**6.数据库实施**

**数据库的实施主要是根据逻辑结构设计和物理结构设计的结果，在计算机系统上建立实际的数据库结构、导入数据并进行程序的调试。它相当于软件工程中的代码编写和程序调试的阶段。**

**创建数据库主要通过SQL语句来实现，分别输入以下语句来执行：**

**创建了卡拉OK点歌系统的数据库：create database karaoke;**

1. **创建歌星表**

**create table 歌星表 ( 歌星编号 int primary key, 歌星名字 char(10) not null, 歌星性别 enum('男','女'), 国籍 char(20), 歌星类型 char(10), 拼音缩写 char(20) not null, 地区 char(20) );**

**insert into 歌星表 values('1','李伟','男','中国','流行','lw','大陆');**

**insert into 歌星表 values('2','周杰伦','男','中国','伟大','zjl','港台');**

**insert into 歌星表 values('3','许嵩','男','中国','摇滚','xs','港台');**

**insert into 歌星表 values('4','陈音','女','中国','流行','cy','港台');**

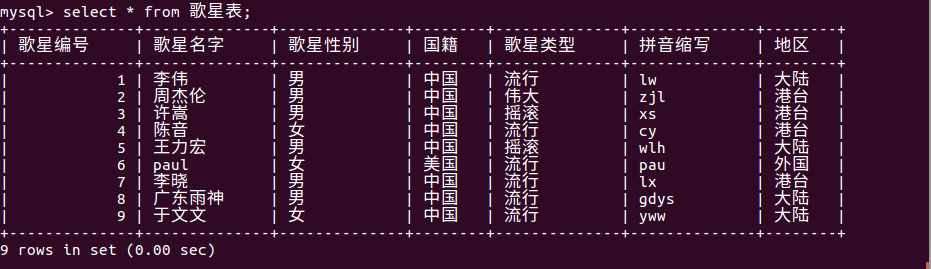
**insert into 歌星表 values('5','王力宏','男','中国','摇滚','wlh','大陆');**

**insert into 歌星表 values('6','paul','女','美国','流行','pau','外国');**

**insert into 歌星表 values('7','李晓','男','中国','流行','lx','港台');**

**insert into 歌星表 values('8','广东雨神','男','中国','流行','gdys','大陆');**

**insert into 歌星表 values('9','于文文','女','中国','流行','yww','大陆');**

****

1. **创建歌曲表**

**create table 歌曲表 ( 歌曲编号 int primary key, 歌曲名称 char(10) not null, 歌星编号 int not null, 语种 char(10), 歌曲类型 char(10), 路径 char(20), 歌曲拼音缩写 char(20) not null, 歌曲时长 datetime, foreign key (歌星编号) references 歌星表(歌星编号) );**

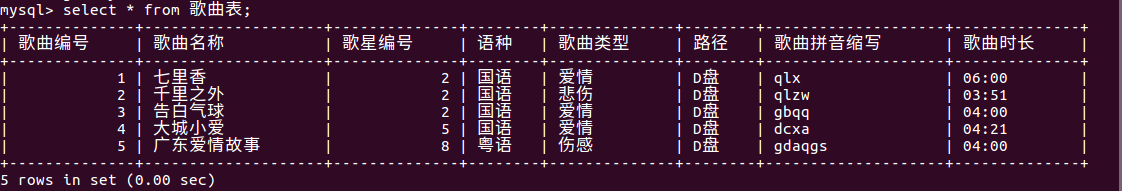
**insert into 歌曲表 values('1','七里香','2','国语','爱情','D盘','qlx','06:00');**

**insert into 歌曲表 values('2','千里之外','2','国语','悲伤','D盘','qlzw','03:51');**

**insert into 歌曲表 values('3','告白气球','2','国语','爱情','D盘','gbqq','04:00');**

**insert into 歌曲表 values('4','大城小爱','5','国语','爱情','D盘','dcxa','04:21');**

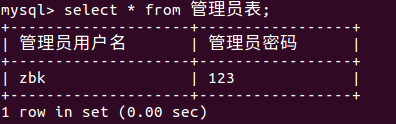
**insert into 歌曲表 values('5','广东爱情故事','8','粤语','伤感','D盘','gdaqgs','04:00');**



1. **创建管理员表**

**create table 管理员表 (管理员用户名 char(20) primary key, 管理员密码 char(20) not null);**

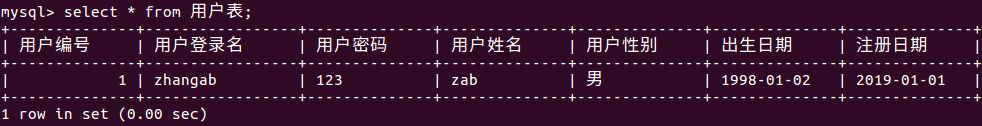
**insert into 管理员表 values ('zbk','123');**



1. **创建用户表**

**create table 用户表 ( 用户编号 int primary key, 用户登录名 char(20) not null unique, 用户密码 char(20) not null, 用户姓名 char(20) not null, 用户性别 enum('男','女'), 出生日期 date not null, 注册日期 date not null );**

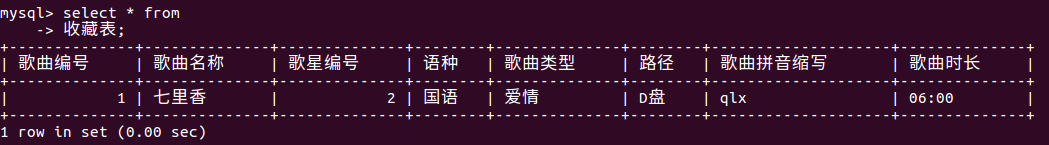
insert into 用户表 values(1, 'zhangab', '123', 'zab','男','1998-01-02', '2019-01-01');



1. **创建收藏表**

**create table 收藏表 ( 歌曲编号 int primary key, 歌曲名称 char(10) not null, 歌星编号 int not null, 语种 char(10), 歌曲类型 char(10), 路径 char(20), 歌曲拼音缩写 char(20) not null, 歌曲时长 char(20) , foreign key (歌星编号) references 歌星表(歌星编号) );**

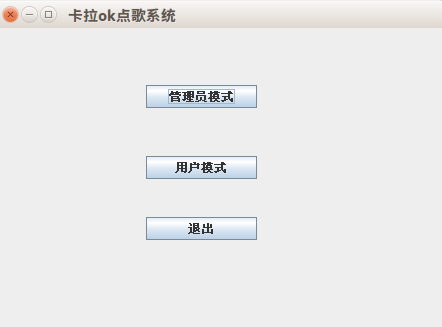
mysql> insert into 收藏表 values (1, '七里香',2, '国语','爱情','D盘','qlx', '06:00');



**7.数据库运行与维护**

**7.1系统功能测试**

**7.1.1 登录界面测试**



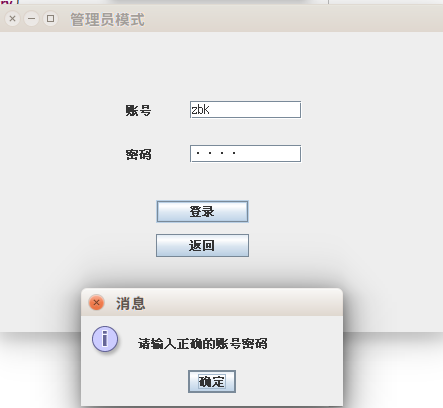
**1管理员登录界面**



**2用户登录界面**



**3登录名错误或者密码错误**

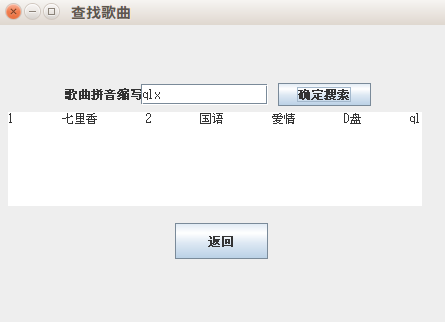


**7.1.2点歌测试**

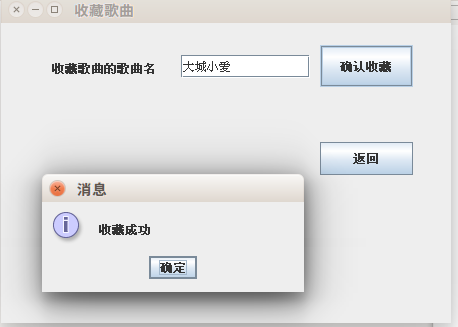
**1拼音点歌**

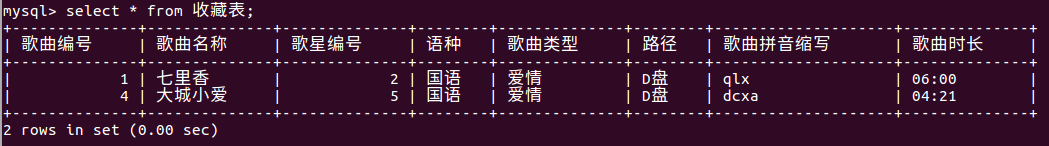






**2收藏歌曲**





**7.2后台管理测试**

**本系统规定后台管理员只有一个，帐号zbk密码123下面开始测试：**

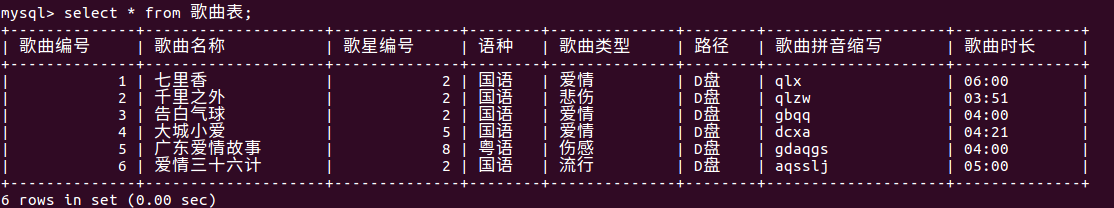
**7.2.1添加歌曲**



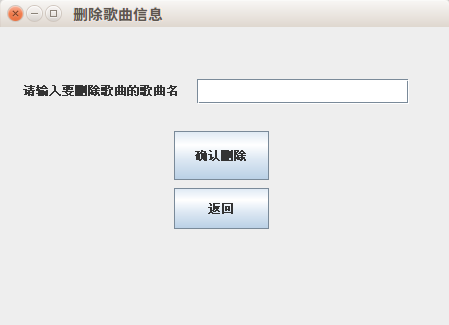






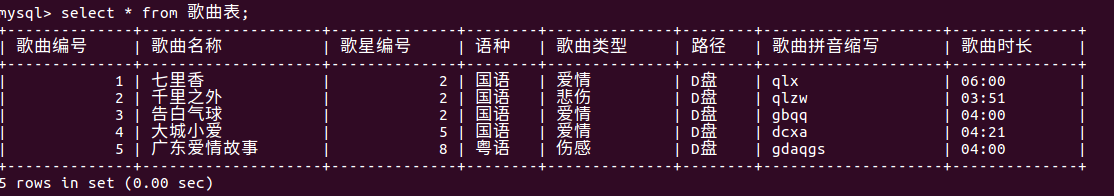


**7.2.2删除歌曲**



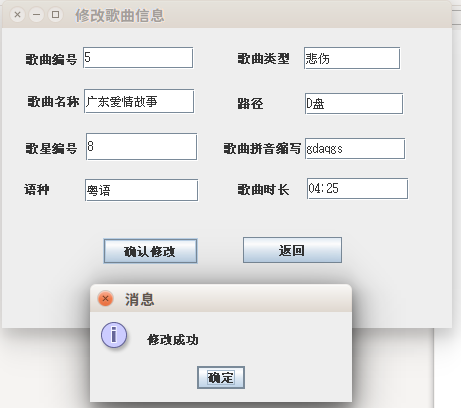


**查看是否从表中删除**

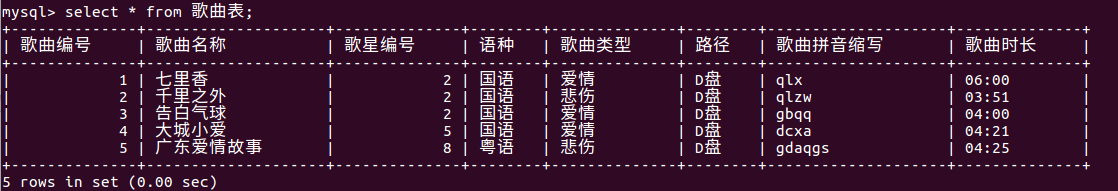
**歌曲已经从表中删除**

**7.2.3修改歌曲信息**

**修改广东爱情故事的歌曲时长和类型**



**查看是否修改成功**



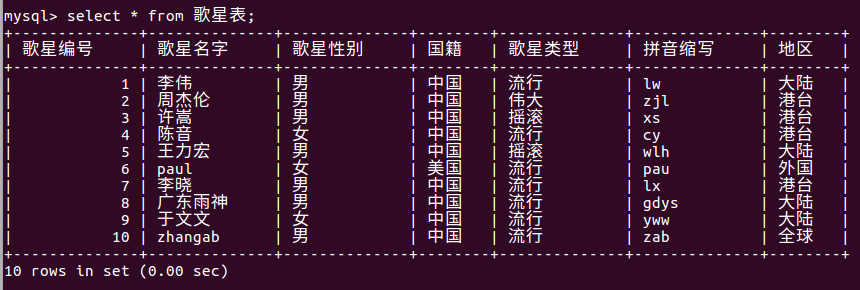
**歌曲类型和歌曲时长的属性修改成功**

**7.2.4添加歌星**





**查看是否添加成功**

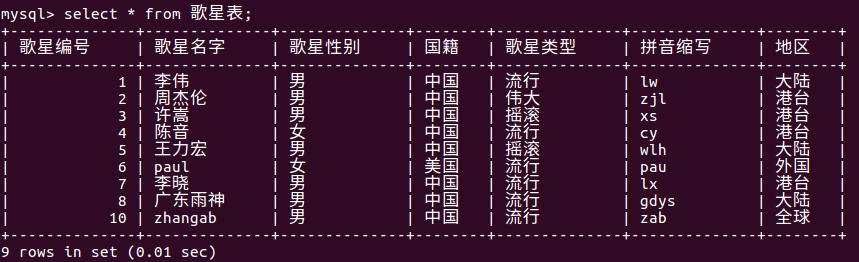
****

**已经添加成功**

**7.2.5删除歌星**



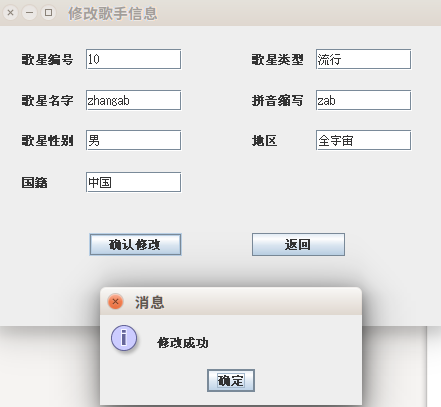
**查看是否删除成功**



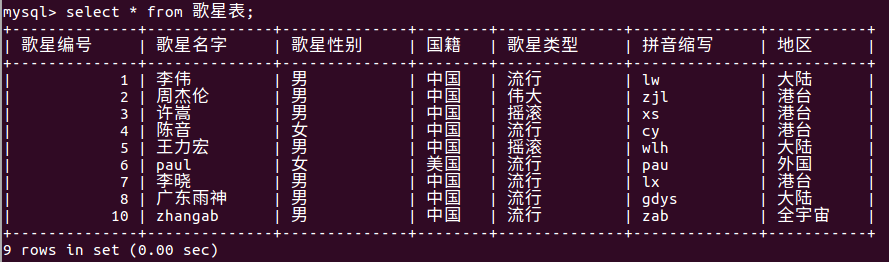
**已经删除成功.**

**7.2.6修改歌星信息**

**把zhangab的地区修改为全宇宙**

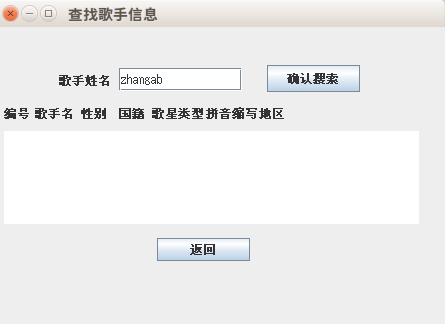


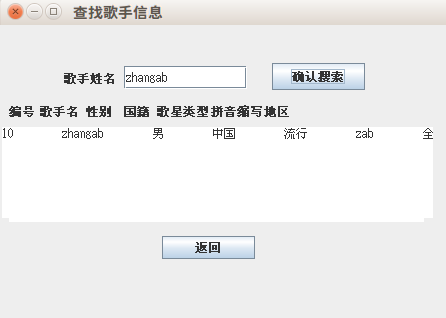
**查看是否修改成功**



**修改 成功**

**7.2.7查找歌星信息**





**7.2.8添加用户**



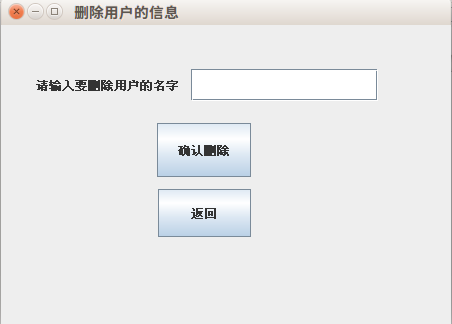


**查看是否添加成功**



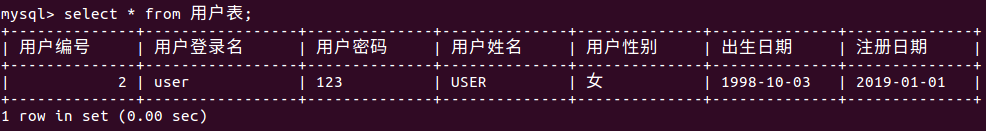
**已经添加成功**

**7.2.9删除用户**





**查看是否删除成功**

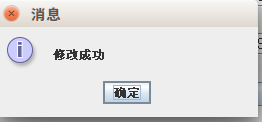


**已经从用户表中删除.**

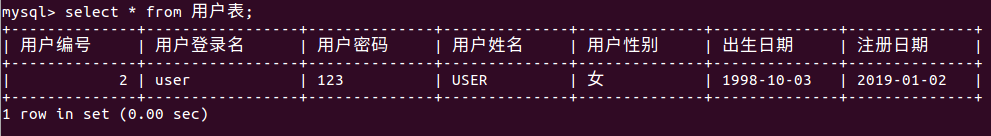
**7.2.10修改用户信息**

**修改user的注册日期为2019-1-2**



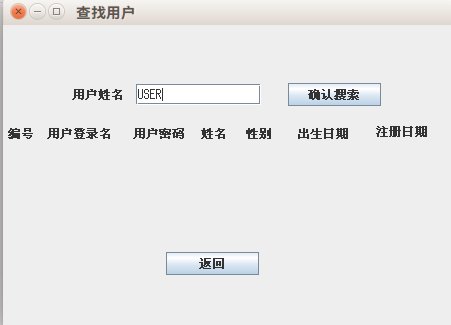


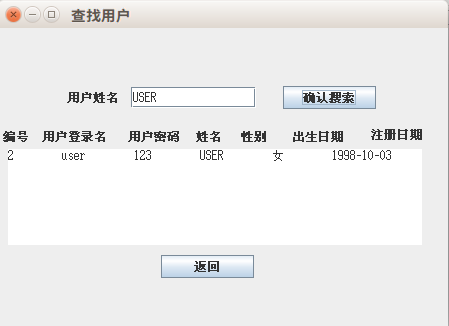
**查看用户信息是否修改**



**修改成功**

**7.2.11查找用户信息**





**7.2.12其他功能**

**功能可以在下面界面中对应进行使用**



**7.3数据库的维护**

**管理员可登录后台管理系统对歌曲、歌星、用户进行管理和维护。进入后台管理系统后，显示的是歌曲管理、歌星管理、用户管理大三模块。主要模块功能如下：**

**歌曲管理：管理员可根据歌曲信息添加歌曲，删除歌曲，查询歌曲。**

**歌星管理：管理员可根据歌星信息添加歌星，删除歌星，查询歌星。    用户管理管理员可根据用户信息添加用户，删除用户，查询用户**