Foreign Key:

+ ----- +

| |

+ ----- +

| create table Listens (userName varchar(50), songName varchar(50), numListens int, Primary Key (userName, songName), Foreign Key (songName) references Song(name) ) ; |

+ ----- +

1 rows

+ ----- +

| |

+ ----- +

| update Listens set songName='Kiss of Life2600' where songName='Kiss of Life26'; |

+ ----- +

1 rows



Procedure:

drop procedure if exists times\_listened;

/\*

Procedure times\_listened(userName, songName) takes in a user's name and a song name and returns the number of times that

the specific user has listened to the specified song

\*/

delimiter //

create procedure times\_listened(uname varchar(50), sname varchar(50))

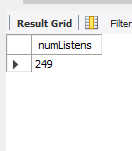
begin

select L.numListens from Listens L where L.userName=uname and L.songName=sname;

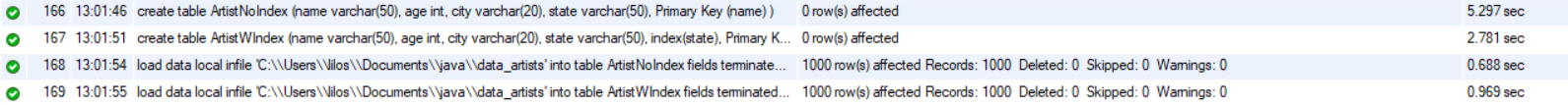
end //

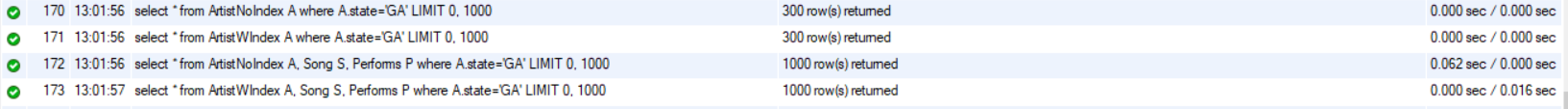
delimiter ;

call times\_listened('Oscar835', 'One Beer6');



Index:





All of code:

set global local\_infile=1;

show databases;

drop database if exists musicAppDB;

create database musicAppDB;

show databases;

use musicAppDB;

drop table if exists Song;

drop table if exists Users;

drop table if exists Artist;

drop table if exists Producer;

drop table if exists Engineer;

drop table if exists MusicLabel;

drop table if exists Listens;

drop table if exists Performs;

drop table if exists Makes;

drop table if exists MixesAndMasters;

drop table if exists Sources;

create table Users (name varchar(50), age int, city varchar(50), state varchar(50), dateJoined date, Primary Key (name) ) ;

create table Song (name varchar(50), genre varchar(50), dateReleased date, isExplicit boolean, numListens long, Primary Key (name) ) ;

create table Artist (name varchar(50), age int, city varchar(20), state varchar(50), Primary Key (name) ) ;

create table Producer (name varchar(50), age int, city varchar(50), state varchar(50), Primary Key (name) ) ;

create table Engineer (name varchar(50), age int, city varchar(50), state varchar(50), Primary Key (name) ) ;

create table MusicLabel (name varchar(50), city varchar(50), state varchar(50), Primary Key (name) ) ;

select 'create table Listens (userName varchar(50), songName varchar(50), numListens int, Primary Key (userName, songName), Foreign Key (songName) references Song(name) ) ;' as '';

create table Listens (userName varchar(50), songName varchar(50), numListens int, Primary Key (userName, songName), Foreign Key (songName) references Song(name) ) ;

create table Performs (artistName varchar(50), songName varchar(50), numArtists int, Primary Key (artistName, songName), Foreign Key (songName) references Song(name)) ;

create table Makes (producerName varchar(50), songName varchar(50), numProducers int, Primary Key (producerName, songName), Foreign Key (songName) references Song(name) ) ;

create table MixesAndMasters (engineerName varchar(50), songName varchar(50), numEngineers int, Primary Key (engineerName, songName), Foreign Key (songName) references Song(name) ) ;

create table Sources (labelName varchar(50), songName varchar(50), Primary Key (labelName, songName), Foreign Key (songName) references Song(name) ) ;

load data local infile 'O:\\School work\\comp sci\\databases\\data\_users' into table Users fields terminated by ',' lines terminated by '\n';

load data local infile 'O:\\School work\\comp sci\\databases\\data\_songs' into table Song fields terminated by ',' lines terminated by '\n';

load data local infile 'O:\\School work\\comp sci\\databases\\data\_artists' into table Artist fields terminated by ',' lines terminated by '\n';

load data local infile 'O:\\School work\\comp sci\\databases\\data\_producers' into table Producer fields terminated by ',' lines terminated by '\n';

load data local infile 'O:\\School work\\comp sci\\databases\\data\_producers' into table Engineer fields terminated by ',' lines terminated by '\n';

load data local infile 'O:\\School work\\comp sci\\databases\\data\_labels' into table MusicLabel fields terminated by ',' lines terminated by '\n';

load data local infile 'O:\\School work\\comp sci\\databases\\data\_listens' into table Listens fields terminated by ',' lines terminated by '\n';

load data local infile 'O:\\School work\\comp sci\\databases\\data\_performs' into table Performs fields terminated by ',' lines terminated by '\n';

load data local infile 'O:\\School work\\comp sci\\databases\\data\_makes' into table Makes fields terminated by ',' lines terminated by '\n';

load data local infile 'O:\\School work\\comp sci\\databases\\data\_mixes\_and\_masters' into table MixesAndMasters fields terminated by ',' lines terminated by '\n';

load data local infile 'O:\\School work\\comp sci\\databases\\data\_sources' into table Sources fields terminated by ',' lines terminated by '\n';

select 'update Listens set songName=''Kiss of Life2600'' where songName=''Kiss of Life26'';' as '';

update Listens set songName='Kiss of Life2600' where songName='Kiss of Life26';

drop procedure if exists times\_listened;

/\*

Procedure times\_listened(userName, songName) takes in a user's name and a song name and returns the number of times that

the specific user has listened to the specified song

\*/

delimiter //

create procedure times\_listened(uname varchar(50), sname varchar(50))

begin

select L.numListens from Listens L where L.userName=uname and L.songName=sname;

end //

delimiter ;

call times\_listened('Oscar835', 'One Beer6');

create table ArtistNoIndex (name varchar(50), age int, city varchar(20), state varchar(50), Primary Key (name) ) ;

create table ArtistWIndex (name varchar(50), age int, city varchar(20), state varchar(50), index(state), Primary Key (name) ) ;

load data local infile 'O:\\School work\\comp sci\\databases\\data\_artists' into table ArtistNoIndex fields terminated by ',' lines terminated by '\n';

load data local infile 'O:\\School work\\comp sci\\databases\\data\_artists' into table ArtistWIndex fields terminated by ',' lines terminated by '\n';

select \* from ArtistNoIndex A where A.state='GA';

select \* from ArtistWIndex A where A.state='GA';

select \* from ArtistNoIndex A, Song S, Performs P where A.state='GA';

select \* from ArtistWIndex A, Song S, Performs P where A.state='GA';