

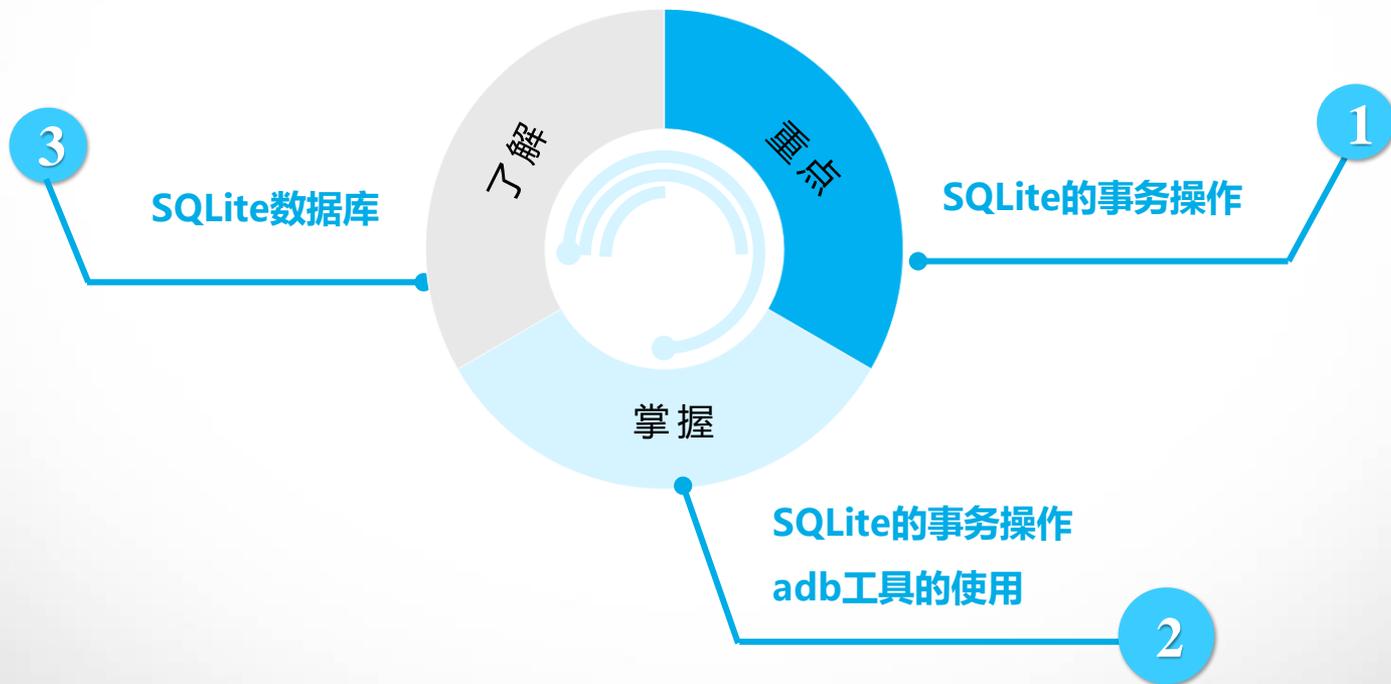


电子信息工程学院

移动终端开发技术

# SQLite数据库存储

讲 师：陈媛媛



 返回目录

## 4.3 SQLite数据库存储

- 4.3.1 ..... ● SQLite数据库简介
- 4.3.2 ..... ● 创建数据库
- 4.3.3 ..... ● adb工具的使用
- 4.3.4 ..... ● 升级数据库
- 4.3.5 ..... ● 对数据的增删改查操作

# SQLite数据库简介



SQLite是一个轻量级**数据库**，第一个版本诞生于2000年5月。它最初是为嵌入式设计的，占用资源非常低，在内存中只需要占用几百KB的存储空间。

SQLite没有服务器进程，它通过文件保存数据，该文件是跨平台的。并且在保存数据时，支持 NULL、INTEGER、REAL（浮点数字）、TEXT(字符串文本)和BLOB(二进制对象)五种数据类型。

SQLite操作API

# SQLite数据库简介

Android SDK提供了一系列对数据库进行操作的类和接口。常用的数据库操作类有：

## SQLiteOpenHelper类

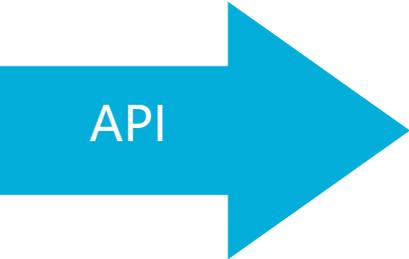
- 该类是一个抽象类，用于创建数据库和数据库版本更新。

## SQLiteDatabase类

- 该类是一个数据库访问类，封装了一系列数据库操作的API，可以对数据进行增删改查操作。

## Cursor接口

- 是一个游标接口，在数据库操作中作为返回值，相当于结果集ResultSet。



API

# 创建数据库

(1) 创建类MyDatabaseHelper.java继承SQLiteOpenHelper

(2) 定义创建表的字符串

```
public static final String create_book="create table Book (" + "id integer  
primary key autoincrement," + "author varchar(20)," + "price double,"  
+ "name varchar(20))";
```

(3) 重写onCreate方法

```
public void onCreate(SQLiteDatabase sqLiteDatabase) {  
    sqLiteDatabase.execSQL(create_book);  
}
```

(4) MainActivity.java

```
private MyDatabaseHelper dbHelper;  
dbHelper=new MyDatabaseHelper(this,"BookStore.db",null,1)  
dbHelper.getWritableDatabase();//创建数据库
```

如何查看刚刚创建的数据库和表呢?

# adb工具的使用

- 配置环境变量：将platform-tools目录配置到环境变量中。
- 运行cmd输入：adb shell（使用adb工具）
- #表示超级管理员 \$表示普通管理员，使用su切换
- # cd /data/data/com.example.cyy.databasetest/databases
- Ls（查看数据库）
- sqlite3 BookStore.db（使用sqlite3操作应用程序下的数据库）
- .table（查看表）
- .schema（查看表结构）



adb工具

# 升级数据库

(1) MyDatabaseHelper类中的onUpgrade () 方法

```
public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {  
    sqLiteDatabase.execSQL("drop table if exists Book");  
    sqLiteDatabase.execSQL("drop table if exists Category");  
    onCreate(sqLiteDatabase);  
}
```

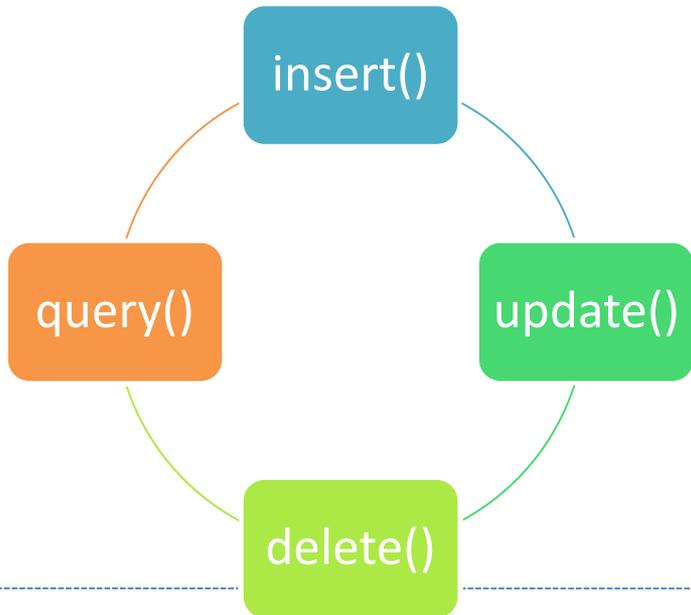
(2) MainActivity.java

```
dbHelper=new MyDatabaseHelper(this,"Book1.db",null,2);  
//创建数据库, 第四个参数大于1就能让onUpgrade () 方法执行
```

# 数据的CRUB操作

Android SDK提供的一系列操作**SQLite数据库**的**API**。

SQLiteDatabase 就是其中的一个，它提供了一些列用于操作数据库的方法。



# 数据的Create操作

-第一步: `getWritableDatabase()`方法返回`SQLiteDatabase`对象, 用于对数据进行增删改查

- `SQLiteDatabase db=dbHelper.getWritableDatabase();`

-第二步: 创建`ContentValues`对象

-`ContentValues values=new ContentValues();`

-第三步: 开始组装一条数据

-`values.put("字段名",值);`

-第四步: 插入记录

-`db.insert("Book",null,values);`

# 数据的Update操作

-第一步: `getWritableDatabase()`方法返回`SQLiteDatabase`对象, 用于对数据进行增删改查

- `SQLiteDatabase db=dbHelper.getWritableDatabase();`

-第二步: 创建`ContentValues`对象

-`ContentValues values=new ContentValues();`

-第三步: 开始组装一条数据

-`values.put("字段名",值);`

-第四步: 调用`update ()`方法进行更新

-`db.update("Book",values,"name=?",new String[]{"Android"});`

有四个参数, 第一个参数为表名, 第二个参数为`ContentValues`对象, 第三个、四个参数用于约束更新某一行或某几行中的数据, 不指定就更新所有行。

# 数据的Delete操作

– **第一步**: `getWritableDatabase()`方法返回`SQLiteDatabase`对象，用于对数据进行增删改查

– `SQLiteDatabase db=dbHelper.getWritableDatabase();`

– **第二步**: 调用`delete ()`方法删除数据

–`db.delete("Book","price>?",new String[]{"100"});`

`delete()`有三个参数，第一个参数是表名，第二个、第三个参数约束删除某一行或某几行的数据，不指定默认删除所有行。

# 数据的Retrieve操作

– 第一步: `getWritableDatabase()`方法返回`SQLiteDatabase`对象, 用于对数据进行增删改查

– `SQLiteDatabase db=dbHelper.getWritableDatabase();`

– 第二步: 调用`query ()`方法查询数据, 将结果赋给`Cursor`对象。

– `Cursor cursor=db.query("Book",null,null,null,null,null);`

– 第三步: 遍历`cursor`, 读取每个字段。

– 常用方法: `cursor.moveToFirst ()` 移到第一条记录; `getColumnIndex("字段名")` 得到某一系列在表中的位置索引; `cursor.getString ()`; `cursor.getInt ()`; `cursor.getDouble ()`; 获取数据。

– 第四步: 关闭`cursor`

– `cursor.close();`

# 数据的Retrieve操作

query()方法参数	对应SQL部分	描 述
table	from table_name	指定查询的表名
columns	select column1, column2	指定查询的列名
selection	where column = value	指定 <u>where</u> 的约束条件
selectionArgs	-	为 <u>where</u> 中的占位符提供具体的值
groupBy	group by column	指定需要 <u>group by</u> 的列
having	having column = value	对group by后的结果进一步约束
orderBy	order by column1, column2	指定查询结果的排序方式



1

SQLite数据库

---

2

adb测试工具的使用

---

3

数据的增删改查操作

---