Android数据的存储

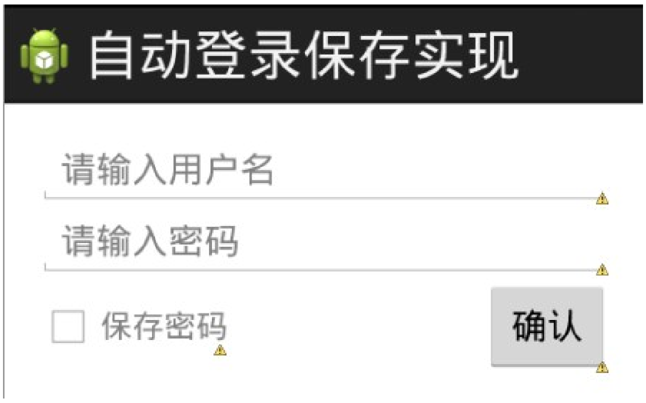
内部存储空间：

1. RAM内存：运行内存，相当于电脑的内存
2. ROM内存：存储内存，相当于电脑的硬盘

外部存储空间：

1. SD卡：相当于电脑的移动硬盘

案例实践：通过对ROM的读写完成登录保存功能(截图如下)：



相应的UI代码如下：

```
<EditText
    android:id="@+id/txt_password"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="请输入密码" />

<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <CheckBox android:id="@+id/chk_save"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerVertical="true"
        android:hint="保存密码" />
    <Button
        android:onClick="btn_login"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:text="确认" />
</RelativeLayout>
```

把用户名与密码,保存到ROM的代码如下:

```
// 1: 获取用户名密码
EditText txt_username = (EditText) findViewById(R.id.txt_username);
EditText txt_password = (EditText) findViewById(R.id.txt_password);
CheckBox chk_save = (CheckBox) findViewById(R.id.chk_save);
String name = txt_username.getText().toString();
String pass = txt_password.getText().toString();
```

```
if (chk_save.isChecked()) {
 // 2: 保存密码
 File file = new File(getFilesDir(), "user.txt");
 OutputStream outStream = null;
 try {
  outStream = new FileOutputStream(file);
  outStream.write((name + "," + pass).getBytes());
 } catch (Exception e) {
  throw new RuntimeException();
 } finally {
  try {
   outStream.close();
  } catch (Exception e) {
   throw new RuntimeException();
  }
 }
}
```

从文件中读取相应的数据代码如下：

```
File file = new File(getFilesDir(), "user.txt");
if (file.exists()) {
 FileInputStream inputStream = null;
 try {
  inputStream = new FileInputStream(file);
        // 返回下一次对此输入流调用的方法可以不受阻塞地从此输入流读取（或跳过）的估计剩余字节数
  byte[] b = new byte[input.available()];
  int len=inputStream.read(b);
  String data = new String(b,0,len);
  EditText txt_username = (EditText) findViewById(R.id.txt_username);
  txt_username.setText(data);
 } catch (Exception e) {
  throw new RuntimeException();
 } finally {
  try {
   inputStream.close();
  } catch (Exception e) {
   throw new RuntimeException();
  }
 }
}
```