数据存储与访问

很多时候我们的软件需要对处理后的数据进行存储或再次访问。Android为数据存储提供了多种方式，分别有如下几种：

* 文件
* SharedPreferences(参数)
* SQLite数据库
* 内容提供者（Content provider）
* 网络

使用文件进行数据存储

首先给大家介绍使用文件如何对数据进行存储，Activity提供了openFileOutput()方法可以用于把数据输出到文件中，具体的实现过程与在J2SE环境中保存数据到文件中是一样的。

public class FileActivity extends Activity {

@Override public void onCreate(Bundle savedInstanceState) {

...

FileOutputStream outStream = this.openFileOutput("itcast.txt", Context.MODE\_PRIVATE);

outStream.write("传智播客".getBytes());

outStream.close();

}

}

openFileOutput()方法的第一参数用于指定文件名称，不能包含路径分隔符“/” ，如果文件不存在，Android 会自动创建它。创建的文件保存在/data/data/<package name>/files目录，如： /data/data/cn.itcast.action/files/itcast.txt ，通过点击Eclipse菜单“Window”-“Show View”-“Other”，在对话窗口中展开android文件夹，选择下面的File Explorer视图，然后在File Explorer视图中展开/data/data/<package name>/files目录就可以看到该文件。

openFileOutput()方法的第二参数用于指定操作模式，有四种模式，分别为： Context.MODE\_PRIVATE = 0

Context.MODE\_APPEND = 32768

Context.MODE\_WORLD\_READABLE = 1

Context.MODE\_WORLD\_WRITEABLE = 2

**Context.MODE\_PRIVATE**：为默认操作模式，代表该文件是私有数据，只能被应用本身访问，在该模式下，写入的内容会覆盖原文件的内容，如果想把新写入的内容追加到原文件中。可以使用Context.MODE\_APPEND

**Context.MODE\_APPEND**：模式会检查文件是否存在，存在就往文件追加内容，否则就创建新文件。

Context.MODE\_WORLD\_READABLE和Context.MODE\_WORLD\_WRITEABLE用来控制其他应用是否有权限读写该文件。

**MODE\_WORLD\_READABLE**：表示当前文件可以被其他应用读取；**MODE\_WORLD\_WRITEABLE**：表示当前文件可以被其他应用写入。

如果希望文件被其他应用读和写，可以传入：

openFileOutput("itcast.txt", Context.MODE\_WORLD\_READABLE + Context.MODE\_WORLD\_WRITEABLE);

android有一套自己的安全模型，当应用程序(.apk)在安装时系统就会分配给他一个userid，当该应用要去访问其他资源比如文件的时候，就需要userid匹配。默认情况下，任何应用创建的文件，sharedpreferences，数据库都应该是私有的（位于/data/data/<package name>/files），其他程序无法访问。除非在创建时指定了Context.MODE\_WORLD\_READABLE或者Context.MODE\_WORLD\_WRITEABLE ，只有这样其他程序才能正确访问。

读取文件内容

如果要打开存放在/data/data/<package name>/files目录应用私有的文件，可以使用Activity提供openFileInput()方法。

FileInputStream inStream = this.getContext().openFileInput("itcast.txt");

Log.i("FileTest", readInStream(inStream));

readInStream()的方法请看本页下面备注。

或者直接使用文件的绝对路径：

File file = new File("/data/data/cn.itcast.action/files/itcast.txt");

FileInputStream inStream = new FileInputStream(file);

Log.i("FileTest", readInStream(inStream));

注意：上面文件路径中的“cn.itcast.action”为应用所在包，当你在编写代码时应替换为你自己应用使用的包。

对于私有文件只能被创建该文件的应用访问，如果希望文件能被其他应用读和写，可以在创建文件时，指定Context.MODE\_WORLD\_READABLE和Context.MODE\_WORLD\_WRITEABLE权限。

Activity还提供了getCacheDir()和getFilesDir()方法：

getCacheDir()方法用于获取/data/data/<package name>/cache目录

getFilesDir()方法用于获取/data/data/<package name>/files目录

public static String readInStream(FileInputStream inStream){

try {

ByteArrayOutputStream outStream = new ByteArrayOutputStream();

byte[] buffer = new byte[1024];

int length = -1;

while((length = inStream.read(buffer)) != -1 ){

outStream.write(buffer, 0, length);

}

outStream.close();

inStream.close();

return outStream.toString();

} catch (IOException e) {

Log.i("FileTest", e.getMessage());

}

return null;

}

把文件存放在SDCard

使用Activity的openFileOutput()方法保存文件，文件是存放在手机空间上，一般手机的存储空间不是很大，存放些小文件还行，如果要存放像视频这样的大文件，是不可行的。对于像视频这样的大文件，我们可以把它存放在SDCard。 SDCard是干什么的？你可以把它看作是移动硬盘或U盘。

在模拟器中使用SDCard，你需要先创建一张SDCard卡（当然不是真的SDCard，只是镜像文件）。创建SDCard可以在Eclipse创建模拟器时随同创建，也可以使用DOS命令进行创建，如下：

在Dos窗口中进入android SDK安装路径的tools目录，输入以下命令创建一张容量为2G的SDCard，文件后缀可以随便取，建议使用.img：

mksdcard 2048M D:\AndroidTool\sdcard.img

**在程序中访问SDCard，你需要申请访问SDCard的权限。**

在AndroidManifest.xml中加入访问SDCard的权限如下:

<!-- 在SDCard中创建与删除文件权限 -->

<uses-permission android:name="android.permission.MOUNT\_UNMOUNT\_FILESYSTEMS"/>

<!-- 往SDCard写入数据权限 -->

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE"/>

要往SDCard存放文件，程序必须先判断手机是否装有SDCard，并且可以进行读写。

注意：访问SDCard必须在AndroidManifest.xml中加入访问SDCard的权限

if(Environment.getExternalStorageState().equals(Environment.MEDIA\_MOUNTED)){

File sdCardDir = Environment.getExternalStorageDirectory();//获取SDCard目录

File saveFile = new File(sdCardDir, “itcast.txt”);

FileOutputStream outStream = new FileOutputStream(saveFile);

outStream.write("传智播客".getBytes());

outStream.close();

}

Environment.getExternalStorageState()方法用于获取SDCard的状态，如果手机装有SDCard，并且可以进行读写，那么方法返回的状态等于Environment.MEDIA\_MOUNTED。

Environment.getExternalStorageDirectory()方法用于获取SDCard的目录，当然要获取SDCard的目录，你也可以这样写：

File sdCardDir = new File("/mnt/sdcard"); //获取SDCard目录

File saveFile = new File(sdCardDir, "itcast.txt");

//上面两句代码可以合成一句： File saveFile = new File("/mnt/sdcard/itcast.txt");

FileOutputStream outStream = new FileOutputStream(saveFile);

outStream.write("传智播客test".getBytes());

outStream.close();