Scripts – add, update, search CMsc495

The first three examples of SQL code are used to insert new records into the database. All items listed are required and must be used. If you information about the movie is unknown there must be a blank line in the space where the information would go. That will be inserted using single quotes right next to each other(i.e. ''). All tables have an autonumber field that is not listed and that will be filled with the next sequential number.

The movie table requires the following fields:

* title
* type
* rating
* length
* year\_released
* director

The example to insert a new field into the movie database is:

insert into movie (title, type, rating, length, year\_released, director) values ('Life of Pi', 'Adventrue','PG','127 min', '2012', 'not sure');

The dvd\_video table requires the following fields:

* movie\_id
* serial\_no
* late\_fee

The example to insert a new field into the movie database is:

insert into dvd\_video (movie\_id, serial\_no, late\_fee) values ('1', '1D1 ', '2.5');

The customer table requires the following fields:

* last\_name
* first\_name
* street\_address
* city
* state
* zip\_code
* phone
* hold\_on\_account

The hold\_on\_account is a true/false variable. MySQL uses a 1 or a 0 for those functions. Standard binary code uses 1 for on and 0 for off, the database follows this method.

The example to insert a new customer is:

insert into customer (last\_name, first\_name, street\_address, city, state, zip\_code, phone, hold\_on\_account) values ('Jones', 'James','123 J Street','Fallbrook','CA','92028','123456789','1');

The rental table requires the following fields:

* cust\_id
* date\_out
* movie\_id
* late\_fees\_paid

The example to insert a new rental is:

insert into rental (cust\_id, date\_out, movie\_id, late\_fees\_paid) values ('1', date '2013-09-01','1','5.25');

The next scripts are used to search the database. The Interface control 2.4.1 requirements state the user must insert the required information in the search page; however there would be no reason to search if that were the case. The variable is the place holder for the data from the GUI.

* Search for customer using customer id
  + select last\_name, first\_name from customer where cust\_id = 'variable';
* Search for customer id using last name
  + select cust\_id, last\_name, first\_name, phone from customer where last\_name = 'variable';
* Search for customer id using phone number
  + Select cust\_id, last\_name, first\_name from customer where phone = 'variable';

Searching for last name may have to many results if the name is a common last name.

The generic search function that will return all information and all records from a table:

* select \* from movie;
* select \* from customer;
* select \* from rental;
* select \* from dvd\_video;

The following items are used to search the movie table using one field. This is used if a customer wants all “R” rated movies, or all “adventure” movies or all movies by a specific director.

* select movie\_id, title from movie where type = 'variable';
* select movie\_id, title from movie where rating = 'variable';
* select movie\_id, title from movie where length = 'variable';
* select movie\_id, title from movie where year\_released = 'variable';
* select movie\_id, title from movie where director = 'variable';

The following scripts are used to update information on an already existing record.

* Customer table
  + Update last name
    - update customer set last\_name = 'new value' where cust\_id = 'variable';
  + Update phone number
    - update customer set phone = 'new value' where cust\_id = 'variable';
  + Update address
    - update customer set street\_address = 'new value' where cust\_id = 'variable';
* Movie
  + Update title
    - update movie set title = 'new value' where movie\_id = 'variable';
  + Update director
    - update movie set director = 'new value' where movie\_id = 'variable';
* Specific copy of movie
  + Update fee
    - update dvd\_video set late\_fee = 'new value' where item\_id = 'variable';
* Rental
  + Updating the late fees to show that the customer has paid.
    - update rental set late\_fees\_paid = 'new value' where rental\_id = 'variable';

The following scripts are used to permanently remove records from a table.

* Remove a customer
  + delete from customer where cust\_id ='variable';
* Remove a specific copy of a movie (one dvd)
  + delete from dvd\_video where item\_id ='variable';
* Remove all copies of a movie
  + delete from movie where movie\_id ='variable';