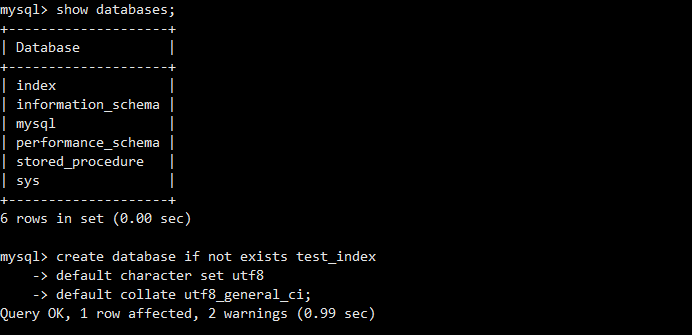
1. **创建 测试索引用MySQL 数据库，并指定字符集和校对规则**

使用 MySQL 命令行工具创建一个测试数据库，命名为 test\_db\_char，指定其默认字符集为 utf8，默认校对规则为 utf8\_general\_ci，输入的 SQL 语句与执行结果如下所示：

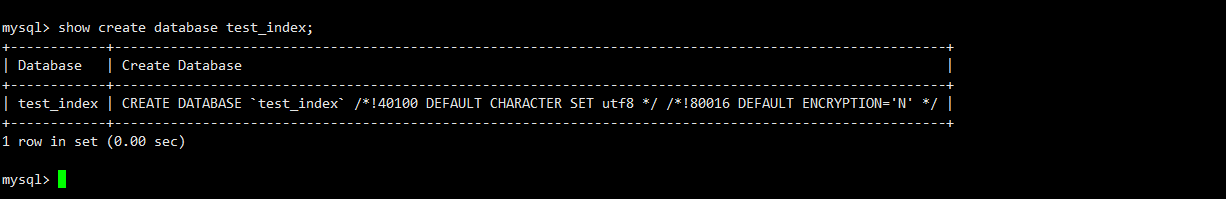
create database if not exists test\_index

default character set utf8

default collate utf8\_general\_ci;



这时，可以使用SHOW CREATE DATABASE查看刚刚创建的 test\_index 数据库的定义声明，发现该数据库的指定字符集为 utf8，运行结果如下所示：



2.查看表容量大小

select

table\_schema as '数据库',

table\_name as '表名',

table\_rows as '记录数',

truncate(data\_length/1024/1024, 2) as '数据容量(MB)',

truncate(index\_length/1024/1024, 2) as '索引容量(MB)'

from information\_schema.tables

where table\_schema='mysql'

order by data\_length desc, index\_length desc;

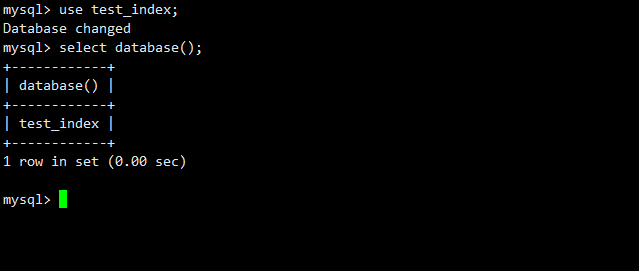
查看当前使用的是哪个数据库

select database();

当前登录的用户

mysql> select user();

1. 切换到刚刚创建的测试库：



并创建表：

create table user(

id int not null auto\_increment,

first\_name varchar(10)not null,

last\_name varchar(10)not null,

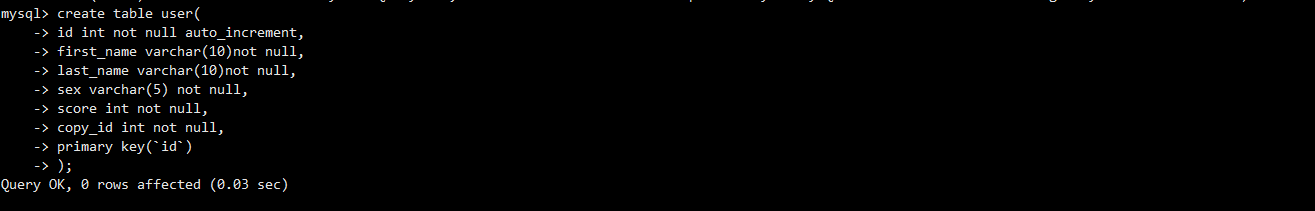
sex varchar(5) not null,

score int not null,

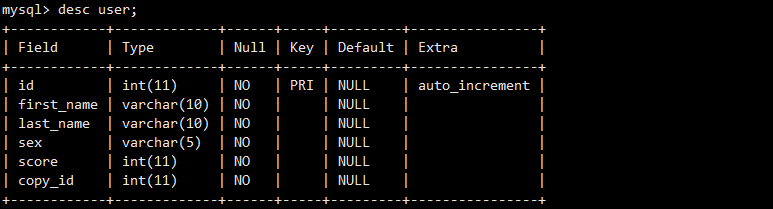
copy\_id int not null,

primary key(`id`)

);



查看刚刚创建的user表结构

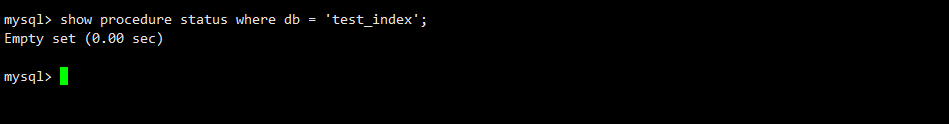


编写存储过程，创建存储过程随机生成一百万条测试数据：

查看测试库test\_index下存储过程成信息：

show procedure status where db = 'test\_index';

show function status where db = 'test\_index';（函数信息）



创建存储过程：

DROP PROCEDURE IF EXISTS add\_user;

DELIMITER //

create PROCEDURE add\_user(in num INT)

BEGIN

DECLARE rowid INT DEFAULT 0;

DECLARE firstname CHAR(1);

DECLARE lastname VARCHAR(3) DEFAULT '';

DECLARE name1 CHAR(1);

DECLARE name2 CHAR(1);

DECLARE sex CHAR(1);

DECLARE score CHAR(2);

SET @exedata = "";

WHILE rowid < num DO

SET firstname = SUBSTRING('赵钱孙李周吴郑王林杨柳刘孙陈江阮侯邹高彭徐',FLOOR(1+21\*RAND()),1);

SET name1 = SUBSTRING('一二三四五六七八九十甲乙丙丁静景京晶名明铭敏闵民军君俊骏天田甜兲恬益依成城诚立莉力黎励',ROUND(1+43\*RAND()),1);

SET name2 = SUBSTRING('一二三四五六七八九十甲乙丙丁静景京晶名明铭敏闵民军君俊骏天田甜兲恬益依成城诚立莉力黎励',ROUND(1+43\*RAND()),1);

SET sex=FLOOR(0 + (RAND() \* 2));

SET score= FLOOR(40 + (RAND() \*60));

SET rowid = rowid + 1;

IF ROUND(RAND())=0 THEN

SET lastname =name1;

END IF;

IF ROUND(RAND())=1 THEN

SET lastname = CONCAT(name1,name2);

END IF;

IF length(@exedata)>0 THEN

SET @exedata = CONCAT(@exedata,',');

END IF;

SET @exedata=concat(@exedata,"('",firstname,"','",lastname,"','",sex,"','",score,"','",rowid,"')");

IF rowid%1000=0

THEN

SET @exesql =concat("insert into user(first\_name,last\_name,sex,score,copy\_id) values ", @exedata);

prepare stmt from @exesql;

execute stmt;

DEALLOCATE prepare stmt;

SET @exedata = "";

END IF;

END WHILE;

IF length(@exedata)>0

THEN

SET @exesql =concat("insert into user(first\_name,last\_name,sex,score,copy\_id) values ", @exedata);

prepare stmt from @exesql;

execute stmt;

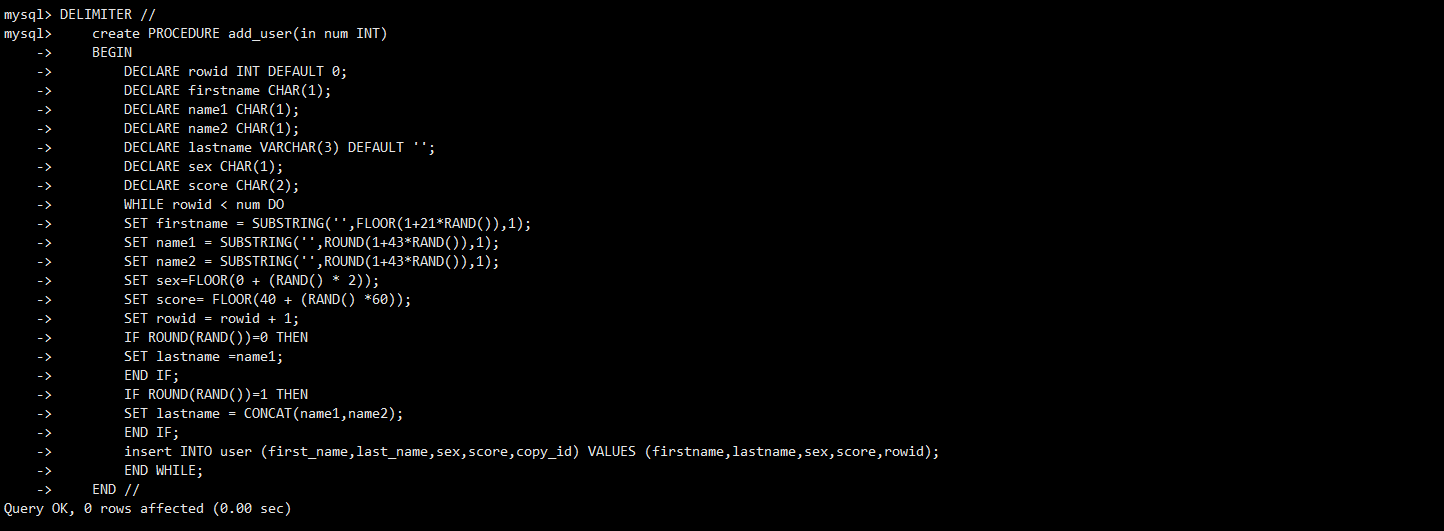
DEALLOCATE prepare stmt;

END IF;

END//

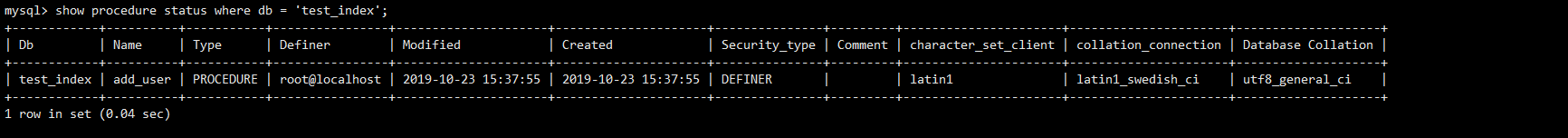
DELIMITER ;

直接粘贴到linux界面（全部粘贴），中文可能不显示，从而影响最后数据的生成。

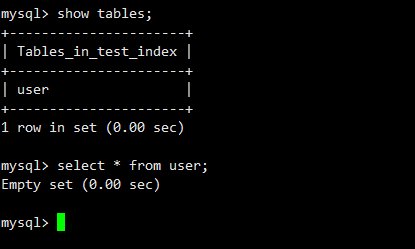


查看刚刚创建的存储过程信息：

show procedure status where db = 'test\_index';



现在调用存储过程添加随机用户数据，这里选择随即插入100W条记录：



call add\_user(1000000);

稍作等待，存储过程执行完毕后，这里我们查看是否成功插入100W条记录：

