20160823

-- 双中划线+空格: 注释(单行注释),也可以用#号

### **登录**

Mysql -u root -p

### 查看数据库版本

mysql> SELECT VERSION();

+-----------+

| VERSION() |

+-----------+

| 5.6.17 |

+-----------+

1 row in set

### 查看当前时间

mysql> select now();

+---------------------+

| now() |

+---------------------+

| 2016-08-23 09:42:13 |

+---------------------+

1 row in set

### 插件当前用户

mysql> SELECT USER();

+----------------+

| USER() |

+----------------+

| root@localhost |

+----------------+

1 row in set

### MySQL语句规范

- 关键字与函数名称全部大写

- 数据库名称、表名称、字段名称全部小写

- SQL语句必须以分隔符 ; 结尾

- 只要不把单词、标记或引号分隔符为两部分，可以在下一行继续编写

- 数据库名称、表名称、字段名称尽量不要使用MySQL的保留字，如果需要使用

的使用的时候反引号引起来使用

### **-- 创建数据库**

CREATE DATABASE [IF NOT EXISTS] db\_name [[DEFAULT] CHARACTER SET[=] charset\_name];

- IF NOT EXISTS 检测是否存在

- DEFAULT CHARACTER SET 设置默认字符集

create database mydatabase charset utf8;

### **-- 查看所有数据库**

show databases;

-- 查看指定部分数据库

show databases like 'pattern' -- pattern 是匹配模式

-- %:表是匹配多个字符

-- \_:表示匹配单个字符

### **-- 创建数据库**

create database informationtest charset utf8;

### **-- 查看以information\_开始的数据库**

show databases like 'information\_%';

show databases like 'information\\_%';-- 相当于information%

### **-- 查看数据库的创建语句**

show create database mydatabase;

- 数据库名称、表名称、字段名称尽量不要使用MySQL的保留字，如果需要使用

的使用的时候反引号引起来使用

-- show create database `mydatabase`;

### **-- 更新数据库**

### **-- 数据库的修改仅限库选项:字符集和校对集(校对集依赖字符集)**

Alter database 数据库名字 [库选项]

Charset/character[=]字符集

Collate 校对集

### **-- 修改数据库informatiotest的字符集**

alter database informationtest charset GBK;

### **-- 删除数据库**

drop database informationtest;

### **-- 表操作**

**-- 表与字段密不可分**

**-- 任何表的设计都必须指定数据库**

### **-- 新增数据表**

Create table [if not exists] 表名(

字段名字 数据类型,

字段名字 数据类型 --最后一行不需要逗号

)[表选项];

if not exists : 如果表名不存在那么就创建,否则不执行创建代码;检查功能

### **表选项: 控制表的表现**

字符集:charset/character set 具体字符集;-- 保证表中的数据存储的字符集

校对集:collat 具体校对集

存储引擎:engine 具体的存储引擎(innodb和myisam)

create table if not exists student(

name varchar(10),

gender varchar(10),

number varchar(10),

age int

)Charset utf8;-- 无法创建,没有选定数据库

### **方案一: 显示的指定表所属的数据库**

Create table 数据库名.表名(); -- 将当前数据表创建到指定的数据库下

create table if not exists mydatabase.student(-- 显示的将student表放到mydatabase下

name varchar(10),

gender varchar(10),

number varchar(10),

age int

)Charset utf8;

### **方案二: 隐试的指定表所属的数据库:先进入到某个数据库环境,然后这样创建的表会自动归属到某个指定的数据库**

**进入数据库环境:use 数据库名字; -- 可以没有分好**

-- 创建数据表

-- 进入数据库

use mydatabase;

### **-- 创建表**

create table class(

name varchar(10),

room varchar(10),

number varchar(10)

)Charset utf8;

### **-- 当创建数据表的SQL指令执行之后到底发生了什么?**

1. 指定数据库下已经存在相应的表

2. 在数据库对应的文件夹下,会产生对应表的结构文件(跟存储引擎有关系)

### **-- 查看数据表**

数据库能查看的方式,表都可以查看.

查看所有表:show tables;

查看部分表:show tables like 'pattern';-- 效率低

### **-- 查看表的创建语句:**

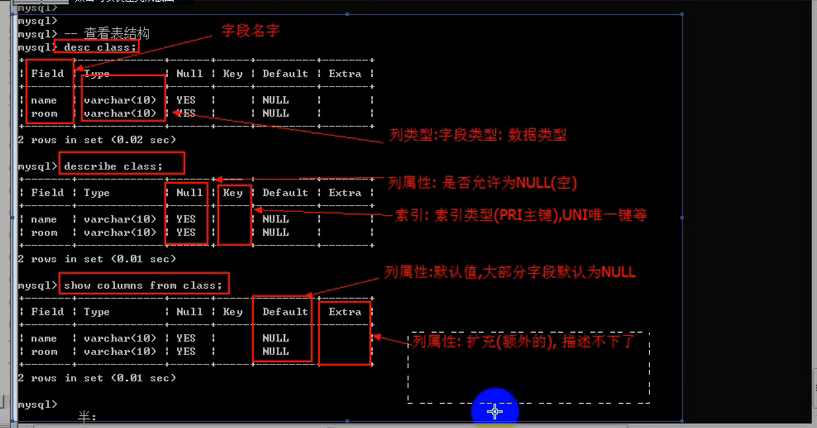
show create table 表名;

show create table 表名\g;-- \g === ;

show create table 表名\G -- \G 将查到的结构旋转90度

### **--查看表结构:查看表中的字段信息**

Desc/describe/show columns from 表名;



### **--修改数据表**

**表本身存在,还包含字段:表的修改分为两部分:修改表本身和修改字段.**

### **修改表本身:**

表本身可以修改:表名和表选项;

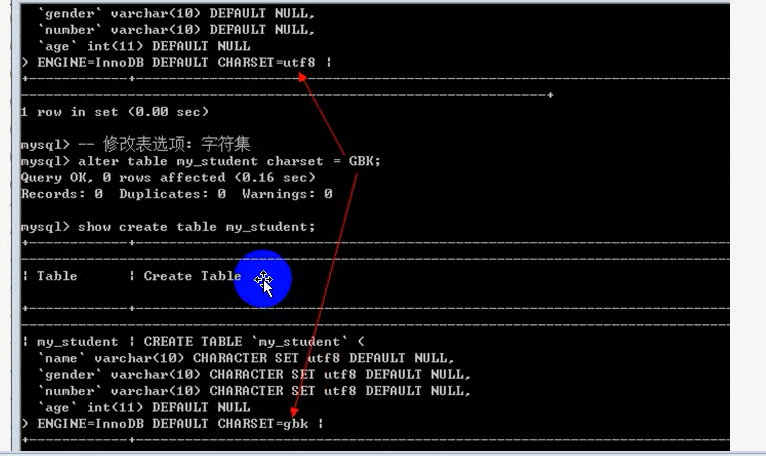
### **修改表名:rename 老表名 to 新表名**

rename table student to my\_student;(去数据库名字的前两个字)

### **修改表选项:字符集,校对集和存储引擎**

### 字符集

Alert table my\_student charset = GBK;



### **修改字段:**

字段操作很多:新增,修改,重命名,删除.

### **新增字段**

Alter table 表名 add [column] 字段名 数据类型 [列属性] [位置] --[]代表可有可无, 新增字段必须要有数据类型.

位置:字段名可以存放在表中任意位置

First:第一个位置

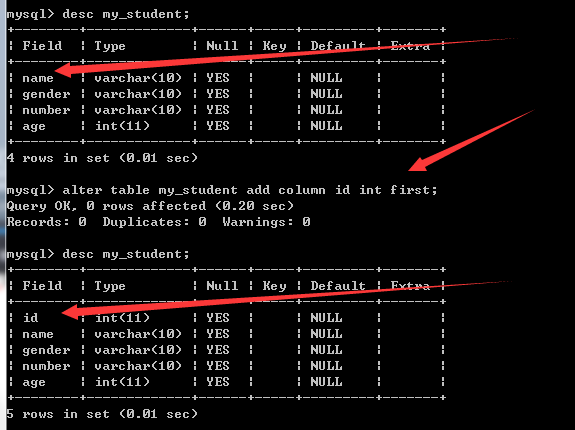
After:在哪个字段名之后:after字段名;默认的是在最后一个字段之后

### **--给学生表增加ID放到第一个位置**

Alter table my\_student

Add column id int

First; --mysql会自动1分号:语句结束符



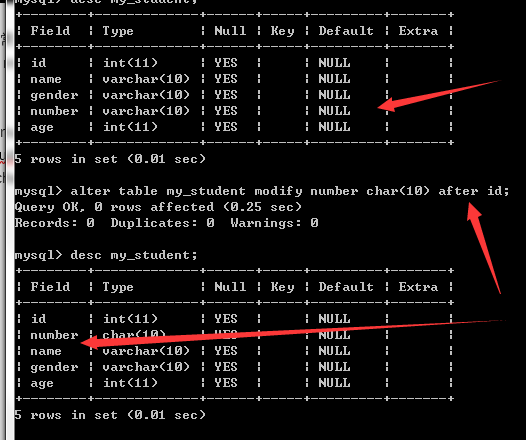
### **修改字段:修改通常是修改属性或者数据类型**

Alter table 表名 modify 字段名 数据类型 [属性] [位置]

### **--将学生表中的number学号字段变成固定长度,且放到第二位(id之后)**

Alter table my\_student

Modify number char(10) after id;



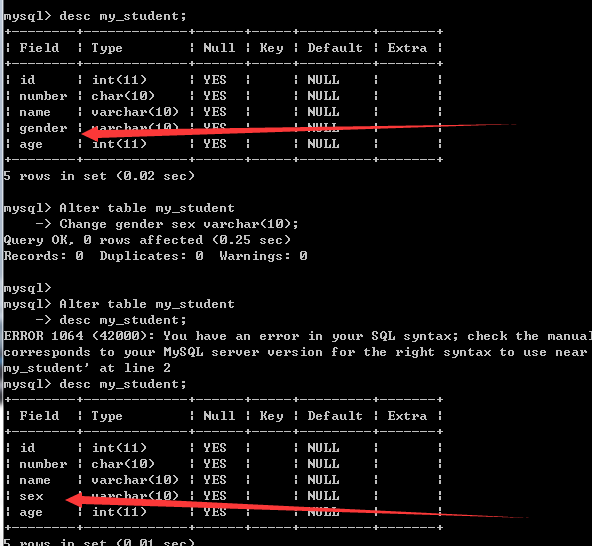
### **重命名字段**

Alter table 表名 change 旧字段 新字段名 数据类型 [属性] [位置];

### **修改学生表中的跟的字段为sex**

Alter table my\_student

Change gender sex varchar(10);

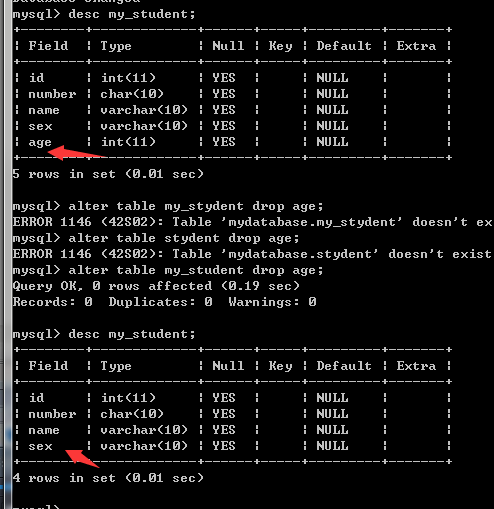


### **删除字段**

Alter table 表名 drop 字段名;

### **--删除学生表中的年龄字段(age)**

Alter table my\_student drop 阿哥;



小心 :如果表中已经存在数据,那么删除字段会清空该字段的所有数据(不可逆)

### 插入数据

Insert into 表名(字段1,字段2…) value(值1,值2…);

### 查询表中所有数据

Select \* from 表名;

### 设置数值型字段无符号

unsigned

### 字段类型

一.数值型

1.整形

1.非常小整形 tinyint -128~127 / 0~255 1 字节

2.较小整形 small

3.中等的整形 mediumint

4.普通整形 int

5.大整形 bigint

2.浮点型

DOUBLE(M , D) M表示总长度 , D表示小数位长度 , 保留小数位是会四舍五入

DECIMAL(M , D) M表示总长度 , D表示小数位长度 , 也会四舍五入 , 但会发出警告

Double 的效率高些,但是精度差一些

Decimal 效率低一些,但是精度高

浮点型数据,尽量不要使用 等于 来进行判断,尽量选择范围运算(比较运算)

Double(5 , 2)

999.99

二.字符串型

1.变长字符串 varchar[(长度)] 0~255之间

2.定长字符串 char[(长度)] 0~255之间

Varchar(4) char(4)

‘’ 1 ‘’ 4

‘ab ’ 3 ‘ab’ 4

‘abcd’ 5 ‘abcd’ 4

‘ab ‘ ‘ab ‘

保留空格 不保留空格

效率较低 效率高

占用空间少 占用空间多

3.超长字符串 L+2个字节 < 2 的 16 次方

4.枚举字符串

可以传值,也可以传对应下标,对应下标从1开始

三.日期型

1.DATE

2.TIME

3.DATETIME

4.TIMESTAMP

一般数据库存储时间时,利于计算,存储空间小

# 20160824

### 字段约束

Unsigned 无符号

Zerofill 前导0 ,设置属性时,默认添加 无符号unsigned

Create table 表名(

One int(4) zerofill not null default 0

);

Not null 非空

Default 默认值

非空和默认值应该配合使用

Auto\_increment 与主键配合使用,每次插入新记录时在最大值的基础上+1

更新数据

update table12 set gender = 1 where id = 5;

删除数据,但是不能更改 auto\_increment 的值

delete from 表名 where 条件;

truncate 清空表,输出数据,并重置 auto\_increment

### 索引

主键索引

Primary key

保证数据唯一性,通常与 auto\_increment 配合使用

创建索引

在字段中创建索引可以提高查询速度，保证数据唯一，提高全文检索速度等

- 主键索引

- 唯一索引

- 常规索引

Show index from 表名;

字段索引 - 主键

- PRIMARY KEY

- 主键保证记录的唯一性，通常会设置 AUTO\_INCREMENT 属性

- 主键自动为NOT NULL

添加主键

- ALTER TABLE 表名 ADD PRIMARY KEY(id);

删除主键

- ALTER TABLE 表名 DROP PRIMARY KEY;

当设置了auto\_increment是,主键不能直接被删除,需要去除自增属性再删除.

唯一索引

Unique

添加方式

建表语句

一:在字段描述中直接添加 unique

二:unique 可以索引名(字段1,字段2……) 一个索引名指定多个字段

修改语句

Alter table 表名 add unique(‘字段名’);添加唯一索引

Alter table 表名 drop index s索引名; 删除唯一索引

应用:常用在where 条件的字段,或者 order by(排序) 的字段

不能是经常修改的字段.

注:当数据由于唯一索引而插入失败时,自增字段依然会增加

普通索引(常规索引)

Index / key’

添加方式:

建表语句:

一:在字段描述中直接添加

二:index / key 索引名(字段1 , 字段2……) 一个索引名,指定多个字段

删除索引名是多个字段的索引都会被取消.

修改语句

Alter table 表名 add index 索引名(字段1 , 字段2……);添加

Alter table 表名 drop index 索引名; 删除唯一索引

### 数据表操作

修改列

Alter table 表名 modify 字段 类型 [字段描述]; 修改字段

修改列名称

ALTER TABLE 表名 CHANGE [COLUMN] 旧名称 新字段 新字段描述；

删除列

ALTER TABLE 表名 DROP [COLUMN] 字段;

添加列

添加单列

- ALTER TABLE 表名 ADD [COLUMN] 字段 字段描述 [after 字段];

添加多列

- ALTER TABLE 表名 ADD [COLUMN] (字段1 字段描述1,...);

- 多个列和描述用逗号分隔

Comment注释

### 存储引擎

Create table 表名(

字段

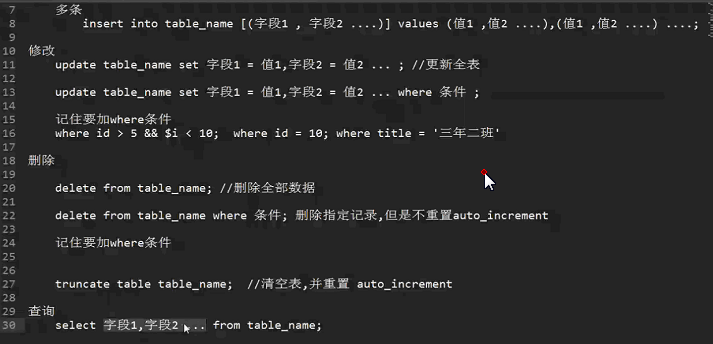
)engine = 存储引擎 default charset = utf8;

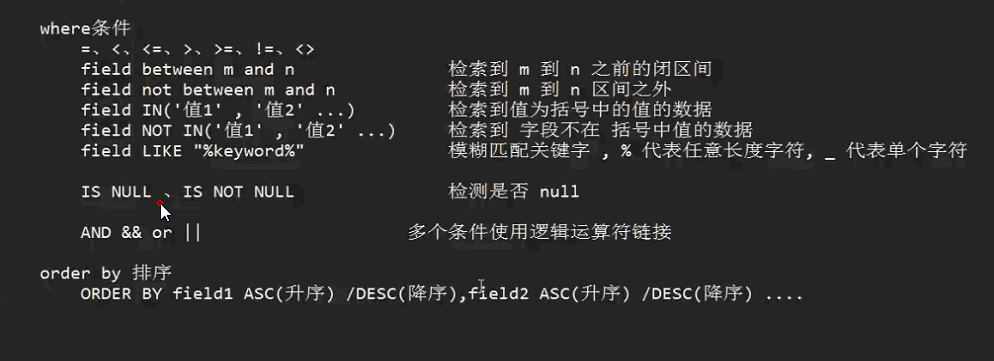
修改语句

Alter table 表名 default charset = utf8;

### 查询

Select 字段1,字段2……from 表名;





### Limit 截取条数

Limit $offset,$num 从 $offset 位置开始取 $num 条,如果只有一个值$num,代表从0的位置开始取

### 聚合函数

AS 取别名

Count($field) 统计总数

Avg($field) 计算平均数

Sum($field) 计算总和

Min($field) 找到最小值

Max($field) 找到最大值

Group by 分组

Group by $field[,$field] [having $条件] 可以对一个字段的想同值进行分组,可以使用count()来统计出现次数,可以选择使用having对分组的结果进行设置条件

关联查询

Select 表1.$field1, 表1.$field2,……表2.$field1, 表2.$field2,…… from 表1,表2 where 表1.字段1 = 表2.字段2

