

## 40 | SQLite: 为什么微信用SQLite存储聊天记录?

2019-09-04 陈旻



我在上一篇文章中讲了WebSQL，当我们在Chrome、Safari和Firefox等浏览器客户端中使用WebSQL时，会直接操作SQLite。实际上SQLite本身是一个嵌入式的开源数据库引擎，大小只有3M左右，可以将整个SQLite嵌入到应用中，而不用采用传统的客户端 / 服务器（Client/Server）的架构。这样做的好处就是非常轻便，在许多智能设备和应用中都可以使用SQLite，比如微信就采用了SQLite作为本地聊天记录的存储。

今天我们就来深入了解一下SQLite，今天的内容主要包括以下几方面：

1. SQLite是什么？它有哪些优点和不足？
2. 如何在Python中使用SQLite？
3. 如何编写SQL，通过SQLite查找微信的聊天记录？

### SQLite是什么

SQLite是在2000年发布的，到目前为止已经有19年了。一直采用C语言编写，采用C语言而非C++面向对象的方式，可以提升代码底层的执行效率。但SQLite也有一些优势与不足。

它的优势在于非常轻量级，存储数据非常高效，查询和操作数据简单方便。此外SQLite不需要安装和配置，有很好的迁移性，能够嵌入到很多应用程序中，与托管在服务器上的RDBMS相比，约束少易操作，可以有效减少服务器的压力。

不足在于SQLite常用于小到中型的数据存储，不适用高并发的情况。比如在微信本地可以使用

**SQLite**，即使是几百M的数据文件，使用**SQLite**也可以很方便地查找数据和管理，但是微信本身的服务器就不能使用**SQLite**了，因为**SQLite**同一时间只允许一个写操作，吞吐量非常有限。

作为简化版的数据库，**SQLite**没有用户管理功能，在语法上也有一些自己的“方言”。比如在SQL中的**SELECT**语句，**SQLite**可以使用一个特殊的操作符来拼接两个列。在**MySQL**中会使用函数**concat**，而在**SQLite**、**PostgreSQL**、**Oracle**和**Db2**中使用**||**号，比如：**SELECT MesLocalID || Message FROM "Chat\_1234"**。

这个语句代表的是从**Chat\_1234**数据表中查询**MesLocalID**和**Message**字段并且将他们拼接起来。

但是在**SQLite**中不支持**RIGHT JOIN**，因此你需要将右外连接转换为左外连接，也就是**LEFT JOIN**，写成下面这样：

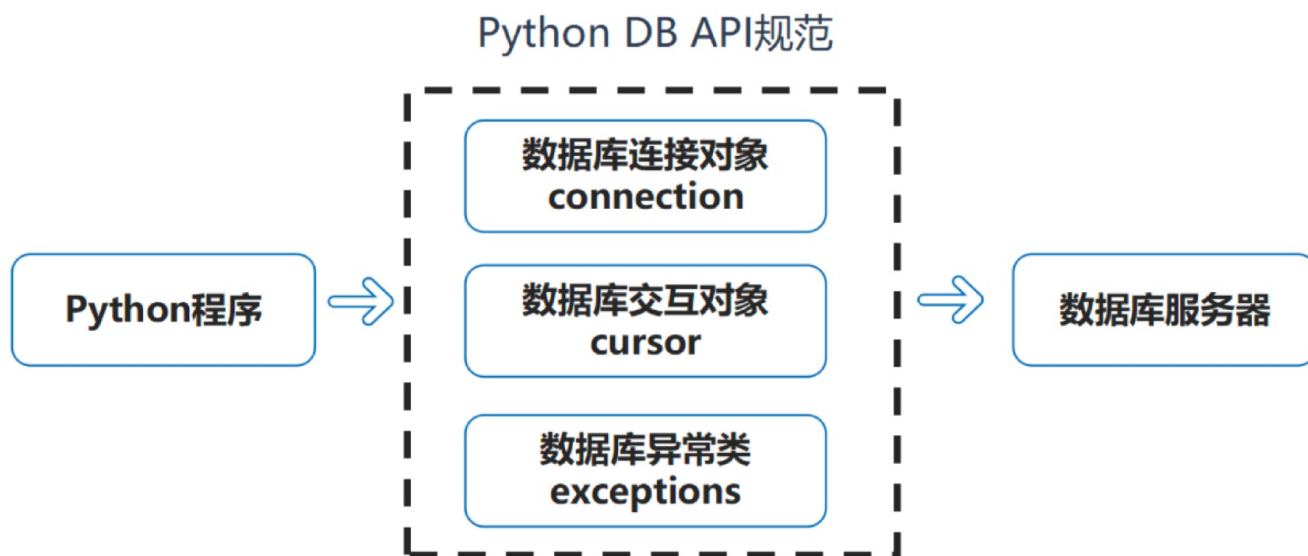
```
SELECT * FROM team LEFT JOIN player ON player.team_id = team.team_id
```

除此以外**SQLite**仅支持只读视图，也就是说，我们只能创建和读取视图，不能对它们的内容进行修改。

总的来说支持SQL标准的RDBMS语法都相似，只是不同的DBMS会有一些属于自己的“方言”，我们使用不同的DBMS的时候，需要注意。

## 在Python中使用SQLite

我之前介绍过如何在Python中使用MySQL，其中会使用到DB API规范（如下图所示）。基于DB API规范，我们可以对数据库进行连接、交互以及异常的处理。



在Python中使用**SQLite**也会使用到DB API规范，与使用**MySQL**的交互方式一样，也会用到

`connection`、`cursor`和`exceptions`。在Python中集成了SQLite3，直接加载相应的工具包就可以直接使用。下面我们就来看下如何在Python中使用SQLite。

在使用之前我们需要进行引用SQLite，使用：

```
import sqlite3
```

然后我们可以使用SQLite3创建数据库连接：

```
conn = sqlite3.connect("wucai.db")
```

这里我们连接的是wucai.db这个文件，如果没有这个文件存储，上面的调用会自动在相应的工程路径里进行创建，然后我们可以使用conn操作连接，通过会话连接conn来创建游标：

```
cur = conn.cursor()
```

通过这一步，我们得到了游标cur，然后可以使用execute()方法来执行各种DML，比如插入，删除，更新等，当然我们也可以进行SQL查询，用的同样是execute()方法。

比如我们想要创建heros数据表，以及相应的字段id、name、hp\_max、mp\_max、role\_main，可以写成下面这样：

```
cur.execute("CREATE TABLE IF NOT EXISTS heros (id int primary key, name text, hp_max real, mp_max real, ro
```

在创建之后，我们可以使用execute()方法来添加一条数据：

```
cur.execute("insert into heros values(?, ?, ?, ?, ?), (10000, '夏侯惇', 7350, 1746, '坦克'))
```

需要注意的是，一条一条插入数据太麻烦，我们也可以批量插入，这里会使用到executemany方法，这时我们传入的参数就是一个元组，比如：

```
cur.executemany("insert into heros values(?, ?, ?, ?, ?)",
                ((10000, '夏侯惇', 7350, 1746, '坦克'),
                 (10001, '钟无艳', 7000, 1760, '战士'),
                 (10002, '张飞', 8341, 100, '坦克'),
                 (10003, '牛魔', 8476, 1926, '坦克'),
                 (10004, '吕布', 7344, 0, '战士')))
```

如果我们想要对**heros**数据表进行查询，同样使用**execute**执行**SQL**语句：

```
cur.execute("SELECT id, name, hp_max, mp_max, role_main FROM heros")
```

这时**cur**会指向查询结果集的第一个位置，如果我们想要获取数据有以下几种方法：

1. **cur.fetchone()**方法，获取一条记录；
2. **cur.fetchmany(n)** 方法，获取**n**条记录；
3. **cur.fetchall()**方法，获取全部数据行。

比如我想获取全部的结果集，可以写成这样：

```
result = cur.fetchall()
```

如果我们对事务操作完了，可以提交事务，使用**conn.commit()**即可。

同样，如果游标和数据库的连接都操作完了，可以对它们进行关闭：

```
cur.close()
conn.close()
```

上面这个过程的完整代码如下：

```

import sqlite3

# 创建数据库连接
conn = sqlite3.connect("wucai.db")

# 获取游标
cur = conn.cursor()

# 创建数据表
cur.execute("CREATE TABLE IF NOT EXISTS heros (id int primary key, name text, hp_max real, mp_max real, role_main text)")

# 插入英雄数据
cur.executemany("insert into heros values(?, ?, ?, ?, ?)",
                ((10000, '夏侯惇', 7350, 1746, '坦克'),
                 (10001, '钟无艳', 7000, 1760, '战士'),
                 (10002, '张飞', 8341, 100, '坦克'),
                 (10003, '牛魔', 8476, 1926, '坦克'),
                 (10004, '吕布', 7344, 0, '战士')))

cur.execute("SELECT id, name, hp_max, mp_max, role_main FROM heros")
result = cur.fetchall()
print(result)

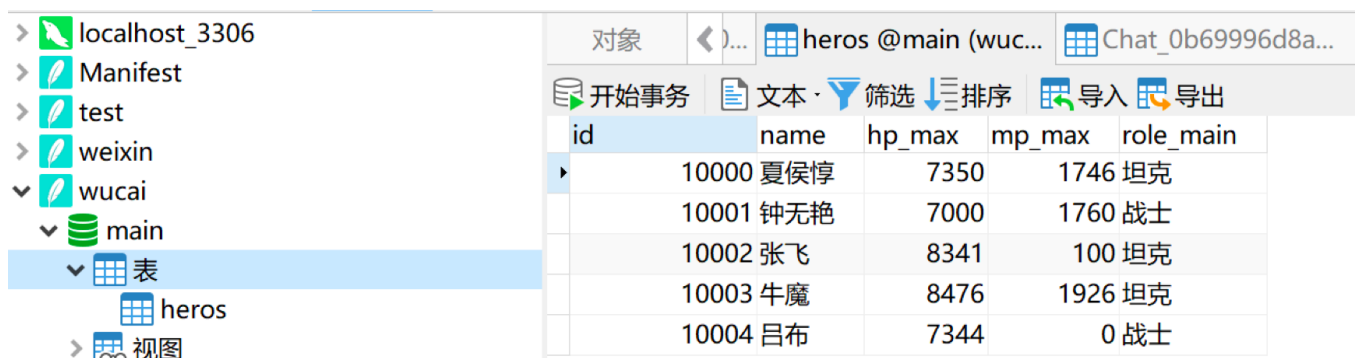
# 提交事务
conn.commit()

# 关闭游标
cur.close()

# 关闭数据库连接
conn.close()

```

除了使用Python操作SQLite之外，在整个操作过程中，我们同样可以使用navicat数据库可视化工具来查看和管理SQLite。



id	name	hp_max	mp_max	role_main
10000	夏侯惇	7350	1746	坦克
10001	钟无艳	7000	1760	战士
10002	张飞	8341	100	坦克
10003	牛魔	8476	1926	坦克
10004	吕布	7344	0	战士

## 通过SQLite查询微信的聊天记录

刚才我们提到很多应用都会集成SQLite作为客户端本地的数据库，这样就可以避免通过数据库服

务器进行交互，减少服务器的压力。

如果你是iPhone手机，不妨跟着我执行以下的步骤，来查找下微信中的SQLite文件的位置吧。

第一步，使用iTunes备份iPhone；第二步，在电脑中查找备份文件。

当我们备份好数据之后，需要在本地找到备份的文件，如果是windows可以在C:\Users\XXXX\AppData\Roaming\Apple Computer\MobileSync\Backup 这个路径中找到备份文件夹。

第三步，查找Manifest.db。

在备份文件夹中会存在Manifest.db文件，这个文件定义了苹果系统中各种备份所在的文件位置。

第四步，查找MM.sqlite。

Manifest.db本身是SQLite数据文件，通过SQLite我们能看到文件中包含了Files数据表，这张表中有fileID、domain和relativePath等字段。

微信的聊天记录文件为MM.sqlite，我们可以直接通过SQL语句来查询下它的位置（也就是fileID）。

```
SELECT * FROM Files WHERE relativePath LIKE '%MM.sqlite'
```

fileID	domain	relativePath
f71743874d7b858a01e3ddb933ce13a9a01f79aa	AppDomain-com.tencent.xin	Documents/01d19db82ea341ba0241808efe19e718/DB/MM.sqlite
be1887d89a9639bab982189f5b825eb4d2f3f310	AppDomain-com.tencent.xin	Documents/4a855dcd6284a87f5c4eaa5ef040cd46/DB/MM.sqlite

你能看到在我的微信备份中有2个MM.sqlite文件，这些都是微信的聊天记录。

第五步，分析找到的MM.sqlite。

这里我们需要在备份文件夹中查找相关的fileID，比如f71743874d7b858a01e3ddb933ce13a9a01f79aa。

找到这个文件后，我们可以复制一份，取名为weixin.db，这样就可以使用navicat对这个数据库进行可视化管理，如下图所示：



MesLocalID	Message	MesSvrID	Status	TableVer	Type
1	我通过了你的朋友验证请求，现在我们可以开始聊天了	989350943152704329	4	1	1
2	你好陈老师	213712047225364655	4	1	1
3	reverse好啊	355415419456701698	3	1	1
4	看到你写的笔记很感动~我好久没回复大家留言了，先准备回复一波	241992913801286708	3	1	1
5	你平时在南京么？	237476177047084507	3	1	1
6	是的	576672744235548064	4	1	1
7	好的 您先准备回复一波 我希望能够通过适当的当时 让同学们能够调整好学习的心态，用适当的方法来	15943969126879922	4	1	1
8	[强]感谢reverse 确实心态+方法很重要，感谢你的分享	763322309433172045	3	1	1
9	陈老师，你客气了，我是希望让同学的学习氛围能够向着好的方向发展，所以我希望群内有充满正能量的	184482674479746347	4	1	1
10	对的 同意的 正能量的话学习推动力也强 好的氛围确实是关键	309571477038112581	3	1	1
11	对于我来说，因为我大学不是本专业，学的土木工程，后来自学编程，去了北京两年，如果我的心态不好，	560529514103091063	4	1	1
12	明白 很多学编程的当时都不是计算机专业的，这个专业确实更靠练习和实战，还有一些悟性。	220209324508117222	3	1	1
13	心态也很重要 reverse的经验应该对于大部分人来说都很有价值	506385512683013537	3	1	1
14	你高抬了 只是自己的心得	332518740727478226	4	1	1
15	[玫瑰]	462881968082859125	2	1	1
16	[握手]	195213025034573926	4	1	1

微信会把你与每一个人的聊天记录都保存成一张数据表，在数据表中会有**MesLocalID**、**Message**、**Status**等相应的字段，它们分别代表在当前对话中的ID、聊天内容和聊天内容的状态）。

如果聊天对象很多的话，数据表也会有很多，如果想知道都有哪些聊天对象的数据表，可以使用：

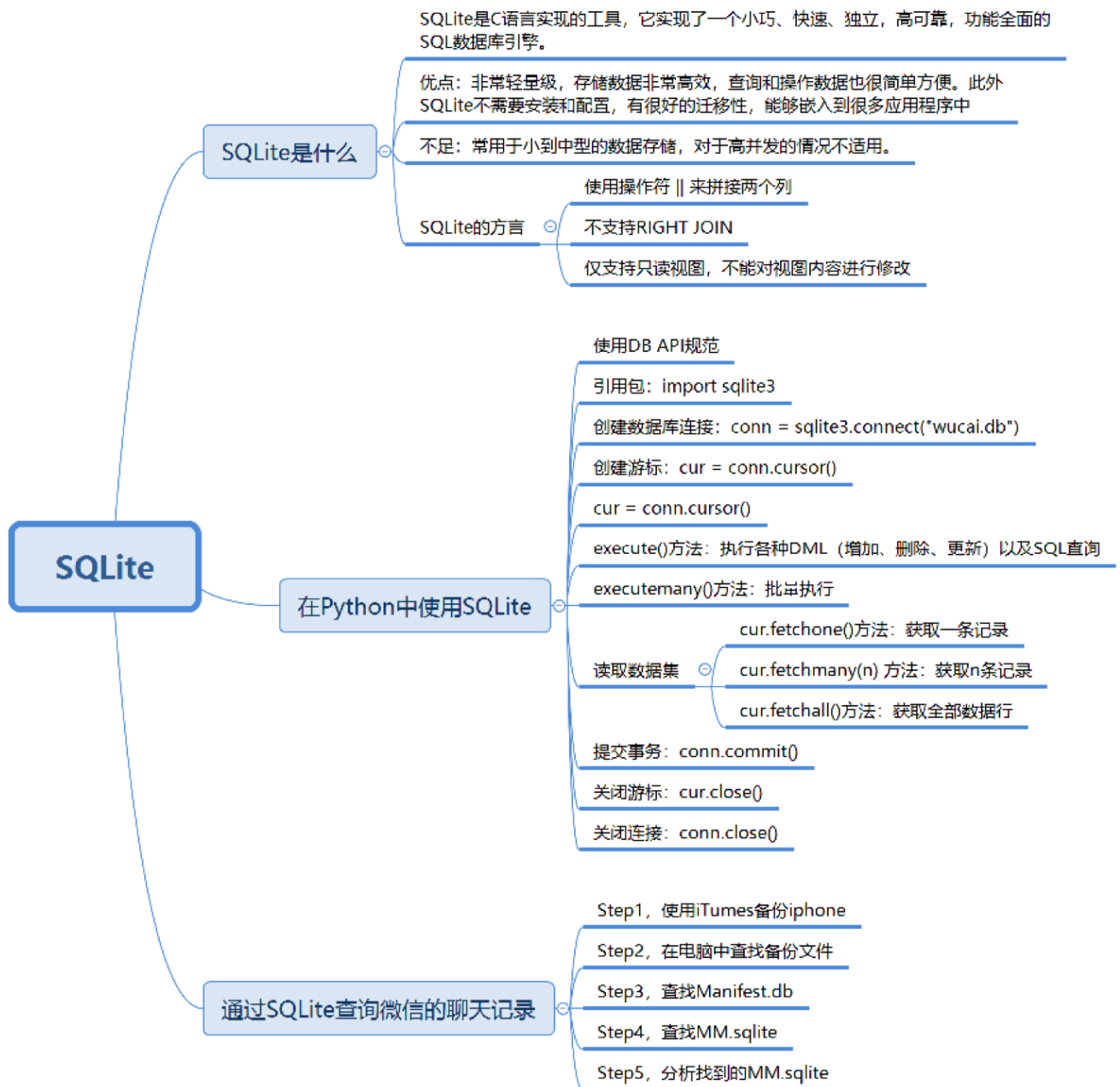
```
SELECT name FROM sqlite_master WHERE type = 'table' AND name LIKE 'Chat\_%' escape '\'
```

这里需要说明的是**sqlite\_master**是**SQLite**的系统表，数据表是只读的，里面保存了数据库中的数据表的名称。聊天记录的数据表都是以**Chat\_**开头的，因为（**\_**）属于特殊字符，在**LIKE**语句中会将（**\_**）作为通配符。所以如果我们想要对开头为**Chat\_**的文件名进行匹配，就需要用**escape**对这个特殊字符做转义。

## 总结

我今天讲了有关**SQLite**的内容。在使用**SQLite**的时候，需要注意**SQLite**有自己的方言，比如在进行表连接查询的时候不支持**RIGHT JOIN**，需要将其转换成**LEFT JOIN**等。同时，我们在使用**execute()**方法的时候，尽量采用带有参数的**SQL**语句，以免被**SQL**注入攻击。

学习完今天的内容后，不如试试用**SQL**查询来查找本地的聊天记录吧。



最后留一道思考题吧。请你使用SQL查询对微信聊天记录中和“作业”相关的记录进行查找。不论是iPhone，还是Android手机都可以找到相应的SQLite文件，你可以使用Python对SQLite进行操作，并输出结果。

欢迎你在评论区写下你的答案，也欢迎把这篇文章分享给你的朋友或者同事，一起交流一下。



# SQL 必知必会

从入门到数据实战

陈旻

清华大学计算机博士



新版升级：点击「👤请朋友读」，20位好友免费读，邀请订阅更有**现金**奖励。

## 精选留言



我

年底可以导出聊天记录做个词云

2019-09-04

作者回复

Good Idea

2019-09-09

👍 6



学习

Navicat如何导入那个wenxin.db呢，有点不太明白，可以说个步骤吗？

2019-09-04

👍 2



DemonLee

👋，微信聊天记录，涨姿势了

2019-09-04

作者回复

👋 手机QQ的本地聊天记录也用的SQLite

2019-09-09

👍 2



Hanqiu\_Tan

Use ".open FILENAME" to reopen on a persistent database.

sqlite> .open Manifest.db

👍 1

```
sqlite> .tables
```

```
Error: file is not a database
```

```
sqlite>
```

这边试了打不开文件是怎么回事呢？

2019-09-05



Tesla

这个聊天记录文件应该是不可编辑和替换的吧？

2019-09-22

👍 0



许童童

学习了，老师。

2019-09-05

👍 0