P3 mySQL入门10条基本语句

1. 连接服务器
2. 查看所有库
3. 创建库
4. 删除库
5. 选择数据库
6. 查看表
7. 创建简单表
8. 删除表
9. 改表名
10. 清空表

连接服务器 mysql> -uusername -ppasswd 来连接

当连上服务器之后首先面对的是？

答：是库，库有一个或多个，因此我们要想对表/行做操作的话，得先选库

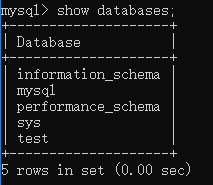
如果不知道有哪些库，想查看一下所有的库，怎么办？

mysql> show databases;

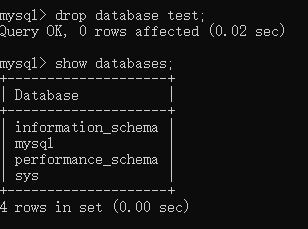
选库语句：use 库名

自己创建一个数据库：create database 数据库名 [charset 字符集]





删除一个数据库：drop database 数据库名;



把数据库改名？

mysql中，表/列可以改名，database不能改名

rename table oldname to newname

当选了库之后，我们面对的是表

查看库下面的所有表：show tables;

先来一个简单的建表语句，供练习用：

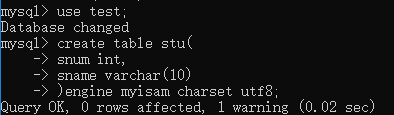
create table stu(

snum int,

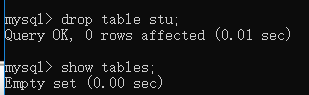
sname varchar(10)

)engine myisam charset utf8;

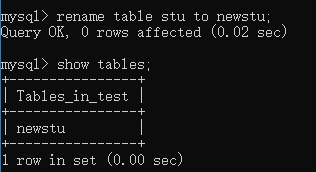
///engine是指表引擎，和性能特点相关，此处先照抄



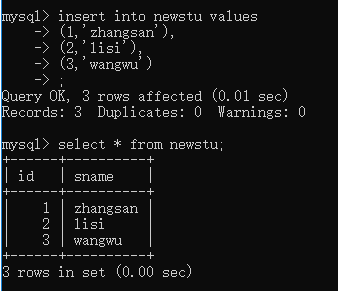
删除表：drop table stu;



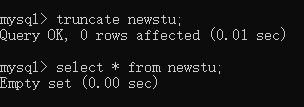
修改表名：rename table oldname to newname;



在表中插入三行数据：



清空表数据：truncate 表名;



truncate和delete是有区别的，

在于truncate相当于删表再重建一张同样结构的表，操作后得到一张全新表，

而delete是从删除数据行的层面来操作的

truncate相当于把旧的学籍表扔了重画一张，

delete相当于用橡皮把学籍表的数据擦掉

如果决定全清空，truncate速度更快

P4 练习中可能遇到的问题

乱码，告诉服务器，客户端使用的是GBK编码

set names gbk;

sql语句可以换行，遇到“；”时认为语句结束

如果语句打错了，怎么办？

\c跳出执行，再重新打

1064错误 为语法错误

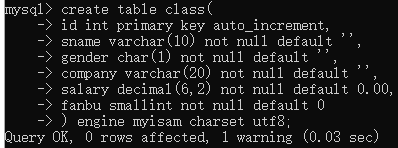
P5-P7 增删改查

tee E:\0801.sql

tee 这句话是把敲的sql及结果都输出到一个sql文件里，便于复习

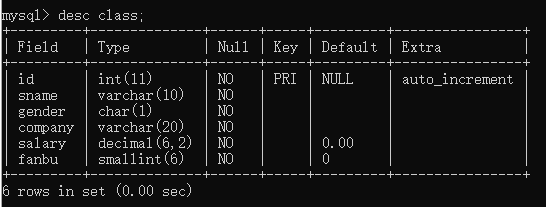
接下来学习增删改查基本语法

得先有一张表才行，先创建一张班级薪水登记表



show tables;

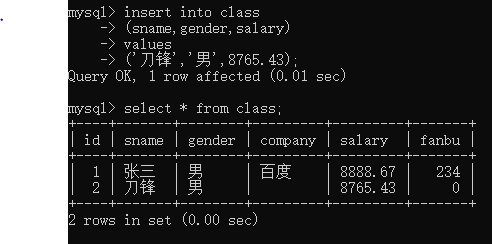
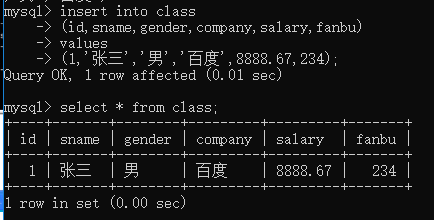
查看一下class表的结构：desc 表名;



往那张表添加行：class

添哪几列 (id, sname, gender, company, salary, fanbu)

分别添加什么值 (1,“张三”，“男”，“百度”，8888.67，234)

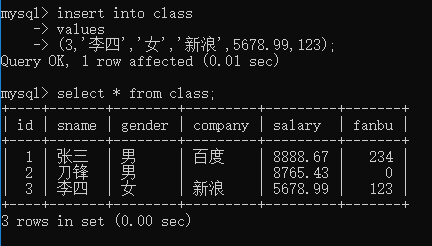


id在上例中虽然没有插入，但id是自增型，因此值为2

回头再来看，插入所有列的情况

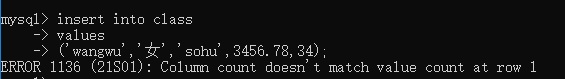
如果插入所有列，则可以不声明待插入的列

即，如果不声明插入的列，则理解为一次插入所有列



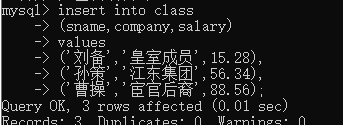
提醒不要犯如下错误：

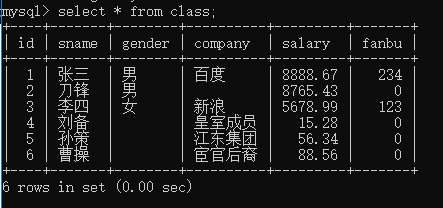
认为id是自增型的，插入时不必为其赋值



这是一种常见的错误，列与值必须按顺序一一对应

插入多行：





练习改 update

改哪张表：class

改哪几列：gender, company

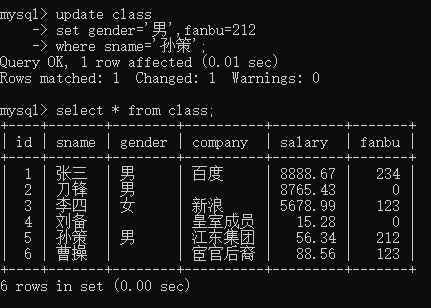
改成什么值：‘女’，‘千度’

改哪一个记录：where expression

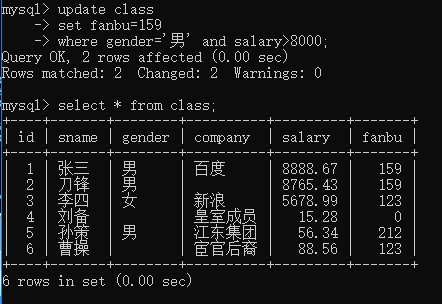
where就只能用id=N这种形式，错误的理解，

where expression,表达式

只要where后的表达式为真，则改行就发挥作用



改性别为男，且工资>8000的用户



这句话是什么意思？

update class set fanbu=99 where 1;

1恒为真，全部都要改

删除的学习

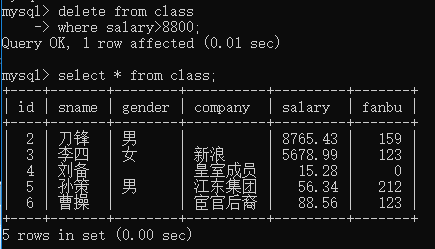
删除就是指删除整行，不存在删除一行中的某几列

删除要素

删哪张表的数据：class

删哪几行：where expression

把薪水大于8800元的删掉



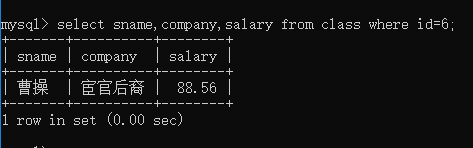
注意：delete from class;

表中的数据全部删除

查询3要素

查哪张表的数据？class

查哪些列？sname,salary,company



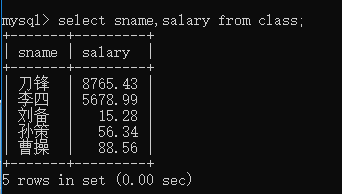
select \* from class;

查所有行，所有列

\*代表所有列，表名后不加where条件，则选择所有行，所有列

取部分列，所有行，取所有人的姓名和工资

select sname,salary from class;



查id>3的人的所有列

select \* from class where id>3;

这是最基本的增删改查

针对单个表来说

后面还要学什么？

如何自己建表？

如何修改表（增加减少列等）

多表联查

子查询

触发器

事务

存储过程

备份恢复

# P8

目的：要学会建表

知识点：列类型

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 学号 | 姓名 | 地址 | 入学年份 | 自我介绍 |
| 001 | 张三 | 衡水 | 2012 |  |

怎么建表？

分析：我们只要把第一行表头建好了，这张表就完成了

至于下面的001，张三，这不是表的概念，而是表中存储的数据

其实建表过程，就是一个画表头的过程

从术语上讲，这张表有4个列，建表的过程就是一个声明字段的过程

那么建表和列类型又有什么关系呢？

分析：再看上面的表，A4纸是数据的存储空间，而A4的大小是有限的，

请问：你准备给学号留多宽？给姓名留多宽？自我介绍又留多宽？

自然的，姓名如果留的过宽，比如20个字都能存，但是一般人的姓名就三四个字……浪费了，

如果留的过窄，导致存不下，更有问题

对应的A4纸空间有限，硬盘空间有限，

我们建列时，自然想的是……能够容纳放置的内容，但是又不浪费

存储同样的数据，不同的列类型，所占据的空间和效率是不一样的……这就是我们建表前

要学列类型的意义

所以……重点学列类型的存储范围与占据的字节关系

MySQL三大列类型

数值型

整型

tinyint

占据空间：1字节，存储范围：-128~127，0~255

tinyint 1个字节 8个位

【】【】【】【】【】【】【】【】

0000 0000 ---> 0

1111 1111 --->2^8-1=255

计算机为了表示一个数是负数，会把最高为（左侧）的0/1，当成符号

来看，如为0，则是正数，如为1，则是负数

0 0000000

0 1111111 --->127

1 0000000 ---> -0

1 1111111 ---> -127

-127~127

二进制补码的问题

如上理解的话，+0，-0，则重复了，浪费了一种存储的可能性，

因此计算机中的负数，不是照着“后面的绝对值直接乘-1得到的”

而是用不嘛规则换算的

负数=绝对值位-128

1111 1111 ---> 127-128=-1

1000 0000 --->0-128=-128

因此最终的结果是：-128~127

smallint

mediumint

int

bigint

1个字节 8个位 0~2^8-1 , 0~255

-2^7~+2^7-1

分析

2个字节，16位 0~2^16-1=65535

-2^15~2^15-1, -32768~32767

一般而言，设某类型N字节

N字节，则8N位

0 ----->2^8N-1

-2^(8N-1) -----> +2^(8N-1)-1

对于int型：占的字节越多，存储的范围也越大

声明：tinyint怎么样让它0-255，又怎么样让它-128 ~127？

int系列声明时的参数：

1. unsigned zerofill

学习tinyint的参数并验证字节与范围的关系

《见笔记》

小数型（浮点型/定点型）

float(D,M), decimal(D,M)

浮点数在计算机里表示是比较复杂的

234.87

23.487

小数点左边 能大到多少？

小数点右边 又能大到多少？

M叫做“精度” ---->代表“总位数”，而D是“标度”，代表小数位（小数右边的位数）

float(6,2)表示-9999.99到9999.99

浮点数占多大的空间呢？

答：float能存10^38, 10^-38

如果M值<=24，占4个字节，否则占8个字节

用来表示数据中的小数 除了float 浮点

还有一种叫定点decimal，定点是把整数部分和小数部分分开存储的，比float精确

123456789.987654321 占8个字节

字符串型

char, varchar, text, blob

char(6) 定长字符串

查找行记录时，如果都是定长

完全可以通过行数与行的长度计算出来文件指针的偏移量.

不论够不够指定的长度，实际都占据N个长度.

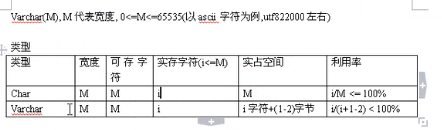
如果不够N个长度，用空格在末尾补至N个长度

varchar(100)也是存储0-100个字符

生活中坐车

不分远近统一投币1元，公司来说效率高，省了一个售票员

而9开头的长途车，加一个售票员专门来记录客户的站数



在取出特点上

char型，如果不够M个字符，内部用空格补齐，取出时再把右侧空格删掉

注：这意味着，如果右侧本身有空格，将会丢失

char的取出速度更快

注意：char(M)和varchar(M)限制的是字符，不是字节

即char(2) charset utf8, 能存2个utf8字符，比如“中国”

text: 文本类型，可以存比较大的文本段，搜索速度稍慢

因此，如果不是特别大的内容，建议用char, varchar来代替

text不用加默认值（加了也没用）

blob是二进制类型，用来存储图像、音频等二进制信息

意义：2进制，0-255都有可能出现

blob类型防止因为字符集的问题导致信息丢失

比如：一张图片中有0xFF字节，这个在ascii字符集认为非法，在入库的时候，被过滤了

# P13

日期/时间类型

P14 建表案例

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 主键id | 用户名 | 性别 | 体重(KG) | 生日 | 工资 | 上次登录时间 | 个人简介 |
| id | username | gender | weight | birth | salary | lastlogin | intro |

|  |  |  |  |
| --- | --- | --- | --- |
| 列名称 | 列类型 | 默认值 | 是否主键 |
| id | int unsigned |  | PRI |
| username | varchar(20) |  |  |
| gender | char(1)/tinyint |  |  |
| weight | tinyint unsigned |  |  |
| birth | date |  |  |
| salary | decimal(8,2) |  |  |
| lastlogin | datetime |  |  |

这张表不够好，可以优化

这张表除了username和intro列之外，每一列都是定长的

我们不妨让其所有列都定长，可以极大提高查询速度

|  |  |  |  |
| --- | --- | --- | --- |
| 列名称 | 列类型 | 默认值 | 是否主键 |
| id | int unsigned |  | PRI |
| username | char(20) |  |  |
| gender | char(1)/tinyint |  |  |
| weight | tinyint unsigned |  |  |
| birth | date |  |  |
| salary | decimal(8,2) |  |  |
| lastlogin | int unsigned |  |  |

username char(20)是会造成空间的浪费，但是提高了速度

intro char(1500)却浪费的太多了，另一方面，人的简介，一旦注册完成改的频率也

并不高，我们可以把intro列单独拿出来，另放一张表里

|  |  |  |  |
| --- | --- | --- | --- |
| 列名称 | 列类型 | 默认值 | 是否主键 |
| id | int unsigned |  | PRI |
| username | char(20) |  |  |
| intro | varchar（1500） |  |  |

在开发中，会员的信息优化往往是把频繁用到的信息，优先考虑效率，存储到一张表中

不常用的信息和比较占据空间的信息，优先考虑空间占用，存储到辅表中

建表语法：所谓建表就是一个声明列的过程

create table 表名 (

列1声明 列1参数，

列2声明 列2参数，

……

列n声明 列n参数

）engine myisam/innodb/bdb charset utf8/gbk/latin1……

create table t7 (

id int unsigned auto\_increment primary key,

username char(20) not null default '',

gender char(1) not null default '',

weight tinyint unsigned not null default 0,

birth date not null default '0000-00-00',

salary decimal(8,2) not null default 0.00,

lastlogin int unsigned not null default 0

)engine myisam charset utf8;

作业：让我们建一个电子商城，如何来设计商品表？

再把商城表的字段一个个删掉，再一个个加上，并穿插该列操作

如果列类型改变，导致数据存不下怎么办？

比如：int 改成smallint 列，如果不匹配，数据会丢失，或者再Mysql的strict mode下，修改不了

P15 修改表的语法

一张表创建完毕，有了N列

之后还有可能要增加或删除或修改列

alter table 表名 add 列名称 列类型 列参数 【加的列在表的最后】

例： alter table m1 add birth date not null default '0000-00-00';

alter table 表名 add 列名称 列类型 列参数 after 某列 【把新列加在某列后】

例： alter table m1 add gender char(1) not null default '' after username;

alter table 表名 add 列名称 列类型 列参数 after 某列 【把新列加在最前面】

例：alter table m1 add pid int not null default 0 first;

删除列：

alter table 表名 drop 列名

修改列类型：

alter table 表名 modify 列名 新类型 新参数

例：alter table m1 modify gender char(4) not null default '';

modify不能改列名

修改列名及列类型

alter table 表名 change 旧列名 新列名 新类型 新参数

例： alter table m1 change id uid int unsigned ;

# P16

select 5种子句介绍

where条件查询

group by 分组

having筛选

order by排序

limit 限制结果条数

分析商城的表并建立类似的小型表

商品表

good\_id商品主键

cat\_id 分类ID

goods\_name

goods\_sn 商品货号

goods\_name\_style 商品名称上加粗加亮

click\_count点击量

brand\_id品牌ID

provider\_name供应商名称

goods\_number库存量

goods\_weight重量

market\_price市场价格

shop\_price本店价格

promote\_price优惠价格

promote\_start\_date优惠起始日期

promote\_end\_date 优惠结束日期

warn\_number商品库存为多少时进行提醒

keywords关键字

goods\_brief商品的简短描述

goods\_desc 商品的详细描述

goods\_thumb小图

goods\_img中等图

original\_img原始图

is\_real是否为真实商品

extension\_code

is\_on\_sale 是否销售/上架

is\_alone\_sale 是否单独销售

is\_shipping 是否包邮

integral 积分

add\_time 添加的时间戳

sort\_order 排序权重

is\_delete 是否删除

is\_best 是否精品

is\_new 是否新品

is\_hot 是否热卖

is\_promote

bonus\_type\_id 优惠券

last\_update上次修改时间

goods\_type商品类型

good\_id商品主键

cat\_id 分类ID

goods\_name

goods\_sn 商品货号

click\_count

goods\_number

market\_price

shop\_price

add\_time

is\_best

is\_new

is\_hot

查询模型（重要）

列是变量，在每一行上，列的值都在变化

where条件是表达式，在哪一行上表达式为真，哪一行就取出来

一道面试题

有如下表和数据

把num值处于[20,29]之间的改为20

num值处于[30,39]之间的改为30

num

3

12

15

25

23

29

34

37

32

45

48

52

1. ,(12),(15),(25),(23),(29),(34),(37),(32),(45),(48),(52)

练习题：

把goods表中商品名为‘诺基亚xxx’的商品，改为‘HTCxxx’,

提示：大胆的把列看成变量，参与运算，甚至调用函数来处理

substring(),concat()

# P19

select count(\*) from 表名，查询的就是绝对的行数，哪怕

某一行所有字段全为NULL，也计算在内

而select count（列名）from表名

查询的是该列不为null的所有行的行数

用count(\*),count(1)谁好呢？

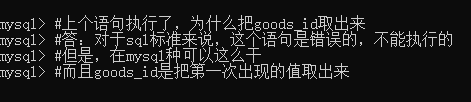
其实对于myisam引擎的表，没有区别

这种引擎内部有一个计数器在维护着行数

Innodb的表，用count(\*)直接读行数，效率很低，因为Innodb

真的要去数一遍

group



思考：

全班同学排队

校长对老师说，统计班级同学的姓名和平均年龄（返回1行）

语义上的疑问：平均年龄好算，只有一个结果，但是，把谁的姓名和

平均年龄放在一块儿返回呢？

语义上解释不通，但是mysql中偏偏可以取姓名，而且是把队伍中第一位同学的姓名返回

这是mysql的一个特点，出于可以执行和规范性，不推荐这么写

严格的讲，select的a,b列必须在group by a,b,c的列里出现

以group by a,b,c为列，则select的列只能在a,b,c里选择，语义上才没有矛盾

P20 having

