**第六章 详解持久化技术**

**1. 持久化简介**

**数据持久化就是指将那些内存中的瞬时数据保存到存储设备中, 保证即使在手机或电脑关机的情况下, 这些数据仍然不会丢失。久化技术则提供了一种机制可以让数据在瞬时状态和持久状态之间进行转換。Android系统中主要提供了3种方式用于简単地实现数据持久化功能, 即文件存储、 sharedPreference存储以及数据库存储。**

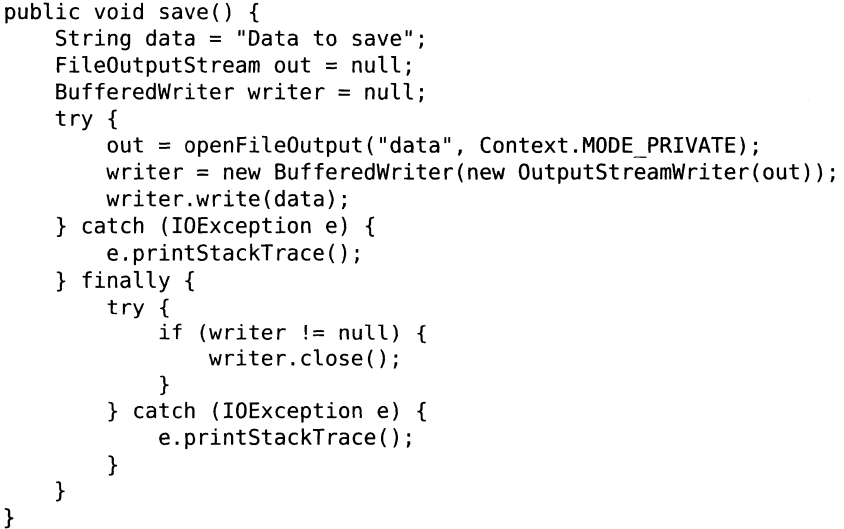
**2. 文件存储**

**（1）将数据存储到文件中**

**Context类中提供了一个 openFileOutput()方法,可以用于将数据存储到指定的文件中。这个方法接收两个参数,第一个参数是文件名，第二个参数是文件的操作模式, 主要有两种模式可选, MODE\_PRIVATE和 MODE\_APPEND。**

**其中 MODE\_PRIVATE是默认的操作模式,表示当指定同样文件名的时候,所写入的内容将会覆盖原文件中的内容,而 MODE\_APPEND则表示如果该文件已存在, 就往文件里面追加内容,不存在就创建新文件。**

**openFileOutput()方法返回的是一个 FileOutputStream对象,得到了这个对象之后就可以使用 Java流的方式将数据写入到文件中。将一段文本文件保存到文件中，代码如下所示：**

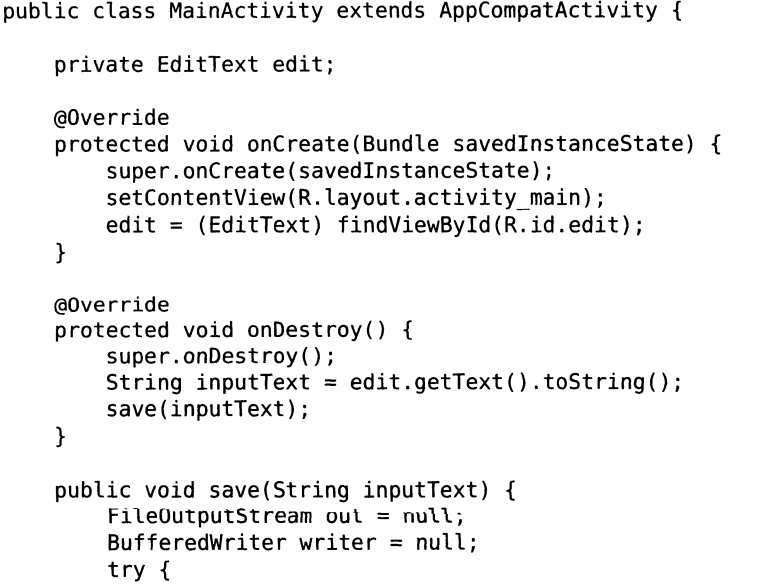


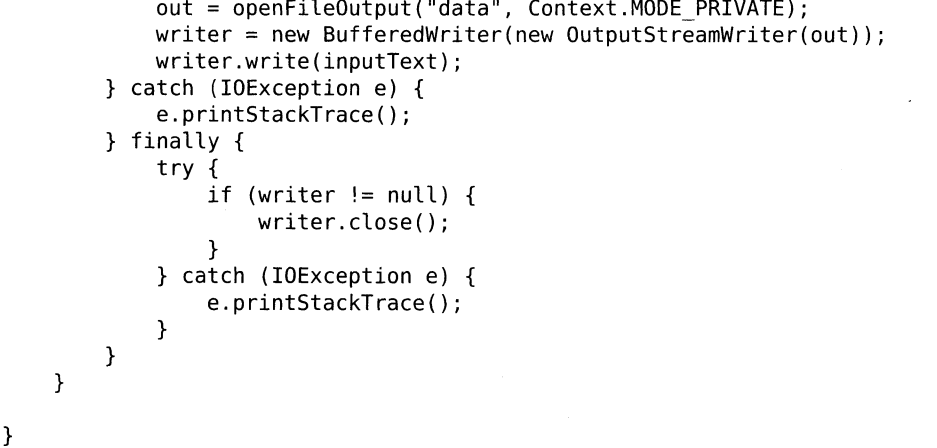
**这里通过 openFileOutput( )方法能够得到一个 FileOutputStream 对象, 然后再借助它构建出一个 OutputStreamWriter对象,接着再使用 OutputStreamWriter构建出一个 BufferedWriter对象,这样就可以通过 BufferedWriter来将文本内容写入到文件中了。**

**下面是一个完整的在 Android项目中使用文件存储例子,** **首先创建一个 FilePersistenceTest项目, 并修改 activity\_main.xm1中的代码, 如下所示:**



**这里在布局中加入了一个EditText。修改MainActivity中的代码，如下所示：**

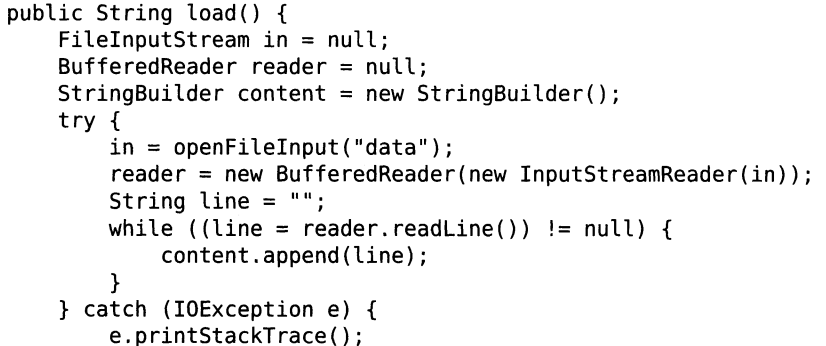


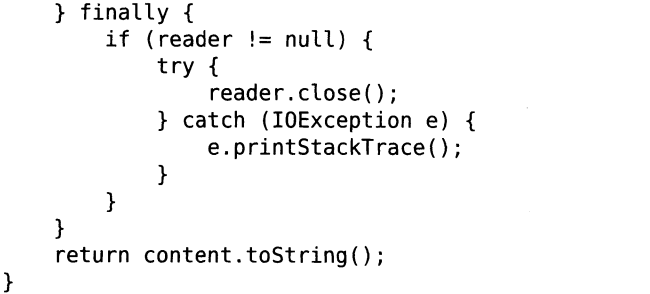


**这里首先在 onCreate()方法中获取了 EditText的实例,然后重写了 onDestroy()** **方法, 这样就可以保证在活动销毁之前一定会调用这个方法 。 在 onDestroy()方法中获取EditText中输入的内容,并调用 save()方法把输入的内容存储到文件中,文件命名为 data。**

**（2）从文件中读取数据**

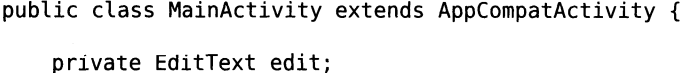
**context类中还提供了一个 openFileInput()方法,用于从文件中读取数据。它只接收一个参数，即要读取的文件名，并返回一个FileInputStream对象，在通过java流的方式就可以将数据读取出来了。从文件中读取文本数据，代码如下所示:**





**首先通过 openFileInput()方法获取到了一个 FileInputstream对象,然后借助它又构建出了一个 InputStreamReader对象, 接着再使用 InputstreamReader构建出一个 BufferedReader对象,这样就可以通过 BufferedReader进行一行行地读取,把文件中所有的文本内容全部读取出来,并存放在一个 stringBunder对象中,最后将读取到的内容返回。**

**修改（1）中的例子, 使得重新启动程序时 EditText中能够保留上次输入的内容。修改 MainActivity 中的代码,如下所示：**





**在 oncreate()方法中调用load()方法来读取文件中存储的文本内容,如果读到的内容不为 null,就调用 EditText的 setText()方法将内容填充到 EditText里,并调用 setselection()方法将输入光标移动到文本的末尾位置以便于继续输入,然后弹出一句还原成功的提示。**

**3. SharedPreferences存储**

**SharedPreferences使用键值得方式来存储数据，给保存的数据提供一个对应的键，在读取数据时通过相应的键值把数据读取出来。SharedPreferences还支持多种不同数据类型的存储。**

1. **将数据存储到SharedPreferences中**

**要想使用 SharedPreferences来存储数据,首先需要获取到 SharedPreferences对象。Android中主要提供了3种方法用于得到 sharedPreferences对象。**

1. **Context类中的 getSharedPreferences()方法**

**此方法接收两个参数,第一个参数用于指定 SharedPreferences文件的名称,** **第二个参数用于指定操作模式, 目前只有 MODE\_PRIVATE这一种模式可选。**

1. **Activity类中的 getPreferences()方法**

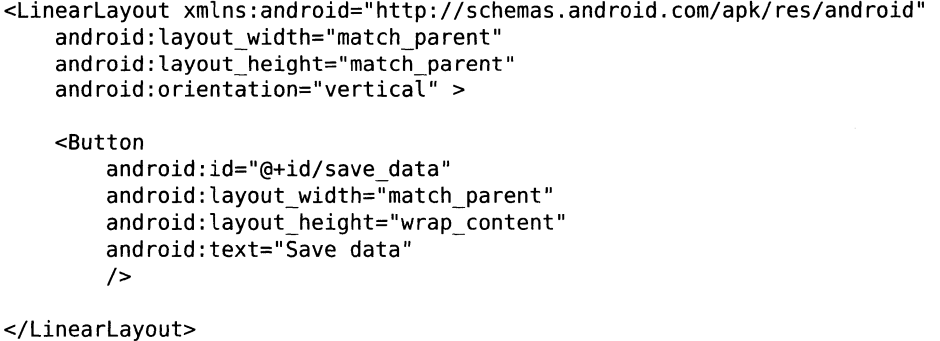
**只接收一个操作模式参数,使用这个方法时会自动将当前活动的类名作为 SharedPreferences的文件名。**

1. **PreferenceManager类中的 getDefaultSharedPreferences()方法**

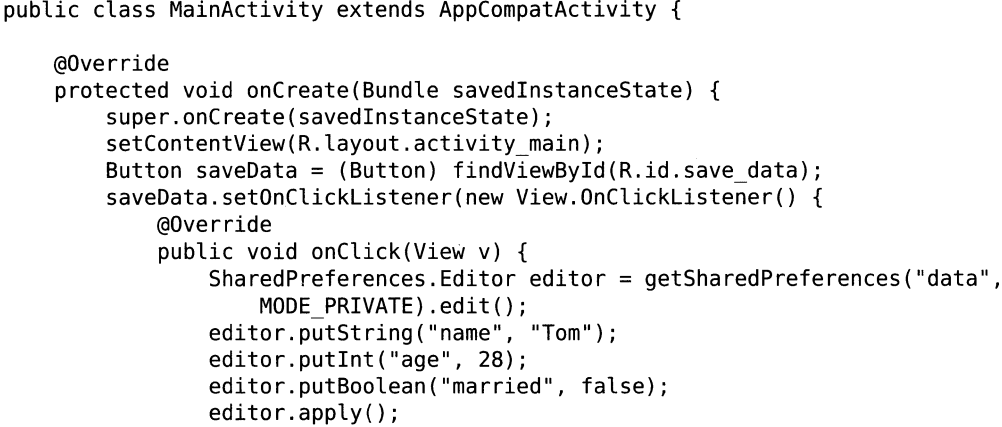
**这是一个静态方法,它接收一个 context参数,并自动使用当前应用程序的包名作为前缀来命名 SharedPreferences文件。向 SharedPreferences文件中存储数据, 主要可以分为3步实现:**

1. **调用 SharedPreferences对象的 edit()方法来获取一个 SharedPreferences.Editor对象。**
2. **向 SharedPreferences.Editor对象中添加数据,添加一个布尔型数据就使用putBoolean()方法, 添加一个字符串则使用 putString()方法, 以此类推。**
3. **调用 apply()方法将添加的数据提交, 从而完成数据存储操作。**

**新建一个 SharedPreferencesTest项目,然后修改 activity\_main.xm1中的代码, 如下所示:**



**这里放置了一个按钮用于将一些数据存储到Sharedpreferences文件当中。然后修改 MainActivity中的代码, 如下所示:**



**}**

**});**

**}**

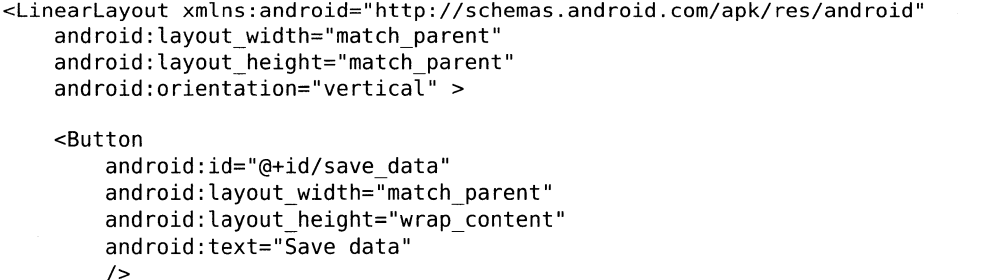
**}**

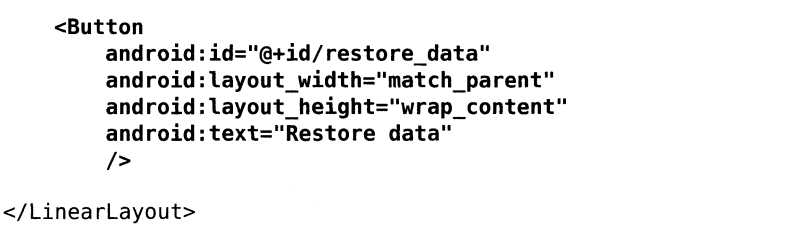
**这里首先给按钮注册了一个点击事件, 然后在点击事件中通过 getsharedPreferences()方法指定 SharedPreferences的文件名为 data,并得到了SharedPreferences.Editor对象。接着向这个对象中添加了3条不同类型的数据,最后调用 apply()方法进行提交,从而完成了数据存储的操作。**

1. **从SharedPreference中读取数据**

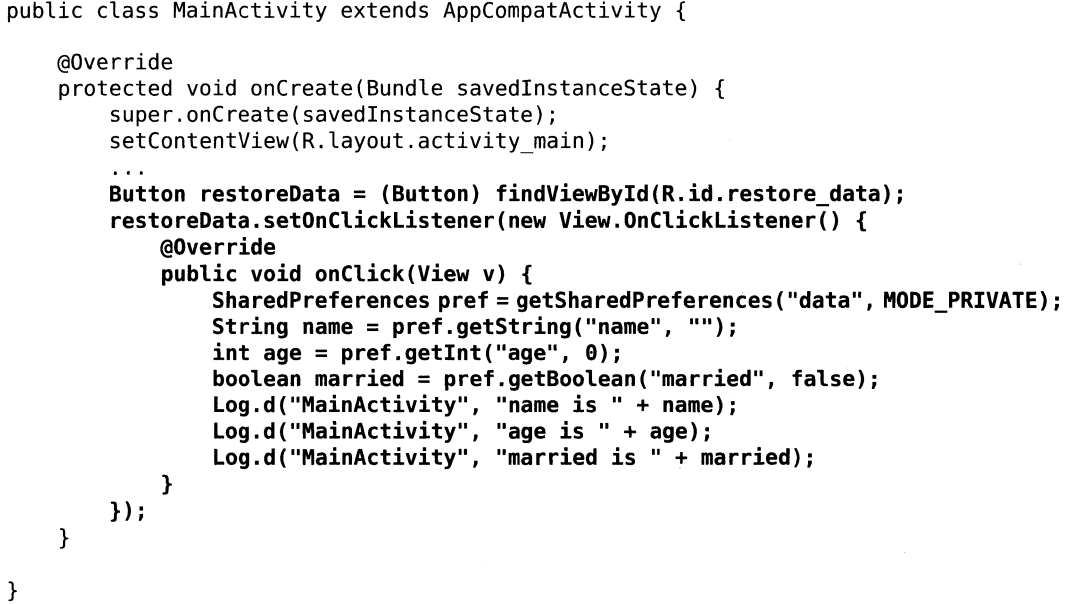
**SharedPreferences对象中提供了一系列的get方法,用于对存储的数据进行读取,每种 get方法都对应了 sharedPreferences.Editor中的一种 put方法。这些get方法都接收两个参数，第一个参数是键，第二个参数是默认值，即当传入的键找不到对应的值时会以什么样的默认値进行返回。**

**在 SharedPreferencesTest项目的基础上继续开发,修改 activity\_main.xm1中的代码,如下所示：**





**这里增加了一个还原数据的按钮,通过点击这个按钮来从 SharedPreferences文件读取数据。修改 MainActivity中的代码,如下所示:**



**首先通过getSharedPreferences()方法得到了 SharedPreferences对象,然后分别调用它的 getString()、 getInt()和 getBoolean()方法,去获取前面所存储的姓名、年龄和是否已婚,如果没有找到相应的值,就会使用方法中传入的默认值来代替,最后通过 Log将这些值打印出来。**

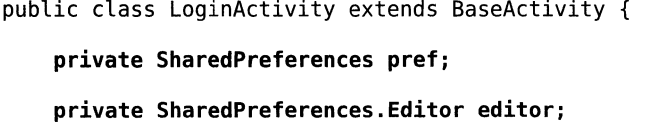
1. **实现记住密码功能**

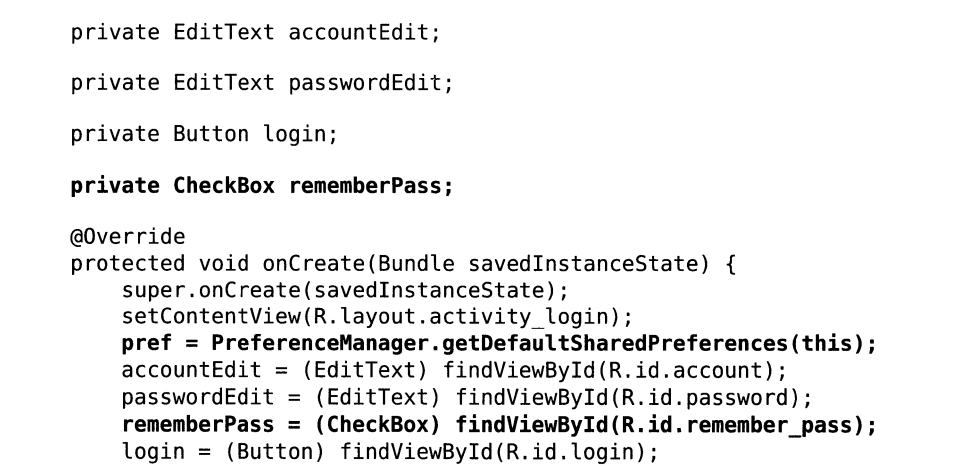
**打开 BroadcastBestPractice项目,编辑登录界面的布局。修改 activity\_1ogin.xm1中的代码,如下所示：**



**这里使用了一个新控件 checkBox。这是一个复选框控件,用户可以通过点击的方式来进行选中和取消, 这里使用这个控件来表示用户是否需要记住密码。**

**然后修改 LoginActivity中的代码, 如下所示:**







**}**

**}**

**这里首先在 onCreate()方法中获取到了sharedPreferences对象,然后调用它的 getBoolean()方法去获取 remember\_password这个键对应的值。由于不存在对应值，所以使用默认值false。接着在登录成功之后,会调用checkBox的isChecked()方法来检查复选框是否被选中,被选中则表示用户想要记住密码，这时将remember\_password设置为 true,然后把 account和 password对应的值都存入到 SharedPreferences文件当中并提交。如果没有被选中,就简単地调用一下 clear( )方法, 将SharedPreferences文件中的数据全部清除掉。**

**\*\*\*\*\*\*\*\*\*\***

**将密码以明文的形式存储在 SharedPreferences文件中是非常不安全的,很容易就会被别人盗取, 因此在正式的项目里还需要结合一定的加密算法来对密码进行保护。**

**\*\*\*\*\*\*\*\*\*\*\*\*\***