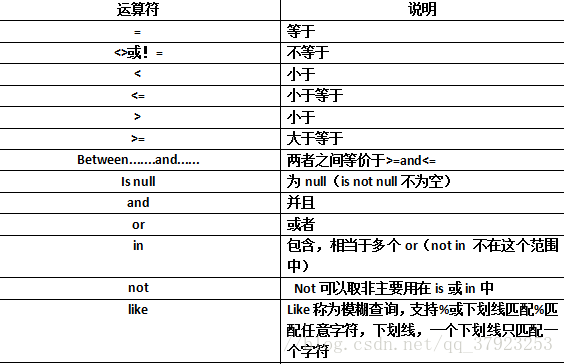
### 查询表记录



1、select \* from 表名;

2、select 字段1,字段名2,...,字段名N from 表名;

3、# 查询customers表中id介于6到8之间的数据并返回name、address字段

query = "SELECT name、address FROM customers WHERE id BETWEEN %s AND %s"

cursor.execute(query, (6, 8))

# 循环取出结果集中的每条数据并打印

for (name, address) in cursor:

print("%s家的地址是%s" % (name, address))

# 输出结果为：

# Peter家的地址是Lowstreet 4

# Amy家的地址是Apple st 652

# Hannah家的地址是Mountain 21

# [mysql倒序查询](https://www.cnblogs.com/lumama520/p/10904295.html)

通过 ”order by“实现，语法 ：order by 字段 [asc](https://www.baidu.com/s?wd=asc&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao)/desc。  
sql:select \* from AAAA order by BBBB desc;  
解释：上面语句的意思就是根据”BBBB字段“排序，倒叙输出AAAA表中的数据。  
备注：[asc](https://www.baidu.com/s?wd=asc&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao)是表示升序，desc表示降序。

sql:select \* from AAAA limit 0,10 order by BBBB desc。

通常0是可以省略的，直接写成limit 10。 0代表从第0条记录后面开始，也就是从第一条开始到第10条。

#### fetchone

有点像从管道中取一个，如果再来一个fetchone，会又取下一个，如果取完了再取，就返回None

每条记录为元组格式

下面脚本和上面增加数据，除了执行sql语句部分不一样，其余都一样

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | # 执行sql语句  rows=cursor.execute('select \* from test;')  print(cursor.fetchone())  print(cursor.fetchone())  print(cursor.fetchone())  print(cursor.fetchone())  print(cursor.fetchone()) |

运行结果：

(4, 'qzcsbj4')  
(5, '全栈测试笔记5')  
(6, '全栈测试笔记6')  
(7, 'qzcsbj')  
None

每条记录为字典格式

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | # 获取游标  cursor=conn.cursor(pymysql.cursors.DictCursor)    # 执行sql语句  rows=cursor.execute('select \* from test;')  print(cursor.fetchone())  print(cursor.fetchone())  print(cursor.fetchone())  print(cursor.fetchone())  print(cursor.fetchone()) |

运行结果：

{'id': 4, 'name': 'qzcsbj4'}  
{'id': 5, 'name': '全栈测试笔记5'}  
{'id': 6, 'name': '全栈测试笔记6'}  
{'id': 7, 'name': 'qzcsbj'}  
None

#### fetchmany

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | # 获取游标  cursor=conn.cursor(pymysql.cursors.DictCursor)    # 执行sql语句  rows=cursor.execute('select \* from test;')  print(cursor.fetchmany(2)) |

运行结果：

[{'id': 4, 'name': 'qzcsbj4'}, {'id': 5, 'name': '全栈测试笔记5'}]

#### fetchall

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | # 获取游标  cursor=conn.cursor(pymysql.cursors.DictCursor)    # 执行sql语句  rows=cursor.execute('select \* from test;')  print(cursor.fetchall())  print(cursor.fetchall()) |

运行结果：

[{'id': 4, 'name': 'qzcsbj4'}, {'id': 5, 'name': '全栈测试笔记5'}, {'id': 6, 'name': '全栈测试笔记6'}, {'id': 7, 'name': 'qzcsbj'}]  
[]

##### 相对绝对位置移动

从头开始跳过n个

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | # 获取游标  cursor=conn.cursor(pymysql.cursors.DictCursor)    # 执行sql语句  rows=cursor.execute('select \* from test;')  cursor.scroll(3,mode='absolute')  print(cursor.fetchone()) |

运行结果：

{'id': 7, 'name': 'qzcsbj'}

##### ****相对当前位置移动****

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | # 获取游标  cursor=conn.cursor(pymysql.cursors.DictCursor)    # 执行sql语句  rows=cursor.execute('select \* from test;')  print(cursor.fetchone())  cursor.scroll(2,mode='relative')  print(cursor.fetchone()) |

运行结果：

{'id': 4, 'name': 'qzcsbj4'}  
{'id': 7, 'name': 'qzcsbj'}

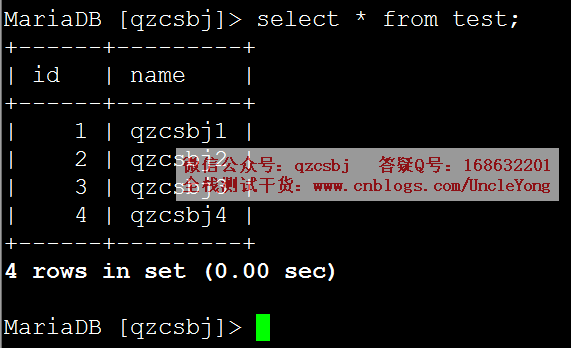
### 在表中插入记录

1、insert into 表名 values(值1),(值2),...,(值N);

2、insert into 表名(字段名列表) values(值1),...(值N);

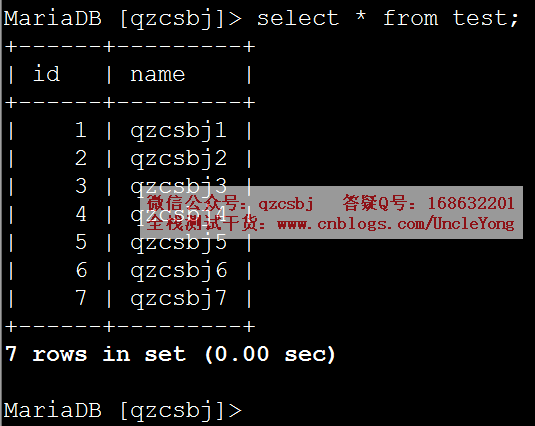
**单条**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27 | import pymysql    # 建立数据库连接  conn=pymysql.connect(      host='192.168.168.168',      port=3306,      user='root',      password='mysql',      db='qzcsbj',      charset='utf8'  )    # 获取游标  cursor=conn.cursor()    # 执行sql语句  sql='insert into test(id,name) values(%s,%s)'  rows=cursor.execute(sql,('4','qzcsbj4'))    # 提交  conn.commit()    # 关闭游标  cursor.close()    # 关闭连接  conn.close() |



**多条**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27 | import pymysql    # 建立数据库连接  conn=pymysql.connect(      host='192.168.168.168',      port=3306,      user='root',      password='mysql',      db='qzcsbj',      charset='utf8'  )    # 获取游标  cursor=conn.cursor()    # 执行sql语句  sql='insert into test(id,name) values(%s,%s)'  rows=cursor.executemany(sql,[('5','qzcsbj5'),('6','qzcsbj6'),('7','qzcsbj7')])    # 提交  conn.commit()    # 关闭游标  cursor.close()    # 关闭连接  conn.close() |



**大批量新增**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30 | import pymysql    # 建立数据库连接  conn=pymysql.connect(      host='192.168.168.168',      port=3306,      user='root',      password='mysql',      db='qzcsbj',      charset='utf8'  )    # 获取游标  cursor=conn.cursor(pymysql.cursors.DictCursor)    # 执行sql语句  values=[]  for i in range(100, 201):      values.append((i, 'qzcsbj'+str(i)))  sql='insert into test(id,name) values(%s,%s)'  rows=cursor.executemany(sql,values)    # 提交  conn.commit()    # 关闭游标  cursor.close()    # 关闭连接  conn.close() |

### 修改数据

2、更新表记录

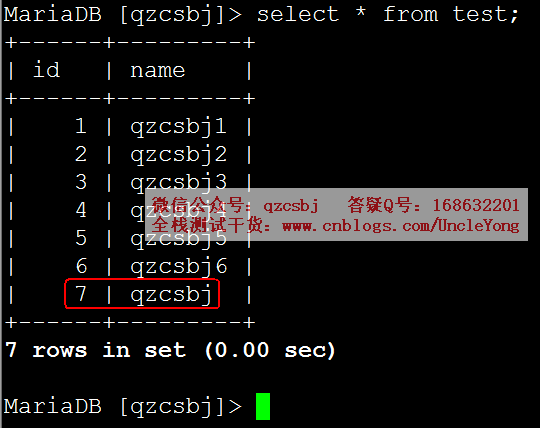
1、update 表名 set 字段名1=值1,字段名2=值2,... where 条件;

2、注意

update语句后如果不加where条件子句会将表中所有记录全部更改

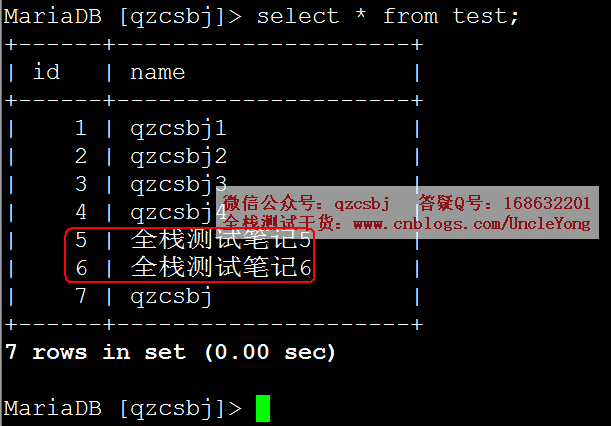
#### 单条

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27 | import pymysql    # 建立数据库连接  conn=pymysql.connect(      host='192.168.168.168',      port=3306,      user='root',      password='mysql',      db='qzcsbj',      charset='utf8'  )    # 获取游标  cursor=conn.cursor()    # 执行sql语句  sql='update test set name = %s where id = %s'  rows=cursor.execute(sql,('qzcsbj','7'))    # 提交  conn.commit()    # 关闭游标  cursor.close()    # 关闭连接  conn.close() |



#### 多条

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27 | import pymysql    # 建立数据库连接  conn=pymysql.connect(      host='192.168.168.168',      port=3306,      user='root',      password='mysql',      db='qzcsbj',      charset='utf8'  )    # 获取游标  cursor=conn.cursor()    # 执行sql语句  sql='update test set name = %s where id = %s'  rows=cursor.executemany(sql,[('全栈测试笔记5','5'),('全栈测试笔记6','6')])    # 提交  conn.commit()    # 关闭游标  cursor.close()    # 关闭连接  conn.close() |



[回到顶部](https://www.cnblogs.com/UncleYong/p/10938993.html#_labelTop)

### 删除数据

1、删除表记录

1、delete from 表名 where 条件;

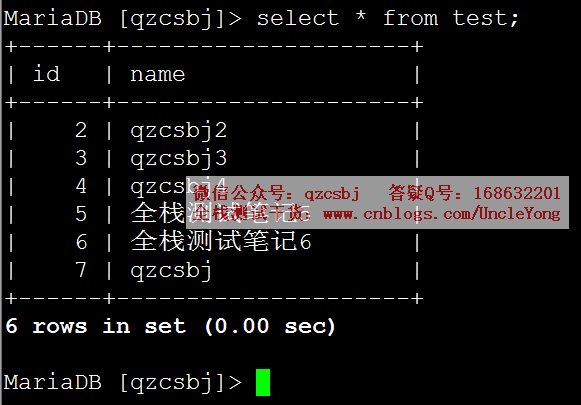
2、注意

delete语句后如果不佳where条件子句,会将表中所有记录全部删除

#### 单条

下面脚本和上面增加数据，除了执行sql语句部分不一样，其余都一样

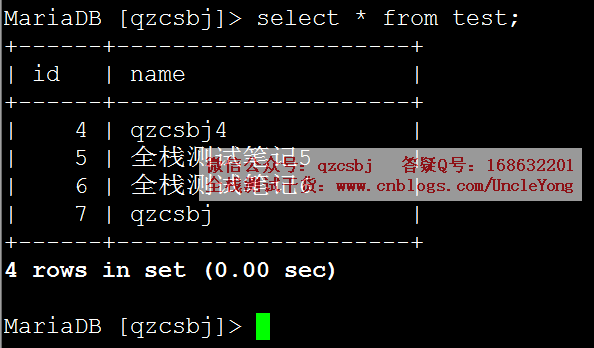
|  |  |
| --- | --- |
| 1  2  3 | # 执行sql语句  sql='delete from test where id = %s'  rows=cursor.execute(sql,('1',)) |



#### 多条

下面脚本和上面增加数据，除了执行sql语句部分不一样，其余都一样

|  |  |
| --- | --- |
| 1  2  3 | # 执行sql语句  sql='delete from test where id = %s'  rows=cursor.executemany(sql,[('2'),('3')]) |



把上面大批量新增的数据删除，delete from test where id>=100;

### 表字段的操作

1、语法：alter table 表名 执行动作;

1、添加字段

alter table 表名 add 字段名 数据类型 first | after 字段名;

2、删除字段

alter table 表名 drop 字段名;

3、修改字段数据类型

alter table 表名 modify 字段名 新数据类型;

# 修改数据类型时会受到表中原有数据的限制

4、修改字段名

alter table 表名 change 旧名 新名 数据类型;

5、修改表名

alter table 表名 rename 新表名;