# 实施方案 数据库操作SQLiteDatabase

***=======任务描述======***

**（1）编写Myhelper类，使程序运行时自动创建数据库db\_stu并创建用户表user(id,name,pwd)。**

**（2）编写注册事件，要求向user表中插入记录（用户名、密码由用户输入）。（可暂时不考虑新增用户系统内已存在的问题）**

**（3）编写登录事件，要求根据用户输入的用户名和密码到user表中查询是否有此用户。**



***=======关键技术======***

1. **SQLiteDatabase**
   * **数据库类**
   * **用于管理数据表(创建表)及数据记录(查询、添加、删除、修改记录)**
2. 数据库操作的执行步骤：
3. 创建数据库辅助类对象(SQLiteOpenHelper的子类)
4. 创建数据库类对象，建立连接(SQLiteDatabase类)
5. 执行SQL语句(execSQL、rawQuery方法)
6. 对获取的查询等操作结果进行加工
7. 回收资源，关闭数据库(close方法)
8. Cursor：游标

* 保存查询结果集，通过游标的移动依次访问记录
* 游标的移动：moveToFirst()、moToLast()、moToNext()、moveToPrevious()
* 游标的判断：isFirst()、isLast、isBeforeFirst、isAfterLast()
* 字段操作：
  + - getColumnCount() – 字段个数
    - getColumnIndex(字段名) -- 表中指定字段对应的索引序号
    - getString|Int|Double|Float(index) – 第index个字段的值

***=======项目实施======***

**步骤1、数据库创建连接应用（上个任务已实现）**

***MyHelper.java***

**package** com.jvtc.ad.ys0302;  
**import** android.content.Context;  
**import** android.database.sqlite.SQLiteDatabase;  
**import** android.database.sqlite.SQLiteOpenHelper;  
*/\*\*  
 \* 自定义数据库辅助类，创建数据库的同时创建表  
 \* Created by Administrator on 2017/3/22.  
 \*/***public class** MyHelper **extends** SQLiteOpenHelper {  
 **private static final** String ***CREATE\_TABLE***=**"create table user(\_id integer primary key autoincrement,name,pwd)"**;  
 *//定义类常量保存创建表的SQL语句* **public** MyHelper(Context context, String name, SQLiteDatabase.CursorFactory factory, **int** version) {  
 **super**(context, name, factory, version);  
 }  
 @Override  
 **public void** onCreate(SQLiteDatabase db) {  
 db.execSQL(***CREATE\_TABLE***); *//执行创建表的sql语句  
 //db.execSQL(String sql);---执行创建表，记录的添加、修改、删除操作  
 //Cursor c=db.rawQuery(String sql);---执行查询操作* }  
 @Override  
 **public void** onUpgrade(SQLiteDatabase db, **int** oldVersion, **int** newVersion) {  
 }  
}

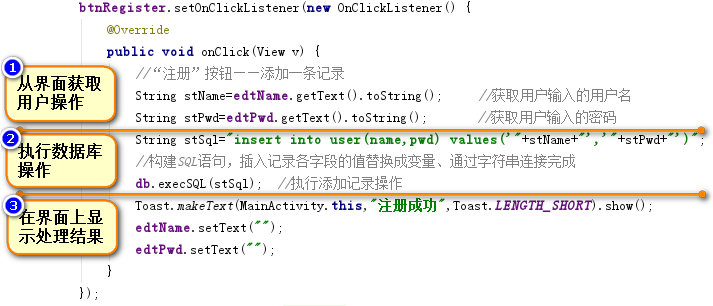
***MainActivity.java （1.0）***

**package** com.jvtc.ad.ys0302;  
**import** android.database.sqlite.SQLiteDatabase;  
**import** android.support.v7.app.AppCompatActivity;  
  
**public class** MainActivity **extends** AppCompatActivity {  
 **private** MyHelper **dbHelper**; *//数据库辅助类对象* **private** SQLiteDatabase **db**; *//数据库对象* @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 *//------------初始化数据库--------------* **dbHelper**=**new** MyHelper(**this**,**"db\_stu"**,**null**,1); *//声明数据库* **db**=**dbHelper**.getWritableDatabase(); *//获得数据库对象（执行创建、打开数据库方法）* }

}

**步骤2、数据操作——更新记录**

***关键技术解析1：数据操作基本流程***



***用户注册功能实现***

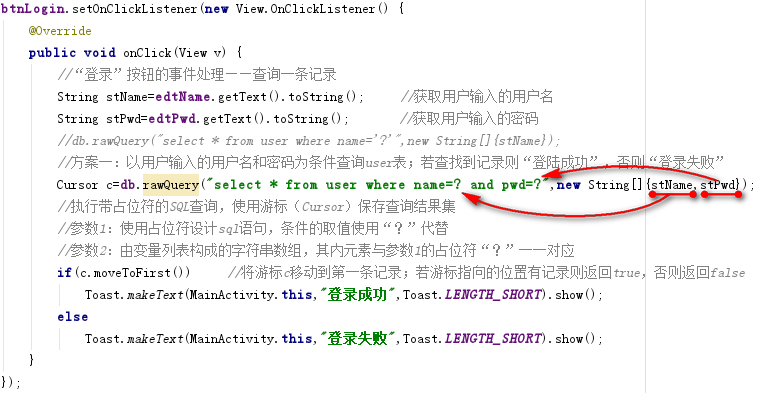
**btnRegister**.setOnClickListener(**new** OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 *//“注册”按钮——添加一条记录* String stName=**edtName**.getText().toString(); *//获取用户输入的用户名* String stPwd=**edtPwd**.getText().toString(); *//获取用户输入的密码* String stSql=**"insert into user(name,pwd) values('"**+stName+**"','"**+stPwd+**"')"**;  
 *//构建SQL语句，插入记录各字段的值替换成变量、通过字符串连接完成* **db**.execSQL(stSql); *//执行添加记录操作* Toast.*makeText*(MainActivity.**this**,**"注册成功"**,Toast.***LENGTH\_SHORT***).show();  
 **edtName**.setText(**""**);  
 **edtPwd**.setText(**""**);  
 }  
});

**步骤3、数据操作——查询记录**

***用户登录功能实现 【方法1】【方法2】***

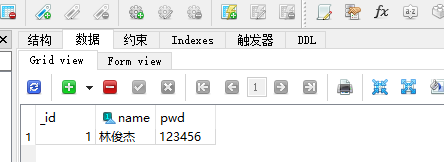
**btnLogin**.setOnClickListener(**new** OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 *//“登录”按钮——查询一条记录* String stName=**edtName**.getText().toString(); *//获取用户输入的用户名* String stPwd=**edtPwd**.getText().toString(); *//获取用户输入的密码  
 /\*  
 //方案一：依据用户输入的用户名和密码到用户表中查询记录，若查询到一条记录则“登录成功”、否则“登录失败”  
 Cursor c=db.rawQuery("select \* from user where name=? and pwd=?",new String[]{stName,stPwd});  
 //执行带占位符的SQL查询  
 //参数1：使用占位符设计sql语句，查询条件的具体取值处用“？”占位  
 //参数2：由变量列表构成的字符串数组，变量依次对应参数1中的“？”  
 if(c.moveToFirst()) //将游标移动到第一条记录，若有记录返回true、否则返回false  
 Toast.makeText(MainActivity.this,"登录成功",Toast.LENGTH\_SHORT).show();  
 else  
 Toast.makeText(MainActivity.this,"登录失败",Toast.LENGTH\_SHORT).show();  
 \*/  
 //方案二：依据用户输入的用户名到用户表中查询记录，用查询到的密码和用户输入的密码比较；相同则“登录成功”、否则“登录失败”* Cursor c=**db**.rawQuery(**"select \* from user where name=?"**,**new** String[]{stName});  
 String userPwd=**null**; *//从用户表中查询到的密码* **if**(c.moveToFirst()) *//令游标移到第1条记录，同时判断是否查询到记录，若有记录进一步获取密码* {  
 **int** n=c.getColumnIndex(**"pwd"**); *//获取用户表中“密码”字段的索引号* userPwd=c.getString(n); *//获取当前记录第n个字段的值* }  
 **if**(stPwd.equals(userPwd)) *//判断表中查询到的密码和用户输入的密码是否相等* Toast.*makeText*(MainActivity.**this**,**"登录成功"**,Toast.***LENGTH\_SHORT***).show();  
 **else** Toast.*makeText*(MainActivity.**this**,**"登录失败"**,Toast.***LENGTH\_SHORT***).show();  
 }  
});

***关键技术解析2：使用占位符的sql语句编写方法***

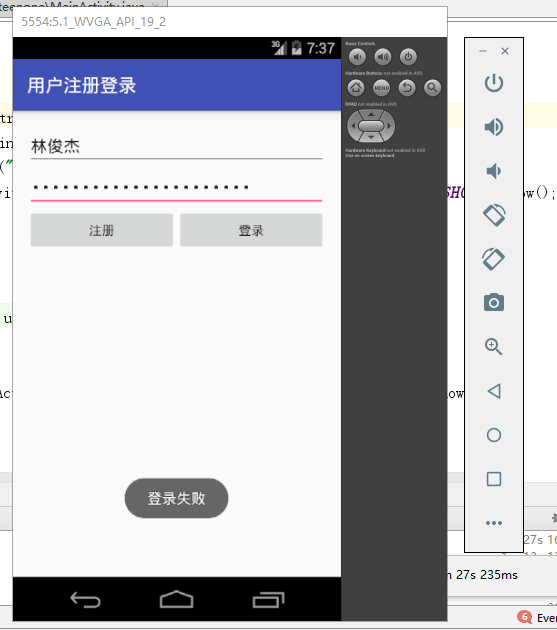


**运行截图：**









**代码：**

MyHelper.java

**package** com.jvtc.ad.task2;  
  
**import** android.content.Context;  
**import** android.database.sqlite.SQLiteDatabase;  
**import** android.database.sqlite.SQLiteOpenHelper;  
  
*/\*\*  
 \* Created by Administrator on 2020/6/10  
 \*/***public class** MyHelper **extends** SQLiteOpenHelper{  
 *// 表定义* **private final static** String ***CREATE\_TABLE*** = **"create table user(\_id integer primary key autoincrement,name,pwd)"**;  
 **public** MyHelper(Context context, String name, SQLiteDatabase.CursorFactory factory, **int** version){  
 **super**(context, name, factory, version);  
 }  
 @Override  
 **public void** onCreate(SQLiteDatabase db) {  
 db.execSQL(***CREATE\_TABLE***);  
 }  
  
 @Override  
 **public void** onUpgrade(SQLiteDatabase db, **int** oldVersion, **int** newVersion) {  
  
 }  
  
}

MainActivity.java

**package** com.jvtc.ad.task2;  
  
**import** android.database.Cursor;  
**import** android.database.sqlite.SQLiteDatabase;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**public class** MainActivity **extends** AppCompatActivity **implements** View.OnClickListener{  
 **private** MyHelper **dbHelper**;  
 **private** SQLiteDatabase **db**;  
 **private** EditText **txuser**, **txpwd**;  
 **private** Button **btreg**,**btlog**;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 **dbHelper** = **new** MyHelper(**this**, **"db\_stu"**, **null**, 1);  
 **db** = **dbHelper**.getWritableDatabase();*// 创建数据库 并且创建了表* findView();  
 }  
 **protected void** findView(){  
 **txuser** = (EditText)findViewById(R.id.***tx\_user***);  
 **txpwd** = (EditText)findViewById(R.id.***tx\_pwd***);  
 **btreg** = (Button)findViewById(R.id.***bt\_reg***);  
 **btlog** = (Button)findViewById(R.id.***bt\_log***);  
  
 **btreg**.setOnClickListener(**this**);  
 **btlog**.setOnClickListener(**this**);  
 }  
 @Override  
 **public void** onClick(View v) {  
 String stName = **txuser**.getText().toString();  
 String stPwd = **txpwd**.getText().toString();  
 **if**(stName.equals(**""**) || stPwd.equals(**""**)){  
 Toast.*makeText*(MainActivity.**this**, **"用户名和密码不能位空"**,Toast.***LENGTH\_SHORT***).show();  
 **return**;  
 }  
 **switch** (v.getId()){  
 **case** R.id.***bt\_reg***:  
 String stSql = **"insert into user(name, pwd) values('"**+stName+**"','"**+stPwd+**"')"**;  
 *//db执行sql语句* **db**.execSQL(stSql);  
 Toast.*makeText*(MainActivity.**this**, **"注册成功"**,Toast.***LENGTH\_SHORT***).show();  
 **txuser**.setText(**""**);  
 **txpwd**.setText(**""**);  
 **break**;  
 **case** R.id.***bt\_log***:  
 *// 根据输入的用户名与密码与数据库匹配，若有一条数据就登录成功，否则失败* Cursor c = **db**.rawQuery(**"select \* from user where name=? and pwd=?"**, **new** String[]{stName, stPwd});  
 **if**(c.moveToFirst()){  
 Toast.*makeText*(MainActivity.**this**, **"登录成功"**,Toast.***LENGTH\_SHORT***).show();  
 }**else**{  
 Toast.*makeText*(MainActivity.**this**, **"登录失败"**,Toast.***LENGTH\_SHORT***).show();  
 }  
 **break**;  
 }  
 }  
}

**activity\_main.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools" android:id="@+id/activity\_main"  
 android:layout\_width="match\_parent" android:layout\_height="match\_parent"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 tools:context="com.jvtc.ad.task2.MainActivity"  
 android:orientation="vertical"**>  
  
 <**EditText  
 android:id="@+id/tx\_user"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="请输入用户名"** />  
 <**EditText  
 android:id="@+id/tx\_pwd"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:inputType="numberPassword"  
 android:hint="请输入密码"** />  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:layout\_weight="2"**>  
 <**Button  
 android:id="@+id/bt\_reg"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:text="注册"  
 android:layout\_weight="1"**/>  
 <**Button  
 android:id="@+id/bt\_log"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:text="登录"  
 android:layout\_weight="1"**/>  
 </**LinearLayout**>  
</**LinearLayout**>