CIS611 – Spring 2017
Individual Practice Programming Assignment: PA08

Due: Friday Apr 7, 2017 11:59pm

Total Points: 20

Java Client-Server Database Programming

Q1 (20 points)**:**

This assignment has two parts:

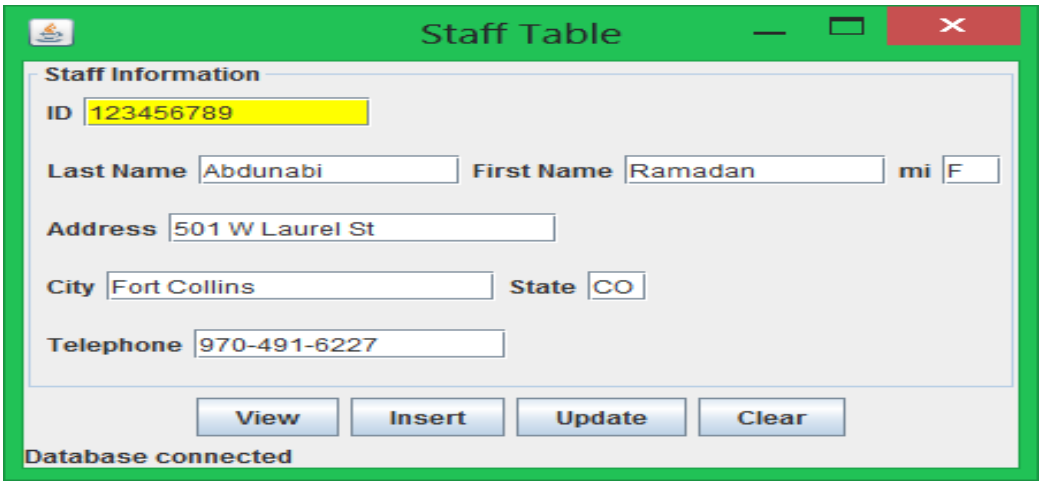
Part I: (Use your individual database for this assignment, review the MYSQL DATABASE CONNECTIVITY documents posted on Canvas under the Resources page)
~~*localhost* MySQL Database installation and instantiation.~~

- ~~Download WAMP Server (http://www.wampserver.com/) for Windows, or LAMP Server (https://bitnami.com/stack/lamp/installer) for Linux, in order to create a *localhost* database server in your personal machine. You can create/manage your database and tables by opening the phpMyAdmin page from W/LAMP server icon in IE or Chrome browser. Alternatively, download a free MySQL Workbench visual tool (https://www.mysql.com/products/workbench/) to create your database and tables. MySQL Workbench is a visual tool that provides data modeling, SQL development, and comprehensive administration tools for server configuration, user administration, and much more. MySQL Workbench is available on Windows, Linux and Mac OS X.~~
- ~~For using a free MySQL Workbench visual tool- for database architects, developers, and DBAs:~~
    - ~~Use the following tutorial link to creating a new MySQL Connection: https://dev.mysql.com/doc/workbench/en/wb-getting-started-tutorial-create-connection.html~~
    - ~~Use the following tutorial link to create a new scheme model (your assignment database and tables, you must name your scheme model (in your local MySQL database as **"MyDatabase"**): https://dev.mysql.com/doc/workbench/en/wb-getting-started-tutorial-creating-a-model.html~~

Part II:
Develop a Java Client/Server application to manipulate data in your MySQL database. The program applies the 3-tier database application architecture. With the 3-tier model, the client interacts with the database server via the server application. The Java client program displays Swing GUI with four buttons to perform view, insert, update, and delete Staff information stored in the MySQL database named **"YourDatabase".** The client GUI layout is presented below.

The view button displays a Staff record based on the Staff ID specified in the GUI textfield component. A correct input format must be validated such as for date. You also need to validate

required inputs such as for the insert and update operations; length of inputs must also be checked. Once a user clicks on a button, the client program sends the Staff data along with the operation type to the server as a serializable message object. User input format must be validated before being sent to the server program.



The Staff table is created in the MySQL database server **as** follows:

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

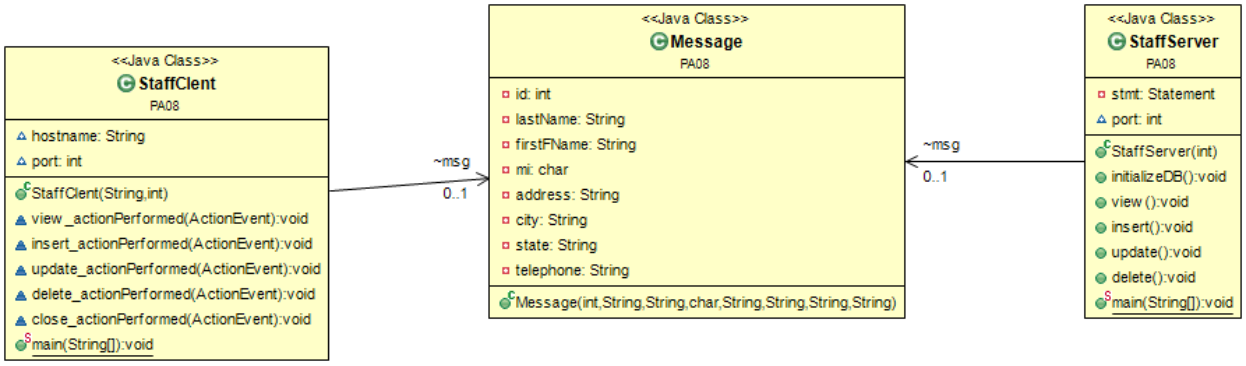
The server application program is a multithreaded server (can open socket connections with multiple clients simultaneously) receives serializable message objects from a client program and performs the view, insert, update, and delete operations on the MySQL database server. In order to connect your server program to your "hostname" MySQL database server, you must configure the class path in your project to point to the External JARs Connecter/J file: mysql-connector-java-5.1.24-bin.jar, which is given in the week 10 practice examples folder. In the server program, you must load the MySQL driver and establish a connection to your **"YourDatabase"** database in the "hostname" of MySQL server as follows:

```
  // Load the JDBC driver
    Class.forName("com.mysql.jdbc.Driver");
    System.out.println("Driver loaded");

// Establish a connection
```

```
   Connection connection = DriverManager.getConnection
   ("jdbc:mysql://hostname/yourdatabase", "username",
"password");
    System.out.println("Database connected\n");
```

The program must be impended with exact classes and methods (names and signatures) as shown in the following class diagrams:



Hint: *Review the provided practice code examples for week 06, such as the Java source files in the database and MySQL Database Project folders. Also, review the provided MySQL links in the Java Resources page in Canvas in order to write SQL statements (Select, Insert, Update, and Delete) for manipulating Staff table in your "Database" defined in the given hostname MySQL server. Be creative in your GUI layout and also the required validations as well as the interactive messages.*

Evaluation Criteria:

1. You must use the class template in your program classes
2. The program must not have any compilation or runtime errors
3. You must use the 3-Tier application database model
4. You must implement the service-oriented model employing a request-response protocol
5. You must use java.net.* package
6. All tasks must be completed to receive a complete credit for this assignment
7. The program must perform all the requirements correctly, including the read and output of data from a user
8. The program must handle incorrect data format and required inputs
9. The program must have a correct logical order and produce the anticipated results
10. The sequence, selection, and iteration structures must constitute correct program logic solutions to the assignment problem
11. The program must terminate gracefully
12. The program must not abnormally abort
13. You must follow the correct submissions format as described in this document

Submission: (This is an individual Assignment!)

Copy the .java source files from the *src* folder in your *work space* to another folder that should be named following the provided naming format in this course, then zip and upload the file under this assignment answer in Canvas.

File Name: *FFFFLLLLPA07.zip (FFFF = your first name and LLLL = your last name)*

Grading Rubric – PA08

Name: _____

Question 1

Requirements	**Comments**	**Max Points Allocated**	**Points Earned**
General Code Structure: Coding of the Java classes, use of Java comments in the source code, indentations, good variable names/class members, successful compilation (no compile error) and running of code (able to execute the program).		3	
Input, Output, User Interface: Proper coding implementation of the logic to read the data and display the expected value, proper coding implementation of user interface, the aesthetics and look and feel of the user interface, handle input of invalid data (required fields, length, input type, ho data is entered, empty space is entered).		5	
Functionality: General algorithm and logic coded for all view, insert, update, delete function, proper coding of 3-Tier model, proper use of the java.net package, module programming by coding methods (e.g., separate method for view, insert, update and delete), method parameters and return type from the method.		12	
Total		**20**	

Total: _____/20