

公告

wiki和教程：www.pythontav.com

免费教学视频：[B站](#)：凸头统治地球

高级专题教程：[网易云课堂](#)：武沛齐

聊技术，加武Sir微信



武沛齐

扫一扫上面的二维码图案，加我微信



Python技术交流群：737658057

软件测试开发交流群：721023555

昵称：武沛齐
园龄：8年1个月
粉丝：9974
关注：44
[+加关注](#)

我的标签

- Python(17)
- ASP.NET MVC(15)
- python之路(7)
- Tornado源码分析(5)
- 每天一道Python面试题(5)
- crm项目(4)
- 面试都在问什么？(2)
- Python开源组件 - Tyrion(1)
- Python面试315题(1)
- Python企业面试题讲解(1)

积分与排名

积分 - 426642
排名 - 785

随笔分类

- JavaScript(1)
- MVC(15)
- Python(17)
- 面试都在问什么系列？【图】(2)
- 其他(37)

随笔 - 140 文章 - 164 评论 - 893

Python开发【第十七篇】：MySQL（一）

概述

- 1、什么是数据库？
答：数据的仓库，如：在ATM的示例中我们创建了一个 db 目录，称其为数据库
- 2、什么是 MySQL、Oracle、SQLite、Access、MS SQL Server等？
答：它们均是一个软件，都有两个主要的功能：
 - a. 将数据保存到文件或内存
 - b. 接收特定的命令，然后对文件进行相应的操作

PS：如果有了以上软件，无须自己再去创建文件和文件夹，而是直接传递 命令 给上述软件，让其来进行文件操作，他们统称为数据库管理系统（DBMS，Database Management System）
- 3、什么是SQL？
答：上述提到MySQL等软件可以接受命令，并做出相应的操作，由于命令中可以包含删除文件、获取文件内容等众多操作，对于编写的命令就是是SQL语句。SQL，是结构化语言（Structured Query Language）的缩写，SQL是一种专门用来与数据库通信的语言。

二、下载安装

MySQL是一个关系型数据库管理系统，由瑞典MySQL AB 公司开发，目前属于 Oracle 旗下公司。MySQL 最流行的关系型数据库管理系统，在 WEB 应用方面MySQL是最好的 RDBMS (Relational Database Management System，关系数据库管理系统) 应用软件之一。

想要使用MySQL来存储并操作数据，则需要做几件事情：

- a. 安装MySQL服务端
- b. 安装MySQL客户端
- b. 【客户端】连接【服务端】
- c. 【客户端】发送命令给【服务端MySQL】服务的接受命令并执行相应操作(增删改查等)

```
1  下载
2      http://dev.mysql.com/downloads/mysql/
3  安装
4      windows:
5          点点点
6      Linux:
7          yum install mysql-server
8      Mac:
9          点点点
```

Window版本

1、下载

```
1  MySQL Community Server 5.7.16
2
3  http://dev.mysql.com/downloads/mysql/
```

2、解压

如果想要让MySQL安装在指定目录，那么就将解压后的文件夹移动到指定目录，如：C:\mysql-5.7.16-winx64

企业面试题及答案(1)

请求响应(6)

设计模式(9)

微软C#(34)

随笔档案

2020年6月(1)

2020年5月(1)

2019年11月(1)

2019年10月(1)

2019年9月(4)

2018年12月(1)

2018年8月(1)

2018年5月(2)

2018年4月(1)

2017年8月(1)

2017年5月(1)

2017年3月(1)

2016年10月(1)

2016年7月(1)

2015年10月(1)

2015年8月(1)

2015年7月(1)

2015年6月(2)

2015年4月(2)

2014年3月(3)

2014年1月(3)

2013年12月(2)

2013年11月(2)

2013年10月(7)

2013年8月(17)

2013年7月(1)

2013年6月(14)

2013年5月(23)

2013年4月(3)

2013年3月(13)

2013年2月(1)

2012年11月(26)

相册

git(14)

最新评论

1. Re:Python开发【第十九篇】：Python操作MySQL

执行本网页中第一段SQL语句，其中执行语句改成： cursor.execute('insert into part(caption) values("AB") ') 报错 unable to reso...

--serene1979

2. Re:python 面向对象 (进阶篇)

class Foo:

pass

这个类不是由,object.new(cls)创建的吗

3、初始化

MySQL解压后的 bin 目录下有一大堆的可执行文件，执行如下命令初始化数据：

```
1 | cd c:\mysql-5.7.16-winx64\bin
2 |
3 | mysqld --initialize-insecure
```

4、启动MySQL服务

执行命令从而启动MySQL服务

```
1 | # 进入可执行文件目录
2 | cd c:\mysql-5.7.16-winx64\bin
3 |
4 | # 启动MySQL服务
5 | mysqld
```

5、启动MySQL客户端并连接MySQL服务

由于初始化时使用的【mysqld --initialize-insecure】命令，其默认未给root账户设置密码

```
1 | # 进入可执行文件目录
2 | cd c:\mysql-5.7.16-winx64\bin
3 |
4 | # 连接MySQL服务器
5 | mysql -u root -p
6 |
7 | # 提示请输入密码，直接回车
```

输入回车，见下图表示安装成功：

```
c:\mysql-5.7.16-winx64\bin>mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.7.16 MySQL Community Server (GPL)

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> _
```

到此为止，MySQL服务端已经安装成功并且客户端已经可以连接上，以后再操作MySQL时，只需要重复上述4、5步骤即可。但是，在4、5步骤中重复的进入可执行文件目录比较繁琐，如想日后操作简便，可以做如下操作。

a. 添加环境变量

将MySQL可执行文件添加到环境变量中，从而执行命令即可

```
1 | 【右键计算机】-->【属性】-->【高级系统设置】-->【高级】-->【环境变量】-->
2 |
3 | 如：
4 | C:\Program Files (x86)\Parallels\Parallels Tools\Applications;%SystemRo
```

如此一来，以后再启动服务并连接时，仅需：

```
1 | # 启动MySQL服务，在终端输入
2 | mysqld
3 |
4 | # 连接MySQL服务，在终端输入：
5 | mysql -u root -p
```

b. 将MySQL服务制作成windows服务

--bajie_new

3. Re:MySQL练习题参考答案

14、查询和“002”号的同学学习的课程完全相同的其他同学学号和姓名； select sid,sname from (select sid,sname from student where sid i...

--BobAylIn

4. Re:MySQL练习题参考答案

13、查询至少学过学号为“001”同学所有课的其他同学学号和姓名； 需要考虑这种情况，001同学选了1、2、4，002同学选了1、2、3。这种情况，两位同学选课有交集，count(选课数)也一样，然是...

--BobAylIn

5. Re:人生没有白走的路，每一步都算数 后续来了吗

--小鱼仔。

上一步解决了一些问题，但不够彻底，因为在执行【mysqld】启动MySQL服务器时，当前终端会被hang住，那么做一下设置即可解决此问题：

```
1 # 制作MySQL的Windows服务，在终端执行此命令：
2 "c:\mysql-5.7.16-winx64\bin\mysqld" --install
3
4 # 移除MySQL的Windows服务，在终端执行此命令：
5 "c:\mysql-5.7.16-winx64\bin\mysqld" --remove
```

注册成服务之后，以后再启动和关闭MySQL服务时，仅需执行如下命令：

```
1 # 启动MySQL服务
2 net start mysql
3
4 # 关闭MySQL服务
5 net stop mysql
```

Linux版本

安装：

```
1 yum install mysql-server
```

服务端启动

```
1 mysql.server start
```

客户端连接

```
1 连接：
2 mysql -h host -u user -p
3
4 常见错误：
5 ERROR 2002 (HY000): Can't connect to local MySQL server through soc
6 退出：
7 QUIT 或者 Control+D
```

三、数据库操作

1、显示数据库

```
1 SHOW DATABASES;
```

默认数据库：

mysql - 用户权限相关数据

test - 用于用户测试数据

information_schema - MySQL本身架构相关数据

2、创建数据库

```
1 # utf-8
2 CREATE DATABASE 数据库名称 DEFAULT CHARSET utf8 COLLATE utf8_general_ci;
3
4 # gbk
5 CREATE DATABASE 数据库名称 DEFAULT CHARACTER SET gbk COLLATE gbk_chinese_ci;
```

3、使用数据库

```
1 USE db_name;
```

显示当前使用的数据库中所有表： SHOW TABLES;

4、用户管理

```
1 创建用户
2 create user '用户名'@'IP地址' identified by '密码';
3 删除用户
4 drop user '用户名'@'IP地址';
```

```
5  修改用户
6      rename user '用户名'@'IP地址'; to '新用户'@'IP地址';;
7  修改密码
8      set password for '用户名'@'IP地址' = Password('新密码')
9
10 PS: 用户权限相关数据保存在mysql数据库的user表中，所以也可以直接对其进行操作（不建
```

5、授权管理

```
1  show grants for '用户'@'IP地址'      -- 查看权限
2  grant 权限 on 数据库.表 to '用户'@'IP地址'  -- 授权
3  revoke 权限 on 数据库.表 from '用户'@'IP地址'  -- 取消权限
```

all privileges	除grant外的所有权限
select	仅查权限
select,insert	查和插入权限
...	
usage	无访问权限
alter	使用alter table
alter routine	使用alter procedure和drop procedure
create	使用create table
create routine	使用create procedure
create temporary tables	使用create temporary tables
create user	使用create user、drop user、rename user和r
create view	使用create view
delete	使用delete
drop	使用drop table
execute	使用call和存储过程
file	使用select into outfile 和 load data infi
grant option	使用grant 和 revoke
index	使用index
insert	使用insert
lock tables	使用lock table
process	使用show full processlist
select	使用select
show databases	使用show databases
show view	使用show view
update	使用update
reload	使用flush
shutdown	使用mysqladmin shutdown(关闭MySQL)
super	□□使用change master、kill、logs、purge、ma
replication client	服务器位置的访问
replication slave	由复制从属使用

对于目标数据库以及内部其他:	
数据库名.*	数据库中的所有
数据库名.表	指定数据库中的某张表
数据库名.存储过程	指定数据库中的存储过程
.	所有数据库

用户名@IP地址	用户只能在改IP下才能访问
用户名@192.168.1.%	用户只能在改IP段下才能访问(通配符%表示任意)
用户名@%	用户可以再任意IP下访问(默认IP地址为%)

```
grant all privileges on db1.tb1 TO '用户名'@'IP'

grant select on db1.* TO '用户名'@'IP'
```

```
grant select,insert on *.* TO '用户名'@'IP'

revoke select on db1.tb1 from '用户名'@'IP'
```



特殊的：

- 1 | `flush privileges`，将数据读取到内存中，从而立即生效。



启动免授权服务端

```
mysqld --skip-grant-tables
```

客户端

```
mysql -u root -p
```

修改用户名密码

```
update mysql.user set authentication_string=password('666') where user='root'
flush privileges;
```



四、数据表基本

1、创建表

```
1 create table 表名(
2     列名 类型 是否可以为空,
3     列名 类型 是否可以为空
4 )ENGINE=InnoDB DEFAULT CHARSET=utf8
```



是否可空, null表示空, 非字符串

```
not null    - 不可空
null        - 可空
```



默认值, 创建列时可以指定默认值, 当插入数据时如果未主动设置, 则自动添加默认值

```
create table tbl(
    nid int not null default 2,
    num int not null
)
```



自增, 如果为某列设置自增列, 插入数据时无需设置此列, 默认将自增 (表中只能有一个自增列)

```
create table tbl(
    nid int not null auto_increment primary key,
    num int null
)
```

或

```
create table tbl(
    nid int not null auto_increment,
    num int null,
    index(nid)
)
```

注意: 1、对于自增列, 必须是索引 (含主键)。

2、对于自增可以设置步长和起始值

```
show session variables like 'auto_inc%';
set session auto_increment_increment=2;
set session auto_increment_offset=10;

show global variables like 'auto_inc%';
```

```
set global auto_increment_increment=2;
set global auto_increment_offset=10;
```

主键，一种特殊的唯一索引，不允许有空值，如果主键使用单个列，则它的值必须唯一，如果

```
create table tbl(
    nid int not null auto_increment primary key,
    num int null
)
或
create table tbl(
    nid int not null,
    num int not null,
    primary key(nid,num)
)
```

外键，一个特殊的索引，只能是指定内容

```
creat table color(
    nid int not null primary key,
    name char(16) not null
)

create table fruit(
    nid int not null primary key,
    smt char(32) null ,
    color_id int not null,
    constraint fk_cc foreign key (color_id) references color(nid)
)
```

2、删除表

```
1 drop table 表名
```

3、清空表

```
1 delete from 表名
2 truncate table 表名
```

4、修改表

```
1 添加列: alter table 表名 add 列名 类型
2 删除列: alter table 表名 drop column 列名
3 修改列:
4     alter table 表名 modify column 列名 类型; -- 类型
5     alter table 表名 change 原列名 新列名 类型; -- 列名, 类型
6
7 添加主键:
8     alter table 表名 add primary key(列名);
9 删除主键:
10    alter table 表名 drop primary key;
11    alter table 表名 modify 列名 int, drop primary key;
12
13 添加外键: alter table 从表 add constraint 外键名称(形如: FK_从表_主表) foreign key (列名) references 主表(列名);
14 删除外键: alter table 表名 drop foreign key 外键名称
15
16 修改默认值: ALTER TABLE testalter_tbl ALTER i SET DEFAULT 1000;
```

17 | 删除默认值: ALTER TABLE testalter_tbl ALTER i DROP DEFAULT;

5、基本数据类型

MySQL的数据类型大致分为：数值、时间和字符串

```
bit[(M)]
```

二进制位 (101001), m表示二进制位的长度 (1-64), 默认m=1

```
tinyint[(m)] [unsigned] [zerofill]
```

小整数, 数据类型用于保存一些范围的整数数值范围:

有符号:

-128 ~ 127.

无符号:

0 ~ 255

特别的: MySQL中无布尔值, 使用tinyint(1)构造。

```
int[(m)] [unsigned] [zerofill]
```

整数, 数据类型用于保存一些范围的整数数值范围:

有符号:

-2147483648 ~ 2147483647

无符号:

0 ~ 4294967295

特别的: 整数类型中的m仅用于显示, 对存储范围无限制。例如: int(5), 当插入数据

```
bigint[(m)] [unsigned] [zerofill]
```

大整数, 数据类型用于保存一些范围的整数数值范围:

有符号:

-9223372036854775808 ~ 9223372036854775807

无符号:

0 ~ 18446744073709551615

```
decimal[(m[,d])] [unsigned] [zerofill]
```

准确的小数值, m是数字总个数 (负号不算), d是小数点后个数。 m最大值为65, d最

特别的: 对于精确数值计算时需要用此类型

decimal能够存储精确值的原因在于其内部按照字符串存储。

```
FLOAT[(M,D)] [UNSIGNED] [ZEROFILL]
```

单精度浮点数 (非准确小数值), m是数字总个数, d是小数点后个数。

无符号:

-3.402823466E+38 to -1.175494351E-38,

0

1.175494351E-38 to 3.402823466E+38

有符号:

0

1.175494351E-38 to 3.402823466E+38

**** 数值越大, 越不准确 ****

```
DOUBLE[(M,D)] [UNSIGNED] [ZEROFILL]
```

双精度浮点数 (非准确小数值), m是数字总个数, d是小数点后个数。

无符号:

-1.7976931348623157E+308 to -2.2250738585072014E-308

0

2.2250738585072014E-308 to 1.7976931348623157E+308

有符号:

0

2.2250738585072014E-308 to 1.7976931348623157E+308
 **** 数值越大，越不准确 ****

char (m)

char数据类型用于表示固定长度的字符串，可以包含最多达255个字符。其中m代表字符串长度。
 PS：即使数据小于m长度，也会占用m长度

varchar(m)

varchar数据类型用于变长的字符串，可以包含最多达255个字符。其中m代表该字符串的最大长度。

注：虽然varchar使用起来较为灵活，但是从整个系统的性能角度来说，char数据类型

text

text数据类型用于保存变长的大字符串，可以组多到65535 (2**16 - 1)个字符。

mediumtext

A TEXT column with a maximum length of 16,777,215 (2**24 - 1) characters.

longtext

A TEXT column with a maximum length of 4,294,967,295 or 4GB (2**32 - 1) characters.

enum

枚举类型，

An ENUM column can have a maximum of 65,535 distinct elements. (MySQL 5.7.18 and later)
 示例：

```
CREATE TABLE shirts (
  name VARCHAR(40),
  size ENUM('x-small', 'small', 'medium', 'large', 'x-large'),
);
INSERT INTO shirts (name, size) VALUES ('dress shirt','large');
```

set

集合类型

A SET column can have a maximum of 64 distinct members.

示例：

```
CREATE TABLE myset (col SET('a', 'b', 'c', 'd'));
INSERT INTO myset (col) VALUES ('a,d'), ('d,a'), ('a,d,a'),
```

DATE

YYYY-MM-DD (1000-01-01/9999-12-31)

TIME

HH:MM:SS ('-838:59:59'/'838:59:59')

YEAR

YYYY (1901/2155)

DATETIME

YYYY-MM-DD HH:MM:SS (1000-01-01 00:00:00/9999-12-31 23:59:59)

TIMESTAMP

YYYYMMDD HHMMSS (1970-01-01 00:00:00/2037 年某时)



二进制数据: TinyBlob、Blob、MediumBlob、LongBlob

更多参考：

- <http://www.runoob.com/mysql/mysql-data-types.html>
- <http://dev.mysql.com/doc/refman/5.7/en/data-type-overview.html>

五、表内容操作

1、增

```

1 insert into 表 (列名,列名...) values (值,值,值...)
2 insert into 表 (列名,列名...) values (值,值,值...),(值,值,值...)
3 insert into 表 (列名,列名...) select (列名,列名...) from 表

```

2、删

```

1 delete from 表
2 delete from 表 where id=1 and name='alex'

```

3、改

```

1 update 表 set name = 'alex' where id>1

```

4、查

```

1 select * from 表
2 select * from 表 where id > 1
3 select nid,name,gender as gg from 表 where id > 1

```

5、其他

```

1 a、条件
2     select * from 表 where id > 1 and name != 'alex' and num = 12;
3
4     select * from 表 where id between 5 and 16;
5
6     select * from 表 where id in (11,22,33)
7     select * from 表 where id not in (11,22,33)
8     select * from 表 where id in (select nid from 表)
9
10 b、通配符
11     select * from 表 where name like 'ale%' - ale开头的所有（多个字符串）
12     select * from 表 where name like 'ale_' - ale开头的所有（一个字符）
13
14 c、限制
15     select * from 表 limit 5; - 前5行
16     select * from 表 limit 4,5; - 从第4行开始的5行
17     select * from 表 limit 5 offset 4 - 从第4行开始的5行
18
19 d、排序
20     select * from 表 order by 列 asc - 根据“列”从小到大排列
21     select * from 表 order by 列 desc - 根据“列”从大到小排列
22     select * from 表 order by 列1 desc,列2 asc - 根据“列1”从大到小排列,
23
24 e、分组
25     select num from 表 group by num
26     select num,nid from 表 group by num,nid
27     select num,nid from 表 where nid > 10 group by num,nid order nid desc
28     select num,nid,count(*),sum(score),max(score),min(score) from 表 group
29
30     select num from 表 group by num having max(id) > 10
31
32     特别的: group by 必须在where之后, order by之前
33
34 f、连表
35     无对应关系则不显示
36     select A.num, A.name, B.name
37     from A,B
38     where A.nid = B.nid
39
40     无对应关系则不显示
41     select A.num, A.name, B.name
42     from A inner join B
43     on A.nid = B.nid
44
45     A表所有显示, 如果B中无对应关系, 则值为null

```

```
46      select A.num, A.name, B.name
47      from A left join B
48      on A.nid = B.nid
49
50      B表所有显示，如果B中无对应关系，则值为null
51      select A.num, A.name, B.name
52      from A right join B
53      on A.nid = B.nid
54
55  g、组合
56      组合，自动处理重合
57      select nickname
58      from A
59      union
60      select name
61      from B
62
63      组合，不处理重合
64      select nickname
65      from A
66      union all
67      select name
68      from B
```



作者：武沛齐
出处：<http://www.cnblogs.com/wupeiqi/>
本文版权归作者和博客园共有，欢迎转载，但未经作者同意必须保留此段声明，且在文章页面明显位置给出原文连接。

好文要顶

关注我

收藏该文



武沛齐
关注 - 44
粉丝 - 9974
[+加关注](#)

260

posted @ 2016-07-28 06:40 武沛齐 阅读(37664) 评论(7) 编辑 收藏

评论列表

- #1楼 2017-10-31 19:29 徐来 回复 引用
写得很详细 谢谢!
支持(0) 反对(0)
- #2楼 2018-07-26 19:41 年轻人——001 回复 引用
银角大王
支持(0) 反对(0)
- #3楼 2018-11-12 21:47 吃瓜的瓜农 回复 引用
博主
默认值 default 写成了 defalut
支持(2) 反对(0)
- #4楼 2018-12-01 23:22 随性岁分享 回复 引用
大王写的很详细,学习了
支持(0) 反对(0)
- #5楼 2019-05-21 19:57 baozhilv 回复 引用

```
1. 却少删除数据库
drop database database_name

2. 修改用户错了
rename user '用户名'@'IP地址'; to '新用户名'@'IP地址';;

to 前面和最后都多个 1 个分号
```

支持(2) 反对(0)

#6楼 2019-11-07 17:39 杨鸿儒
非常棒!!!!!!

回复 引用

支持(0) 反对(0)

#7楼 2019-12-18 15:11 Daisy0x00
修改密码:
set password for '用户名'@'IP地址' = '新密码';





回复 引用

支持(0) 反对(0)

最新评论 刷新页面 返回顶部

发表评论

编辑 预览

B    

支持 Markdown

提交评论 退出 订阅评论

[Ctrl+Enter快捷键提交]

- 【推荐】超50万行VC++源码：大型组态工控、电力仿真CAD与GIS源码库
- 【推荐】了不起的开发者，挡不住的华为，园子里的品牌专区
- 【推荐】Java经典面试题整理及答案详解（二）

相关博文：

- Python之路【第十七篇】 Django进阶篇
- Python开发【第十七篇】：MySQL（一）
- Python开发【第十七篇】：MySQL（一）
- Python开发：MySQL（一）
- Python开发【第十七篇】：MySQL（一）
- » 更多推荐...

最新 IT 新闻：

- 马斯克的Neuralink 是先知的指引还是无知的妄想？
- 2020年度国家“杰出青年”公布， 21人计算机领域贡献突出
- 腾讯不想把半条命给合作伙伴了？
- 听小米员工讲述他们所亲历的小米10年

· [一边退场一边上市 新造车进入“季后赛”](#)
» [更多新闻...](#)