使用SharedPreferences进行数据存储

很多时候我们开发的软件需要向用户提供软件参数设置功能，例如我们常用的QQ，用户可以设置是否允许陌生人添加自己为好友。对于软件配置参数的保存，如果是window软件通常我们会采用ini文件进行保存，如果是j2se应用，我们会采用properties属性文件或者xml进行保存。如果是Android应用，我们最适合采用什么方式保存软件配置参数呢？Android平台给我们提供了一个SharedPreferences类，它是一个轻量级的存储类，特别适合用于保存软件配置参数。使用SharedPreferences保存数据，其背后是用xml文件存放数据，文件存放在/data/data/<package name>/shared\_prefs目录下：

SharedPreferences sharedPreferences = getSharedPreferences("itcast", Context.MODE\_PRIVATE);

Editor editor = sharedPreferences.edit();//获取编辑器

editor.putString("name", "传智播客");

editor.putInt("age", 4);

editor.commit();//提交修改

生成的itcast.xml文件内容如下：

<?xml version='1.0' encoding='utf-8' standalone='yes' ?>

<map>

<string name="name">传智播客</string>

<int name="age" value="4" />

</map>

因为SharedPreferences背后是使用xml文件保存数据，getSharedPreferences(name,mode)方法的第一个参数用于指定该文件的名称，名称不用带后缀，后缀会由Android自动加上。方法的第二个参数指定文件的操作模式，共有四种操作模式，这四种模式前面介绍使用文件方式保存数据时已经讲解过。如果希望SharedPreferences背后使用的xml文件能被其他应用读和写，可以指定Context.MODE\_WORLD\_READABLE和Context.MODE\_WORLD\_WRITEABLE权限。

另外Activity还提供了另一个getPreferences(mode)方法操作SharedPreferences，这个方法默认使用当前类不带包名的类名作为文件的名称。

访问SharedPreferences中的数据代码如下：

SharedPreferences sharedPreferences = getSharedPreferences("itcast", Context.MODE\_PRIVATE);

//getString()第二个参数为缺省值，如果preference中不存在该key，将返回缺省值

String name = sharedPreferences.getString("name", "");

int age = sharedPreferences.getInt("age", 1);

如果访问其他应用中的Preference，前提条件是：该preference创建时指定了Context.MODE\_WORLD\_READABLE或者Context.MODE\_WORLD\_WRITEABLE权限。如：有个<package name>为cn.itcast.action的应用使用下面语句创建了preference。

getSharedPreferences("itcast", Context.*MODE\_WORLD\_READABLE);*

其他应用要访问上面应用的preference，首先需要创建上面应用的Context，然后通过Context 访问preference ，访问preference时会在应用所在包下的shared\_prefs目录找到preference ：

Context otherAppsContext = createPackageContext("cn.itcast.action", Context.*CONTEXT\_IGNORE\_SECURITY*);

SharedPreferences sharedPreferences = otherAppsContext.getSharedPreferences("itcast", Context.MODE\_WORLD\_READABLE);

String name = sharedPreferences.getString("name", "");

int age = sharedPreferences.getInt("age", 0);

如果不通过创建Context访问其他应用的preference，也可以以读取xml文件方式直接访问其他应用preference对应的xml文件，如：

File xmlFile = new File(“/data/data/<package name>/shared\_prefs/itcast.xml”);//<package name>应替换成应用的包名