSharedPreferences(MYPREFS, mode);
        //is there an existing Preferences from previous executions of this app?
        if (mySharedPreferences != null  &&
            mySharedPreferences.contains("custName")) {
            //object and key found, show all saved values
            showSavedPreferences();
        }
        else
        {
            txtPref.setText("nada");
        }
    }//onCreate
```

Preferences

Example: Saving/Retrieving a SharedPreference Object

```
@Override
protected void onPause() {
   //warning: activity is on last state of visibility! We are on the
   //edge of been killed! Better save current state in Preference object
   savePreferences();
   super.onPause();
}

protected void savePreferences(){
    //create the shared preferences object
    SharedPreferences mySharedPreferences =
                          getSharedPreferences(MYPREFS, mode);

    //obtain an editor to add data to (my)SharedPreferences object
    SharedPreferences.Editor myEditor = mySharedPreferences.edit();

    //put some <key/value> data in the preferences object
    myEditor.putString("custName", "Maria Macarena");
    myEditor.putInt("custAge", 21);
    myEditor.putFloat("custCredit", 1500000.00F);
    myEditor.putLong("custNumber", 9876543210L);
    myEditor.putString("custDateLastCall", new Date().toLocaleString());
    myEditor.commit();
}//savePreferences
```

Preferences

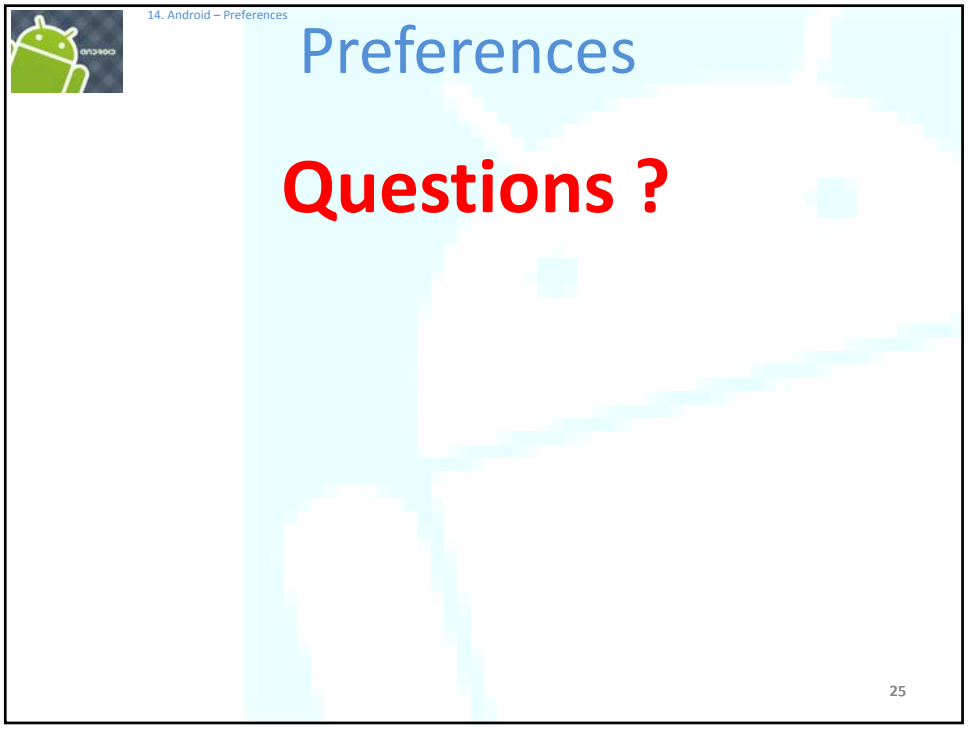
Example: Saving/Retrieving a SharedPreference Object

```
public void showSavedPreferences() {
      //retrieve the SharedPreferences object

      SharedPreferences mySharedPreferences =
                              getSharedPreferences(MYPREFS, mode);

      //extract the <key/value> pairs, use default param for missing data
      custName = mySharedPreferences.getString("custName", "defNameValue");
      custAge = mySharedPreferences.getInt("custAge", 18);
      custCredit = mySharedPreferences.getFloat("custCredit", 1000.00F);
      custNumber = mySharedPreferences.getLong("custNumber", 1L);
      custDateLastCall = mySharedPreferences.getString("custDateLastCall",
                                            new Date().toLocaleString());
      //show saved data on screen
      String msg = "name: " + custName + "\nAge: " + custAge +
                   "\nCredit: " + custCredit +
                   "\nLastCall: " + custDateLastCall;
      txtPref.setText(msg);
   }//loadPreferences

}//Preferences1
```


14. Android – Preferences
Preferences
Questions ?
25