Android SQLite DB的封装

# DbOpenHelper:

**package** com.example.db\_demo01.DB;

**import** android.content.Context;

**import** android.database.sqlite.SQLiteDatabase;

**import** android.database.sqlite.SQLiteOpenHelper;

**public** **class** DbOpenHelper **extends** SQLiteOpenHelper {

**private** **static** String *name* = "mydb.db"; //数据库名

**private** **static** **int** *version* = 1; //版本号

**public** DbOpenHelper(Context context) {

**super**(context, *name*, **null**, *version*);

}

@Override

**public** **void** onCreate(SQLiteDatabase db) {

String sql = "create table person(id integer primary key autoincrement, name varchar(64), address varchar(64))";

db.execSQL(sql); //创建表

}

@Override

**public** **void** onUpgrade(SQLiteDatabase arg0, **int** arg1, **int** arg2){

//更新表结构时要同时更新版本号

}

}

# DB操作的封闭:

**package** com.example.db\_demo01.DB;

**import** java.util.ArrayList;

**import** java.util.HashMap;

**import** java.util.List;

**import** java.util.Map;

**import** android.content.Context;

**import** android.database.Cursor;

**import** android.database.sqlite.SQLiteDatabase;

**public** **class** DB {

**private** DbOpenHelper helper = **null**;

**public** DB(Context context){

helper = **new** DbOpenHelper(context);

}

**public** **boolean** ExecSQL(String sql){

**boolean** flag = **false**;

SQLiteDatabase database = **null**;

**try** {

database = helper.getWritableDatabase();

database.execSQL(sql);

flag = **true**;

} **catch** (Exception e) {

e.printStackTrace();

}**finally**{

**if**(database != **null**){

database.close();

}

}

**return** flag;

}

//插入

**public** **boolean** Insert(String sql){

**return** ExecSQL(sql);

}

//删除

**public** **boolean** Delete(String sql){

**return** ExecSQL(sql);

}

//更新

**public** **boolean** Update(String sql){

**return** ExecSQL(sql);

}

//查询

**public** List<Map<String, String>> Query(String sql){

SQLiteDatabase database = **null**;

List<Map<String, String>> list = **new** ArrayList<Map<String, String>>();

**try** {

database = helper.getWritableDatabase();

Cursor cursor = database.rawQuery(sql, **null**);

**int** colums = cursor.getColumnCount();

**while**(cursor.moveToNext()){

Map<String, String> map = **new** HashMap<String, String>();

**for**(**int** i=0; i<colums; i++){

String cols\_name = cursor.getColumnName(i);

String cols\_value = cursor.getString(cursor.getColumnIndex(cols\_name));

**if**(cols\_value == **null**){

cols\_value = "";

}

map.put(cols\_name, cols\_value);

}

list.add(map);

}

} **catch** (Exception e) {

e.printStackTrace();

}**finally**{

**if**(database != **null**){

database.close();

}

}

**return** list;

}

}

# 单元测试类:

**package** com.example.db\_demo01.test;

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.Map;

**import** android.test.AndroidTestCase;

**import** com.example.db\_demo01.DB.DB;

**import** com.example.db\_demo01.DB.DbOpenHelper;

**public** **class** MyTest **extends** AndroidTestCase {

**public** MyTest(){}

**public** **void** createDb(){

DbOpenHelper helper = **new** DbOpenHelper(getContext());

helper.getWritableDatabase();

}

**public** **void** insert(){

DB db = **new** DB(getContext());

String sql = "insert into person(name, address) values('邓', 'zhbit')";

db.Insert(sql);

}

**public** **void** delete(){

DB db = **new** DB(getContext());

String sql = "delete from person where name = '邓'";

System.*out*.println(db.Delete(sql));

}

**public** **void** update(){

DB db = **new** DB(getContext());

String sql = "update person set name = '邓2' where name = '邓'";

System.*out*.println(db.Delete(sql));

}

**public** **void** query(){

DB db = **new** DB(getContext());

String sql = "select \* from person";

List<Map<String, String>> list = **new** ArrayList<Map<String, String>>();

list = db.Query(sql);

System.*out*.println(list.toString());

}

}