**ITC 5201 Database Programming using Java**

**Assignment #4**

**Description:** Write a Java 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 Staff table is created as follows:

**create t**

**able** 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)

);

Graphical user interface, text, application

Description automatically generated

I have created 4 different files in my code as follows: -

1. StaffGUI.java is for the designing part of the file.
2. OracleDatabase.java is for all the validation of fields and database queries.
3. DatabaseInfor.java is for information on database and driver manager classification.
4. Main.java which has the main method.

This is the look of my code result, I have used gride layout and 3 different panels for creating GUI.

Graphical user interface, application

Description automatically generated

In this data, there is no error which is why I’m able to insert data into the database without any problem. Which we can confirm in MySQL.

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Here, I’m trying to search for a non-existing Id, which is why we are getting the error message.

Graphical user interface, text, application

Description automatically generated

Here, I’m entering fewer digits than 9 which is why we are having an error message.

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generatedHere is View Button is working perfectly fine.

`

Here is my update functionality that works well, and the confirmation for that is the output of MySQL.

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Validation for email id.

Graphical user interface, text, application

Description automatically generated

Validation for Telephone number.

Graphical user interface, text, application, chat or text message

Description automatically generated

Validation for State field.

Graphical user interface, application

Description automatically generated

Validation for MI field.

**StaffGUI.Java**

Graphical user interface

Description automatically generated with low confidence

A picture containing table

Description automatically generated

Application

Description automatically generated with low confidence

Text

Description automatically generatedText

Description automatically generated

Text

Description automatically generated

andGraphical user interface, text, application

Description automatically generated

Text

Description automatically generated

In This File, I have code for GUI Design like labels and text fields, and buttons. And some methods for creating the functionality of those buttons. In this functionality, I’m checking for the validation of the fields and defining error messages when needed. I have one extra function to check the valid email address which I’m defining the pattern for a valid email address and implementing separately and then using in the validation of the statement.

**OracleDatabase.Java**

Graphical user interface, text, application, email

Description automatically generated

In this file, I have coded for the making connection between MySQL and java, and then after code for SQL queries to do operations when we are using different buttons on GUI.

We have to follow the first 3 steps to establish a connection with MySQL.

Here, First, we are giving the path of the driver with the help of the following statement.

Class.*forName*("oracle.jdbc.driver.OracleDriver");

System.***out***.println("Drivers are Loaded"); //message

Then after, We are creating an object of a Driver Manager class as conn and giving all the required information to get access of the MySQL from eclipse.

*Conn*=DriverManager.*getConnection*("jdbc:oracle:thin:@calvin.humber.ca:1521:grok","n01496176","oracle");

System.***out***.println("Database is Connected"); //message

Then, the final step, create statement for establishing connection.

*stat* = *conn*.createStatement();

Graphical user interface, text, application, email

Description automatically generated

In these statements, we are comparing or checking for duplicate IDs or does the user input is matching the criteria of our condition or not each time when a user enters in GUI. For that, we are fetching data from the database with a SELECT statement.

Graphical user interface, text, application, email

Description automatically generated

For performing the Insert statement, We are again checking the id first if that id is not in the database then we are performing Insert queries to Insert all the data into the database with the prepared Statement and execute the query with executeUpdate();

Graphical user interface, text, application, email

Description automatically generated

This code of block is for view statement, we are checking for the existence of id in the database and returning information of that particular id with related fields with help of this statement [ String selectRow = "SELECT \* FROM staff WHERE id= " + id;], and updating row after the user changes any values in the field.

Graphical user interface, text, application, email

Description automatically generated

Text

Description automatically generated with low confidence

This code is for the Update query, in this, we are updating not null values from the text field and updating the string values using the substring() function. And execute the statement with prepare statement.

**DatabaseInfo.Java**Graphical user interface, text, application, email

Description automatically generated

Here is the code for the Database Info file, In this, I’m making the connection with the database and having essential information which will be required to get connected to the database.

**Main.Java**

**Graphical user interface, text, application

Description automatically generated**

This file contains the main class for the program.