电子科技大学信息与软件工程学院

**实 验 报 告**

学 号 2017221303023

姓 名 陈奎

（实验） 课程名称 移动计算

理论教师 许毅

实验教师 许毅

**电子科技大学教务处制表**

**电 子 科 技 大 学**

**实 验 报 告**

**学生姓名：陈奎 学号：2017221303023 指导教师：许毅**

**实验地点：信软楼西304 实验时间：19.11.8**

**一、实验名称：**移动端数据存取

**二、实验学时：**2学时

**三、实验目的：**

1. 学习并掌握轻量级关系型数据库SQLite
2. 学习并掌握数据库的操作方式
3. 理解数据库的ACID原则
4. 学习并掌握SQLite数据库的使用方式
5. 初步了解数据库的调试和测试方法

**四、实验原理：**

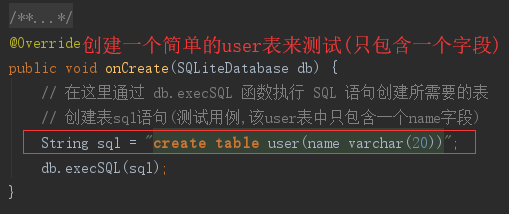
基于AndroidStdio进行一系列的产品开发实践。

**五、实验内容：**

界面设计

|  |  |
| --- | --- |
|  | Screenshot_20191030_113112_com |

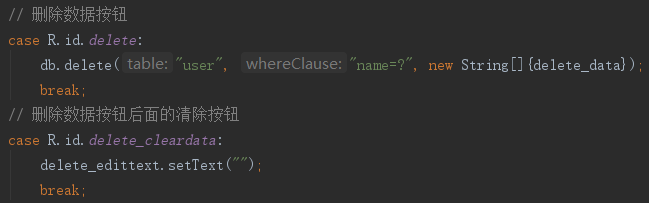
1. 编写程序在Android系统中创建数据库



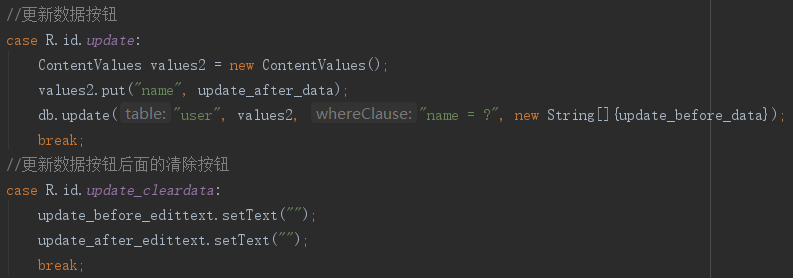
1. 向数据库中添加数据



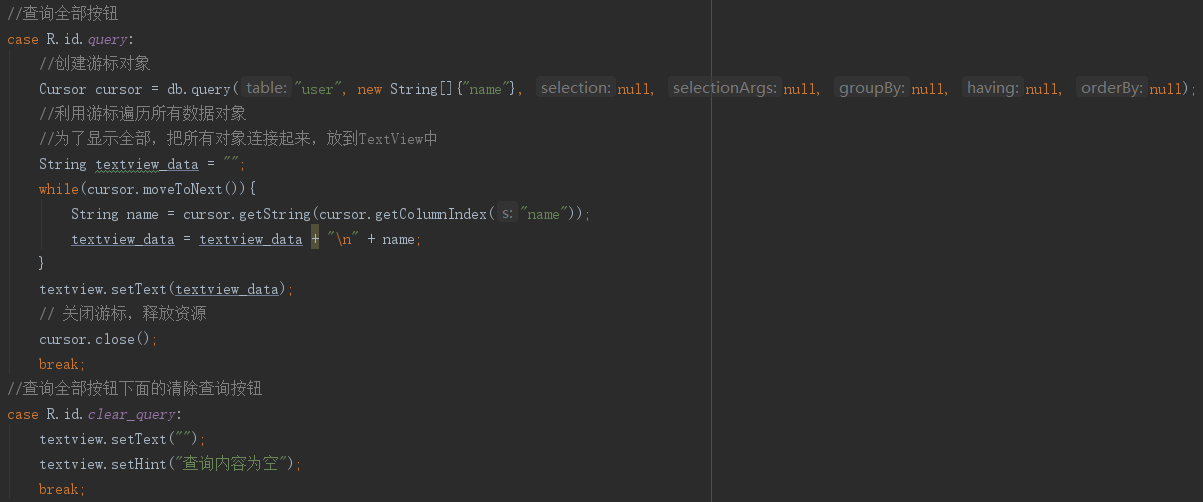
1. 在数据库中删除数据



1. 在数据库中更新数据



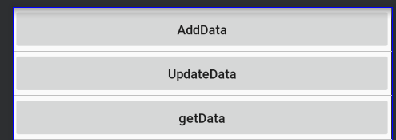
1. 查询数据库中的内容

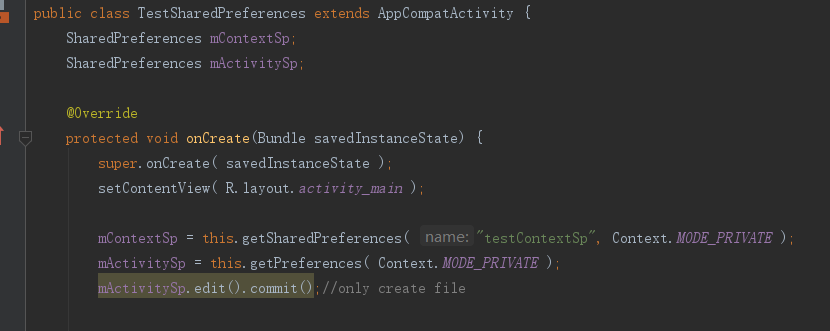


1. 使用SharedPreferences存储

SharedPreferences适用于保存少量数据，且这些数据的格式都是基本类型的值。保存数据基于XML文件存储的key-value键值对数据。

**优点**  
SharedPreferences对象与SQLite数据库相比显得格外轻量级，免去了创建数据库，创建表，写SQL语句等诸多操作，相对而言更加方便，简洁。  
**缺点**  
只能存储boolean，int，float，long和String五种简单的数据类型。  
无法进行条件查询等。





1. 编译、调试和查看程序运行结果

**六、实验器材（设备、元器件）：**

本实验开设方式为个人实验；实验授课时间0.5学时，上机3.5学时，每人一台PC单独完成实验，使用Eclipse（或Android Studio）、模拟器（或Android手机）编程调试。

**七、实验结果与分析（含重要数据结果分析或核心代码流程分析）**

1.在Android Studio开发环境中创建Android项目

2.界面设计



界面设计xml代码

|  |
| --- |
| *<?***xml version="1.0" encoding="utf-8"***?>* <**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"**>  <**LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:paddingTop="80dp"**>  <**EditText  android:id="@+id/insert\_edittext"  android:layout\_width="0dp"  android:layout\_height="wrap\_content"  android:layout\_weight="5"  android:hint="请输入要插入的数据"**/>  <**Button  android:id="@+id/insert"  android:layout\_width="0dp"  android:layout\_height="wrap\_content"  android:layout\_weight="2"  android:text="执行插入"**/>  <**Button  android:id="@+id/insert\_clear"  android:layout\_width="0dp"  android:layout\_height="wrap\_content"  android:layout\_weight="2"  android:text="清除"**/>  </**LinearLayout**>   <**LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"**>  <**EditText  android:id="@+id/upadte\_before\_edittext"  android:layout\_width="0dp"  android:layout\_height="wrap\_content"  android:layout\_weight="2.5"  android:hint="请输入更新前的内容"**/>  <**EditText  android:id="@+id/update\_after\_edittext"  android:layout\_width="0dp"  android:layout\_height="wrap\_content"  android:layout\_weight="2.5"  android:hint="请输入更新后的内容"**/>  <**Button  android:id="@+id/update"  android:layout\_width="0dp"  android:layout\_height="wrap\_content"  android:layout\_weight="2"  android:text="执行更新"**/>  <**Button  android:id="@+id/update\_clear"  android:layout\_width="0dp"  android:layout\_height="wrap\_content"  android:layout\_weight="2"  android:text="清除"**/>  </**LinearLayout**>   <**LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"**>  <**EditText  android:id="@+id/delete\_edittext"  android:layout\_width="0dp"  android:layout\_height="wrap\_content"  android:layout\_weight="5"  android:hint="请输入要删除的数据"**/>  <**Button  android:id="@+id/delete"  android:layout\_width="0dp"  android:layout\_height="wrap\_content"  android:layout\_weight="2"  android:text="执行删除"**/>  <**Button  android:id="@+id/delete\_clear"  android:layout\_width="0dp"  android:layout\_height="wrap\_content"  android:layout\_weight="2"  android:text="清除"**/>  </**LinearLayout**>   <**LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:orientation="vertical"**>  <**Button  android:id="@+id/query"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="查询所有"**/>  <**Button  android:id="@+id/query\_clear"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="清除查询"**/>   </**LinearLayout**>   <**LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:orientation="vertical"  android:padding="10dp"  android:paddingTop="40dp"**>  <**TextView  android:id="@+id/show\_text"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:text="查询内容为空"**/>   </**LinearLayout**> </**LinearLayout**> |

3.MainActivity代码

|  |
| --- |
| **package** com.example.application2;  **import** androidx.appcompat.app.AppCompatActivity; **import** android.content.ContentValues; **import** android.content.Context; **import** android.database.Cursor; **import** android.database.sqlite.SQLiteDatabase; **import** android.database.sqlite.SQLiteOpenHelper; **import** android.os.Bundle; **import** android.view.View; **import** android.widget.Button; **import** android.widget.EditText; **import** android.widget.TextView;  *//创建数据库* **class** Database1 **extends** SQLiteOpenHelper {  **private** Context **context**;  **public** Database1(Context context, String name, SQLiteDatabase.CursorFactory factory, **int** version){  **super**(context,name,factory,version);  **this**.**context** =context;  }  @Override  **public void** onCreate(SQLiteDatabase db) {  String sql = **"create table user (name varchar(20) primary key)"**;  db.execSQL(sql);  }  @Override  **public void** onUpgrade(SQLiteDatabase db, **int** oldVersion, **int** newVersion) {   } }  *//主程序* **public class** MainActivity **extends** AppCompatActivity **implements** View.OnClickListener {   **private** Database1 **dbHelper**;  **private** EditText **delete\_text**;  **private** EditText **insert\_text**;  **private** EditText **update\_before\_text**;  **private** EditText **update\_after\_text**;  **private** EditText **show\_text**;  @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_main***);  Button button\_insert = (Button) findViewById(R.id.***insert***);  Button button\_insert\_clear = (Button) findViewById(R.id.***insert\_clear***);  Button button\_update = (Button) findViewById(R.id.***update***);  Button button\_update\_clear = (Button) findViewById(R.id.***update\_clear***);  Button button\_delete = (Button) findViewById(R.id.***delete***);  Button button\_delete\_clear = (Button) findViewById(R.id.***delete\_clear***);  Button button\_query = (Button) findViewById(R.id.***query***);  Button button\_query\_clear = (Button) findViewById(R.id.***query\_clear***);  button\_insert.setOnClickListener(**this**);  button\_insert\_clear.setOnClickListener(**this**);  button\_update.setOnClickListener(**this**);  button\_update\_clear.setOnClickListener(**this**);  button\_delete.setOnClickListener(**this**);  button\_delete\_clear.setOnClickListener(**this**);  button\_query.setOnClickListener(**this**);  button\_query\_clear.setOnClickListener(**this**);   **dbHelper** = **new** Database1(**this**, **"user.db"**, **null**, 1);  **insert\_text** = (EditText) findViewById(R.id.***insert\_edittext***);  **delete\_text** = (EditText) findViewById(R.id.***delete\_edittext***);  **update\_before\_text** = (EditText) findViewById(R.id.***upadte\_before\_edittext***);  **update\_after\_text** = (EditText) findViewById(R.id.***update\_after\_edittext***);  }    **public void** onClick(View v) {  SQLiteDatabase db = **dbHelper**.getWritableDatabase();  String update\_before\_data=**update\_before\_text**.getText().toString();  String update\_after\_data=**update\_after\_text**.getText().toString();  String insert\_data=**insert\_text**.getText().toString();  String delete\_data=**delete\_text**.getText().toString();  TextView textview=(TextView) findViewById(R.id.***show\_text***);  **switch** (v.getId()) {  *//插入数据* **case** R.id.***insert***:  ContentValues values = **new** ContentValues();  values.put(**"name"**, insert\_data);  db.insert(**"user"**, **null**, values);  **break**;  *//插入数据按钮后的清除按钮* **case** R.id.***insert\_clear***:  **insert\_text**.setText(**""**);  **break**;  *//删除数据按钮* **case** R.id.***delete***:  db.delete(**"user"**, **"name=?"**, **new** String[]{delete\_data});  **break**;  *//删除按钮后的清除按钮* **case** R.id.***delete\_clear***:  **delete\_text**.setText(**""**);  **break**;  *//更新数据按钮* **case** R.id.***update***:  ContentValues values2 = **new** ContentValues();  values2.put(**"name"**, update\_after\_data);  db.update(**"user"**, values2, **"name=?"**, **new** String[]{update\_before\_data});  **break**;  *//更新数据按钮后的清除按钮* **case** R.id.***update\_clear***:  **update\_before\_text**.setText(**""**);  **update\_after\_text**.setText(**""**);  **break**;  *//查询全部按钮* **case** R.id.***query***:  Cursor cursor = db.query(**"user"**, **new** String[]{**"name"**}, **null**, **null**, **null**, **null**, **null**);  String textview\_data = **""**;  **while** (cursor.moveToNext()) {  String name = cursor.getString(cursor.getColumnIndex(**"name"**));  textview\_data = textview\_data + **"\n"** + name;  }  textview.setText(textview\_data);  cursor.close();  **break**;  *//查询按钮后的清除查询按钮* **case** R.id.***query\_clear***:  textview.setText(**""**);  textview.setHint(**"查询内容为空"**);  **break**;  }  } } |

4.运行结果

插入数据12345并查询



插入清空，并更新为admin然后查询



更新清空，删除数据并查询，因为没有数据所以下方无显示



**八、总结及心得体会：**

通过本次实验了解了Android开发中最基本的数据库操作，包括数据的插入、查询、更新以及删除，增加了对SQLite的理解。在此次编程中重写了onClick函数从而减少了代码的臃肿量，使之更简洁。

**报告评分：**

**指导教师签字：**