安装mysql 参考网上图解

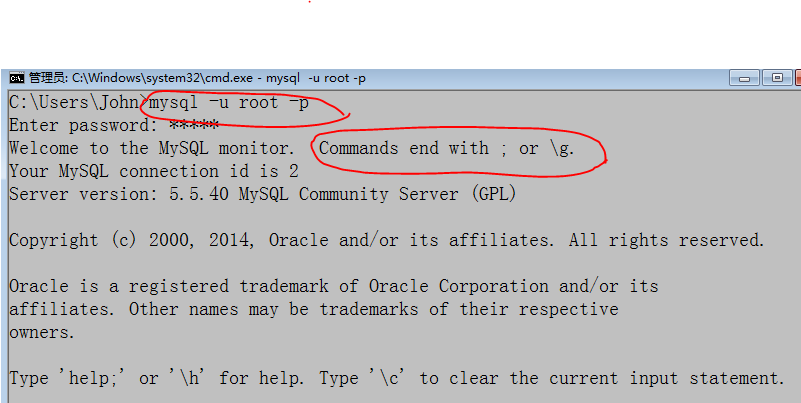
在cmd中进入mysql使用 mysql –u roo –p 加密码 命令

可以为： mysql –uroot –pmysql

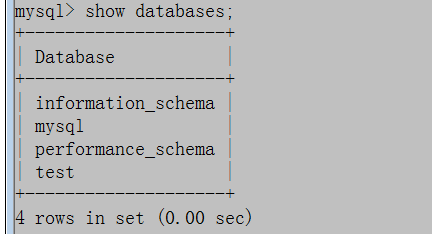
Mysql –u root –p 回车 密码

在mysql中， 每一条命令的结束都是使用“分号”

退出mysql 使用 “quit”命令



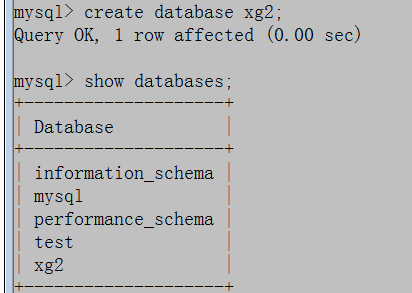
# 库

1. 查看所有的数据库
   1. show databases;
   2. 

2, 创建数据库

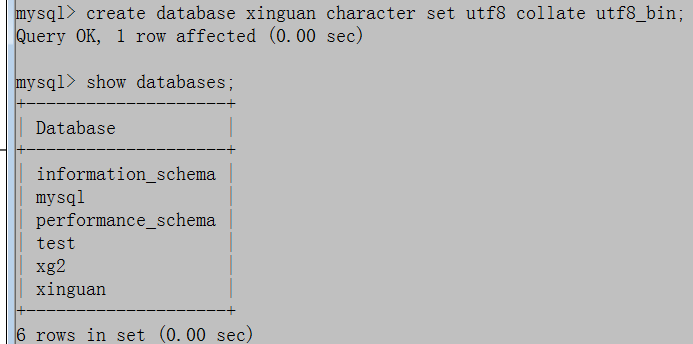
create database 数据库名称;

create database xg2;



|  |
| --- |
| CREATE {DATABASE | SCHEMA} [IF NOT EXISTS] *db\_name*      [*create\_specification* [, *create\_specification*] ...]    *create\_specification*:      [DEFAULT] CHARACTER SET *charset\_name*    | [DEFAULT] COLLATE *collation\_name* |

create database xinguan character set utf8 collate utf8\_bin;



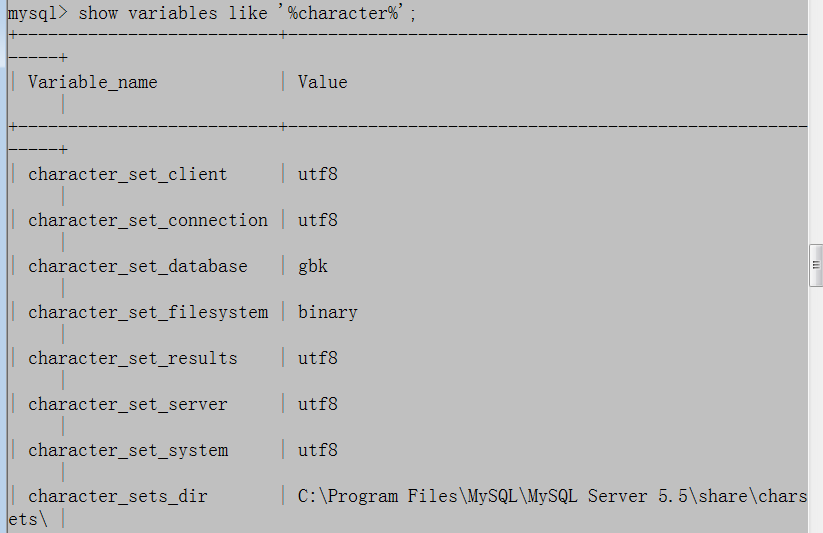
3, 修改数据库的字符集

alter database xinguan character set gbk;



查看数据库中的字符集

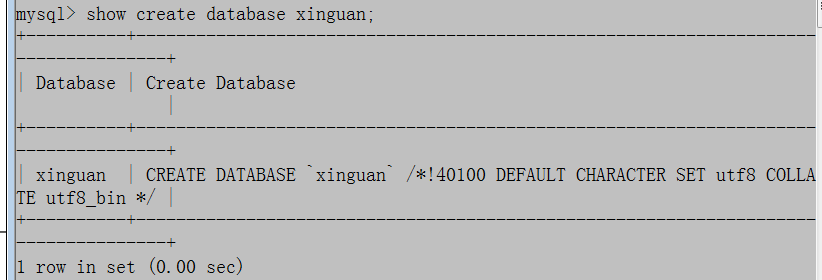
show variables like ‘character\_set\_database’;



4, 修改数据库的名称

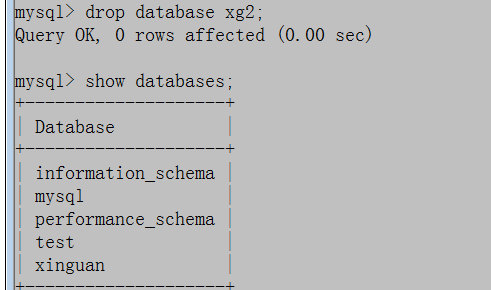
5, 查看数据库的建库语句

show create database 数据库名称;

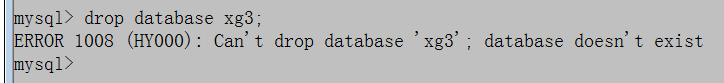


6, 删除数据库

drop database xg2;

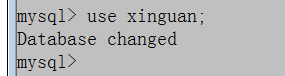


注意点:



7, 使用数据库

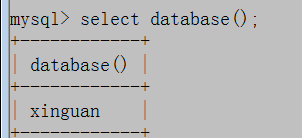
use 数据库名称;



# 表

首先要进入到数据库

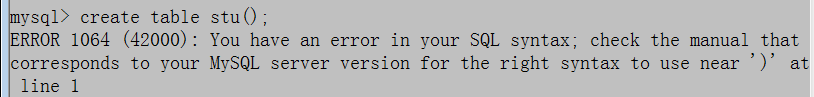
查看当前正在哪个数据库实例中.



1, 创建表

|  |
| --- |
| CREATE [TEMPORARY] TABLE [IF NOT EXISTS] *tbl\_name*      [(*create\_definition*,...)]      [*table\_options*] [*select\_statement*]  或：  CREATE [TEMPORARY] TABLE [IF NOT EXISTS] *tbl\_name*      [(] LIKE *old\_tbl\_name* [)];    *create\_definition*:  *column\_definition*    | [CONSTRAINT [*symbol*]] PRIMARY KEY [*index\_type*] (*index\_col\_name*,...)    | KEY [*index\_name*] [*index\_type*] (*index\_col\_name*,...)    | INDEX [*index\_name*] [*index\_type*] (*index\_col\_name*,...)    | [CONSTRAINT [*symbol*]] UNIQUE [INDEX]          [*index\_name*] [*index\_type*] (*index\_col\_name*,...)    | [FULLTEXT|SPATIAL] [INDEX] [*index\_name*] (*index\_col\_name*,...)    | [CONSTRAINT [*symbol*]] FOREIGN KEY          [*index\_name*] (*index\_col\_name*,...) [*reference\_definition*]    | CHECK (*expr*) |

create table stu();



create table stu(

id int primary key auto\_increment,

sname varchar(32),

gender bit default 0,

age int

);

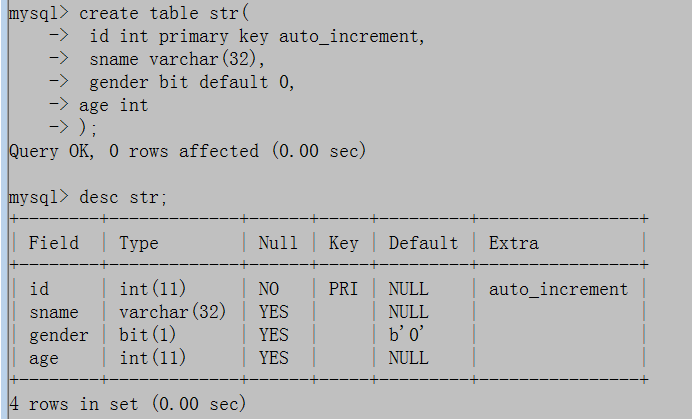
primary key 主键, 唯一, 并且非空

unique 唯一键

not null 非空键

auto\_increment : 自动增长

desc 表名称 查看表结构



拓展:

Java中的数据类型和 Mysql中数据类型的对应

在Java中数据类型可以分为: 基本数据类型和引用数据类型

基本数据类型包括:

byte short int long

float double

char

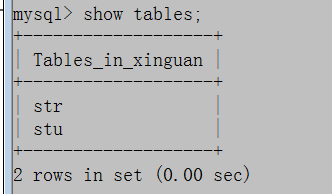
boolean

拓展: char类型和varchar类型和varchar2的区别

2, 查看表

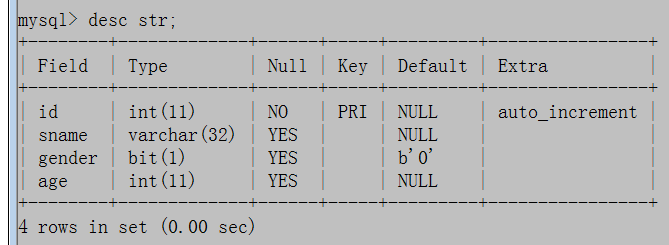
a, 查看一个数据库实例中所有的表

show tables;



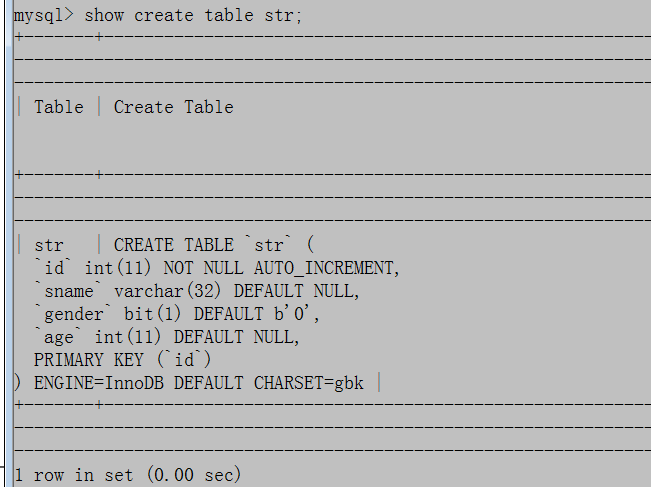
b, 查看表结构

desc 表名称;



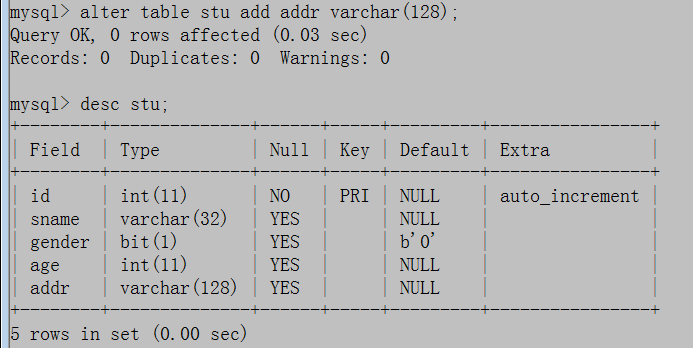
c, 查看建表语句

show create table 表名称;



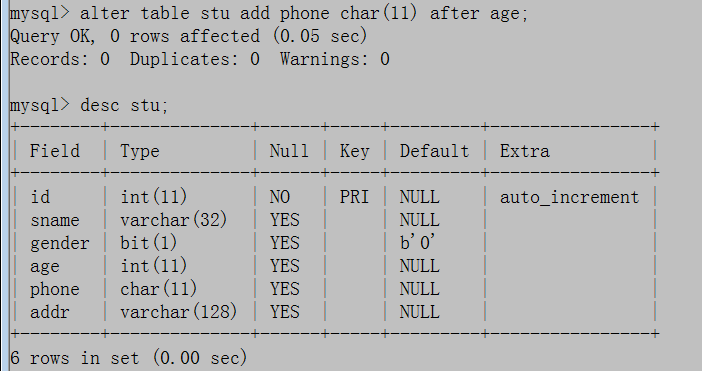
3, 修改表

a, 增加一个字段



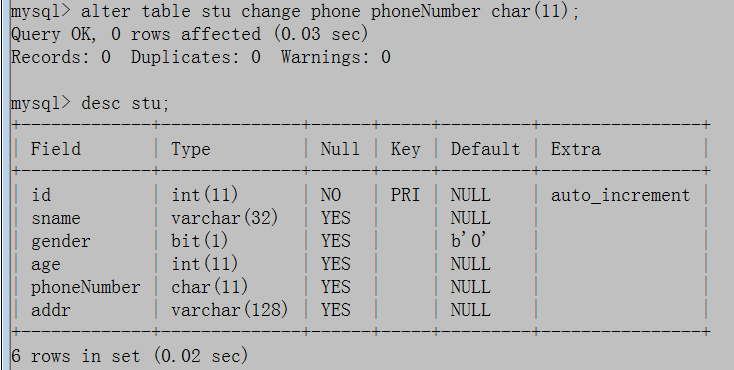
b, ;在指定的位置增加一个字段

alter table stu add phone char(11) after age;



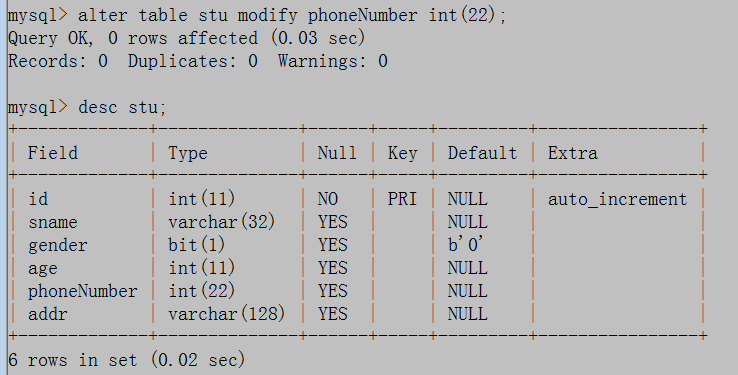
c, 修改字段的名称

alter table stu change phone phoneNumber char(11);



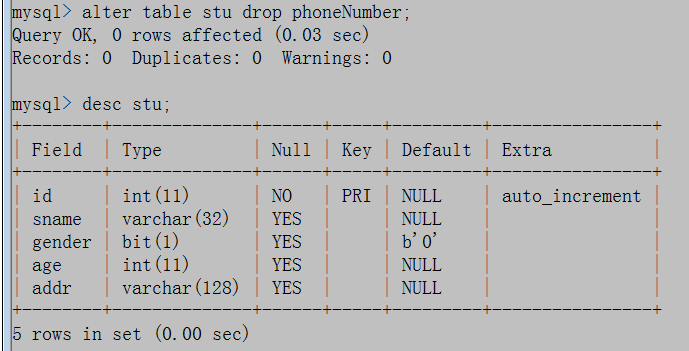
d, 修改字段的类型

alter table stu modify phoneNumber int(22);



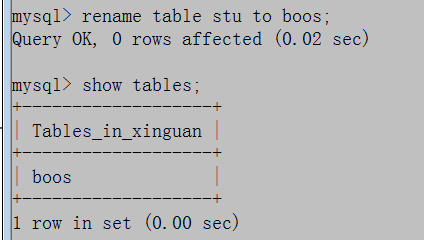
e, 删除某一个列

alter table stu drop phoneNumber;

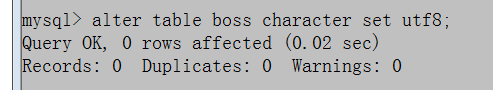


f, 修改表的名称

rename table stu to boos;



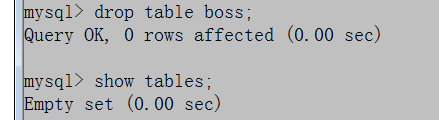
g, 修改表的字符集



拓展: alter, change, modify 区别???

h, 删除表

drop table 表名称;



拓展: drop, delete, truncate的区别???

# 数据

CRUD 是什么???

C ------------ create

R -------------- retrieve

U -------------- update

D ------------- delete

1, insert

|  |
| --- |
| INSERT [LOW\_PRIORITY | DELAYED | HIGH\_PRIORITY] [IGNORE]      [INTO] *tbl\_name* [(*col\_name*,...)]      VALUES ({*expr* | DEFAULT},...),(...),...      [ ON DUPLICATE KEY UPDATE *col\_name*=*expr*, ... ]  或：  INSERT [LOW\_PRIORITY | DELAYED | HIGH\_PRIORITY] [IGNORE]      [INTO] *tbl\_name*      SET *col\_name*={*expr* | DEFAULT}, ...      [ ON DUPLICATE KEY UPDATE *col\_name*=*expr*, ... ]  或：  INSERT [LOW\_PRIORITY | HIGH\_PRIORITY] [IGNORE]      [INTO] *tbl\_name* [(*col\_name*,...)]      SELECT ...      [ ON DUPLICATE KEY UPDATE *col\_name*=*expr*, ... ] |

create table stu(

id int primary key auto\_increment,

sname varchar(32) not null,

gender char(2),

age int

);

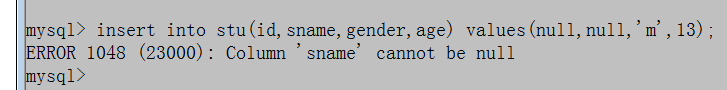
insert into stu(id,sname,gender,age) values(null,'siri','f',5);

insert into stu(id,sname,gender,age) values(null,'lily','f',12), (null,'lucy','f',12);

insert into stu(id,sname,gender,age) values(5,'tom','m',5);

insert into stu(id,sname,gender,age) values(null,'jack','m',13);

insert into stu(id,sname,gender,age) values(null,null,'m',13);



insert into stu(sname)values(‘rose’);

拓展: 如果写的数据超出了数据类型所表示的范围, 会怎么样???

拓展: 我们现在需要做大量的数据来做测试, 那么该怎么样在最短的时间内, 存入大量数据到表中.???

“蠕虫复制”:

insert into stu (sname, gender, age) select sname, gender, age from stu;

insert into stu(sname) values(‘李健铭’)

2, update

3, delete

4 select