CMSC214 Project 12

Derya Ozdemir Kurin

Concepts tested in this program:

* JavaFX Application
* Database Programming Using Java

**\*34.1 Accessing and updating a Staff table**

Write a JavaFX GUI program that views, inserts, and updates staff information stored in a database, as shown in the following figure. The view button displays a record with a specified ID. The Insert button inserts a new record. The Staff table is created as follows:

**create table** Staff (

id **char**(9) **not null**,

lastName **varchar**(15),

firstName **varchar**(15),

mi **char**(1),

address **varchar**(20),

city **varchar**(20),Staff In

state **char**(2),

telephone **char**(10),

email **varchar**(40),

**primary key** (id)

);

**Set up for the project:**

**Enter the root of mysql with the root user password**

**use mysql;**

**Create new user as dozdemir with a password: staff**

grant select, insert, update, delete, create, create view, drop, execute, references on \*.\* to 'dozdemir'@'localhost';

**Grant privileges to the user, in order for it to use necessary sql queries to run the program:**grant select, insert, update, delete, create, create view, drop, execute, references on \*.\* to 'dozdemir'@'localhost';

**Exit.**

**Enter the root with the new credentials:**

./mysql -u dozdemir -p

PASSWORD: staff

**Create a new database for the project:**

create database project12;

Run the script below to create a table “Staff” with the given columns:

**create table** Staff (

id **char**(9) **not null**,

lastName **varchar**(15),

firstName **varchar**(15),

mi **char**(1),

address **varchar**(20),

city **varchar**(20),Staff In

state **char**(2),

telephone **char**(10),

email **varchar**(40),

**primary key** (id)

);

**Note: In case the new tab character causes problem, the script below works well:**

**create table Staff (id char(9) not null, lastName varchar(15), firstName varchar(15), mi char(1), address varchar(20), city varchar(20), state char(2), telephone char(10), email varchar(40), primary key (id));**

Insert some rows into Staff table:

insert into Staff values ('1', 'Zellie', 'Marc', 'M', '34 Mannalee St', 'College Park', 'MD', '2348594855', 'marc@gmail.com');

insert into Staff values ('2', 'Smith', 'John', 'F', '64 Mannalee St', 'College Park', 'MD', '2344942747', 'smith@gmail.com');

insert into Staff values ('3', 'Johnson', 'Steven', 'J', '45 Mannalee St', 'College Park', 'MD', '2344546373', 'johnson@gmail.com');

**Test Cases:**

**View Button:**

When the id is 1:

Graphical user interface, application

Description automatically generated

**Insert Button:**

**First entered the new staff data, the message status at the bottom is “Row not found”**

**Graphical user interface

Description automatically generated**

**After clicking insert button:**

**Message Status; row affected 1**

**Insert Complete**

**View button confirms the change**

**Graphical user interface

Description automatically generated**

**Update Button:**

**Changing the address of the staff and also the middle initial**

**Graphical user interface

Description automatically generated**

**Clicking update button makes the changes, Graphical user interface, application

Description automatically generated**

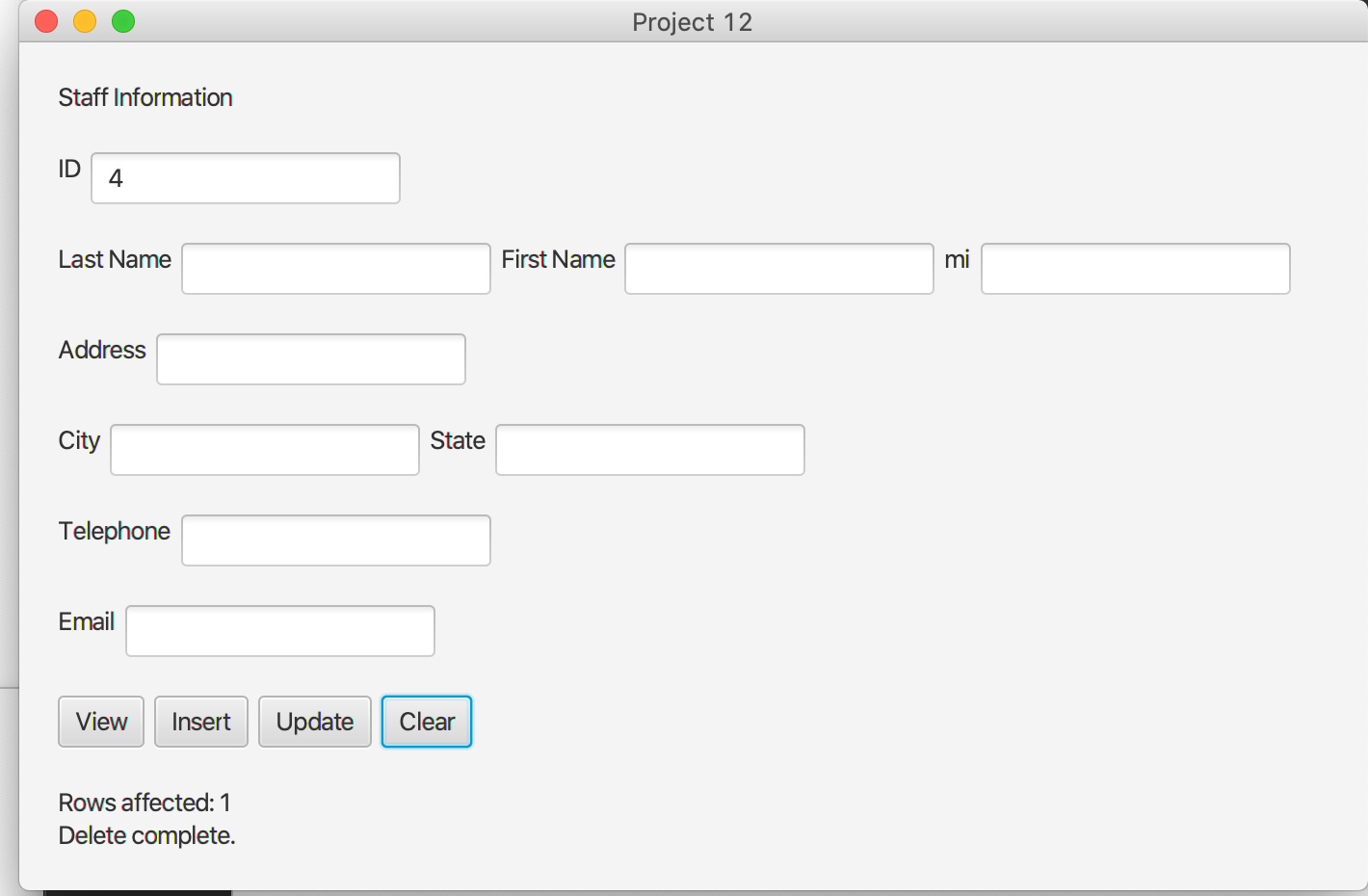
**Message status: Update Complete**

**Clicking view button confirms this update.**

**Clear Button:**

**The staff with id 4 is deleted with delete button**

**The message status: Delete Completed**

****

**Clicking view button for the staff with id finds no rows,**

**Message status: Row not found**

**A picture containing diagram

Description automatically generated**

**Lessons Learned:**

This chapter gave a good summary of how to connect to a relational database system with a java program. The book did not cover the sql queries for update, delete or insert but on the internet, there is enough tutorials which made the hardest part much easier.

I learned that in order to connect to a database system, I need to install that dbms ion my local machine. Also I needed to add a mysql-connector jar file into the project, otherwise the project does not connect to the db.

I learned how to create a user with a password in mysql, also giving the user permissions to use necessary sql queries.

Only after that a user can create databases, tables and perform on these tables.

I practiced the interpolation with variables on sql queries, it is a tedious job, because even a single ‘ mark can be a bug in the program. Also I learned that while select statements return the entire response string, other queries update, insert, delete returns an integer for the number of rows affected. I used that number in an if/else statement to see if the query was successful.

Overall, this project and the set up process gave me a good insight on how java interacts with database systems. I had experience in sql before but hadn’t used it with java. So I found this chapter very useful.

|  |  |  |  |
| --- | --- | --- | --- |
| **#** |  | **Y/N** | **Comments** |
|  | **Source java files** | **Y** |  |
|  | **Word Document** | **Y** |  |
|  |  |  |  |
|  |  |  |  |
|  | **Program compiles** | **Y** |  |
|  | **Program runs** | **Y** |  |
|  | **Checklist is completed and included in the Documentation** | **Y** |  |
|  | **Documentation file:** |  |  |
|  | **Comprehensive Test Plan** | **Y** |  |
|  | **Screenshots based on Test Plan** | **Y** |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  | **Lessons Learned** | **Y** |  |