



北京大学
PEKING UNIVERSITY

信息科学技术学院

实用Python程序设计

郭 炜

微信公众号



微博: <http://weibo.com/guoweiofpku>

学会程序和算法，走遍天下都不怕！



北京大学
PEKING UNIVERSITY

信息科学技术学院 郭炜

Python数据库编程



北京大学
PEKING UNIVERSITY

信息科学技术学院

数据库和SQL



黄山

数据库的概念

- 数据库可以用来存放大量数据，并且提供了方便的快速检索手段
- 便于快速找出符合某种条件的数据。比如：工作年限超过三年，工资超过10000元的北京籍员工
- 一个数据库可以是一个文件，比如
c:/tmp/students. db

数据库中的表

- 一个数据库文件里可以有多张表。比如 `students.db` 里包含"学生信息表"和"课程信息表"
- 表由记录构成， 比如“学生信息表”里的每个记录，代表一个学生的信息。
- 记录由字段构成，描述一个事物的多个属性。比如学生记录，可以由 `name` , `id`, `age`, `gender`, `gpa` 等字段构成

字段

- 字段是有"类型"的。比如

字段名

name

gpa

age

profile

photo

birthday

register time

.....

数据类型

text

real

integer

text

blob

date

datetime

字符串

小数

整数

二进制数据（如图片）

日期（本质上就是text）

日期+时间（本质上就是text）

SQL数据库查询语句

参考: http://www.w3school.com.cn/sql/sql_syntax.asp

SQL命令: 用于进行数据库操作的标准语句

CREATE TABLE

创建表

INSERT INTO

在表中插入记录

UPDATE

在表中更新记录

SELECT

在表中进行查询

DELETE

从表中删除记录

不区分大小写

CREATE

```
CREATE TABLE if not exists students (id integer primary key,  
    name text, gpa real, birthday date, age integer, picture blob)
```

创建了一张名为**students**的表，有以下字段：

字段名

数据类型

id	integer
name	text
gpa	real
birthday	date
age	integer
picture	blob

primary key表示不可重复

字符串

小数

日期(本质上就是text)

整数

二进制数据 (如图片)

INSERT

```
INSERT INTO students VALUES (1000, '张三', 3.81, '2000-09-12',  
18, null)
```

在表 students 中插入一个记录，该记录暂无照片 (null)

VALUES (每个字段的值)



北京大学
PEKING UNIVERSITY

信息科学技术学院

创建数据库



长城入海处：老龙头

创建数据库并写入数据

```
import sqlite3
```

```
db = sqlite3.connect("c:/tmp/test2.db") #连接数据库, 若不存在则自动创建
```

```
#文件夹 c:/tmp 必须事先存在, connect不会创建文件夹
```

```
cur = db.cursor() #获取光标, 要操作数据库一般要通过光标进行
```

```
sql = '''CREATE TABLE if not exists students (id integer primary key,  
name text, gpa real, birthday date, age integer, picture blob)''' #如果  
表 students 不存在就创建它
```

```
cur.execute(sql) #执行SQL命令
```

```
cur.execute("insert into students values(1600, '张三', 3.81, '2000-  
09-12', 18,null)") #插入一个记录
```

```
mylist = [(1700, '李四', "3.25", '2001-12-01', 17, None),  
          (1800, '王五', "3.35", '1999-01-01', 19, None)]
```

```
for s in mylist: #依次插入mylist中的每个记录
```

```
    cur.execute('INSERT INTO students VALUES(?, ?, ?, ?, ?, ?)',  
                (s[0], s[1], s[2], s[3], s[4], s[5])) # ?对应于后面某项
```

```
db.commit() #真正写入, 写入操作都需要这个
```

```
cur.close() #关闭光标
```

```
db.close() #关闭数据库
```




北京大学
PEKING UNIVERSITY

信息科学技术学院

数据库的查询和修改



云岗石窟

SELECT

```
SELECT * FROM students
```

检索students表中全部记录

```
SELECT * FROM students ORDER BY age
```

检索students表中全部记录，并按年龄排序

```
SELECT name, age FROM students
```

检索students表中全部记录，但每个记录只取name和age字段

```
SELECT * FROM students WHERE name = '张三'
```

检索students表中全部name字段为张三的记录

WHERE 表示检索条件

```
SELECT * FROM students WHERE name = '张三' AND age > 20 ORDER BY age DESC
```

检索students表中全部名为张三且年龄大于20的人，结果按年龄降序排列

检索数据库

```
import sqlite3
db = sqlite3.connect("c:/tmp/test2.db")
cur = db.cursor()
sql = 'select * from students' #检索全部记录
cur.execute(sql)
x = cur.fetchone() #fetchone 取满足条件的第一条记录
print(x)           #=>(1600, '张三', 3.81, '2000-09-12', 18, None)
print(x[1])        #=>张三
for x in cur.fetchall(): #fetchall取得所有满足条件的记录
    print(x[:-2])        #age和picture字段不打出
cur.execute("SELECT * FROM students WHERE name='Jack'")
x = cur.fetchone()
if x == None:
    print("can't find Jack")
cur.close()
db.close()
```

```
(1600, '张三', 3.81, '2000-09-12', 18, None)
张三
(1700, '李四', 3.25, '2001-12-01')
(1800, '王五', 3.35, '1999-01-01')
can't find Jack
```

检索数据库

```
import sqlite3
db = sqlite3.connect("c:/tmp/test2.db")
cur = db.cursor()
sql = 'select name, gpa, age from students where gpa > 3.3
order by age desc'
#查找gpa > 3.3的记录，题取其中三个字段，按年龄降序排列
cur.execute(sql)
x = cur.fetchall()
if x != []:
    print("total: ", len(x)) #=>2
    for r in x:
        print(r)
cur.close()
db.close()
```

```
total: 2
('王五', 3.35, 19)
('张三', 3.81, 18)
```

UPDATE

```
UPDATE students SET gpa = 3.9
```

将所有记录的gpa设置成3.9

```
UPDATE students SET gpa = 3.9, age = 18 WHERE name = '李四'
```

修改 李四 的gpa和年龄

UPDATE

```
import sqlite3
db = sqlite3.connect("c:/tmp/test2.db")
cur = db.cursor()
sql = 'UPDATE students SET gpa = ?, age = ? WHERE name = ?'
cur.execute(sql, (4.0, 20, '李四')) #元组三个元素分别对应三个 ?
#修改 李四 的gpa和年龄 。若李四不存在，则无效果
db.commit() #写入操作必须
cur.close()
db.close()
```

DELETE

```
DELETE FROM students WHERE age < 18
```

删除年龄小于18的记录

```
DELETE FROM students
```

删除全部记录

别忘了最后 `commit`

DROP TABLE

```
DROP TABLE IF EXISTS students
```

删除students 表

别忘了最后 commit

DROP TABLE

```
import sqlite3
db = sqlite3.connect("c:/tmp/test2.db")
cur = db.cursor()
cur.execute("DROP TABLE IF EXISTS students")
db.commit()
try:
    cur.execute("select * from students")
    x = cur.fetchall()
    for r in x:
        print(r[:-1])
except:
    print("no table")    #=> no table
cur.close()
db.close()
```

列出数据库中所有的表和表的结构

```
import sqlite3
db = sqlite3.connect("c:/tmp/test3.db")
cur = db.cursor()
sql = 'CREATE TABLE if not exists table2 (id integer, name text)'
cur.execute(sql) #执行SQL命令
sql = 'CREATE TABLE if not exists table1 (id integer, schook text)'
cur.execute(sql)
db.commit()
cur.execute('select name from SQLITE_MASTER where type="table" order by NAME')
x = cur.fetchall()
if x != []:
    print(x)
cur.execute("PRAGMA TABLE_INFO (table1)")
print (cur.fetchall())
cur.close()
db.close()
```

```
[('table1',), ('table2',)]
[(0, 'id', 'integer', 0, None, 0), (1, 'schook', 'text', 0, None, 0)]
```

注意事项

1. 对于修改表的操作，如插入，删除，更新，关闭数据库前不要忘了 `commit`，否则可能无效
2. 必要时用 `try...except` 语句来避免数据库不存在，表不存在时的导致的 `runtime error`



北京大学
PEKING UNIVERSITY

信息科学技术学院

处理二进制字段



大连金石滩

设置blob字段（二进制字段）的值：

```
import sqlite3
import requests
f = open('c:/tmp/tmp.jpg', 'rb') #二进制方式打开图片
img = f.read()
f.close()
db = sqlite3.connect("c:/tmp/test2.db")
cur = db.cursor()
sql = "UPDATE students SET picture=? WHERE name = '李四'"
cur.execute(sql, (img,)) #设置李四的照片。 img对应于 ?
```


设置blob字段（二进制字段）的值：

```
imgUrl =  
"https://img5. duitang. com/uploads/item/201605/19/20160519224441_VfMWR  
. jpeg" #从网络获取图片  
imgStream = requests. get(imgUrl, stream=True)  
sql = "UPDATE students SET picture=? WHERE name = '张三'"  
cur. execute(sql, (imgStream. content,)) #设置张三的照片。  
#imgStream. content 对应于 ?  
db. commit()  
cur. close()  
db. close()
```

读取blob字段（二进制字段）的值：

```
import sqlite3
import requests

db = sqlite3.connect("c:/tmp/test2.db")
cur = db.cursor()
sql = "select name,picture from students WHERE name = '张三' or name = '李四'"
cur.execute(sql)
x = cur.fetchall()
for r in x: # r[0]是姓名, r[1]是图片文件数据
    f = open("c:/tmp/" + r[0] + ".jpg", "wb") #照片写入 张三.jpg和李四.jpg
    f.write(r[1])
    f.close()
cur.close()
db.close()
```