JBK1015 Assignment –MySQL JDBC

JDBC Programs

Program for load mysql class/driver you have build mysql.jar/mysqlconnector.jar

```
import java.sql.Connection;
import java.sql.DriverManager;
public class MySqlConn {
    public static void main(String args[]) {
        try {

    Class.forName("com.mysql.jdbc.Driver");
            Connection con =
DriverManager.getConnection(
        "jdbc:mysql://localhost:3306/test",
"root", "root");
                    //here test is a database name

        System.out.println("Connected....");
        } catch (Exception e) {
    System.out.println("Error to connect mysql");
        } } }
```

Program for Insert Record in Table

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Statement;
```

```
public class InsertRecord {
	public static void main(String[] args) {
		Statement stmt = null;
		try {

	Class.forName("com.mysql.jdbc.Driver");

		Connection con =
DriverManager.getConnection(
	"jdbc:mysql://localhost:3306/test1", "root",
"root");
			stmt = con.createStatement();

sql = "INSERT INTO Person VALUES (11, 'A', 'B',
18)";

stmt.executeUpdate(sql);
	sql = "INSERT INTO Person VALUES (22, 'C',
'D', 25)";

stmt.executeUpdate(sql);
sql = "INSERT INTO Person VALUES (33, 'E', 'F',
30)";

stmt.executeUpdate(sql);
	sql = "INSERT INTO Person VALUES(44, 'S',
'M', 28)";
			stmt.executeUpdate(sql);
```

```
        System.out.println("All record Inserted...");

            } catch (Exception e) {
                e.printStackTrace();
} } }
```

Program for Select Record in Table

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;

public class SelectRecord {
    public static void main(String[] args) {
        Statement stmt = null;
        try {

    Class.forName("com.mysql.jdbc.Driver");

                Connection con =
DriverManager.getConnection(
"jdbc:mysql://localhost:3306/test1", "root",
"root");
                stmt = con.createStatement();
    String sql = "SELECT id, firstName, lastName,
age FROM Person";
                ResultSet rs = stmt.executeQuery(sql);
                while (rs.next()) {
                    // Retrieve by column name
```

```
                int id = rs.getInt("id");
                int age = rs.getInt("age");
            String first = rs.getString("firstName");
            String last = rs.getString("lastName");

                // Display values
            System.out.print("ID is : " + id +"  ||  ");

System.out.print(" Age is:  " + age +"  ||  ");

System.out.print(" FirstName is:  " + first +"  ||  ");

System.out.println(" LastName is:  " + last);
                } } catch (Exception e) {
                e.printStackTrace();
            } } }
```

Program for Update Record in table

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;

public class UpdateRecord {
    public static void main(String[] args) {
        Statement stmt = null;
        try {

    Class.forName("com.mysql.jdbc.Driver");
```

```
            Connection con =
DriverManager.getConnection(
"jdbc:mysql://localhost:3306/test1", "root",
"root");
            stmt = con.createStatement();
String sql = "UPDATE Person SET age = 30
WHERE id in (1, 2)";

            stmt.executeUpdate(sql);
sql = "SELECT id, firstName, lastName,age FROM
Person";

ResultSet rs = stmt.executeQuery(sql);
            while (rs.next()) {
                // Retrieve by column name
                int id = rs.getInt("id");
                int age = rs.getInt("age");
          String first = rs.getString("firstName");
          String last = rs.getString("lastName");

// Display values
          System.out.print("ID is : " + id + "  ||  ");
     System.out.print("  Age is:  " + age + "  ||  ");
System.out.print("  FirstName is:  " + first + "  ||
");
     System.out.println("  LastName is:  " + last);
            } } catch (Exception e) {
            e.printStackTrace();
          } } }
```

Program for Delete Record from table

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
public class DeleteRecord {
	public static void main(String[] args) {
		Statement stmt = null;
		try {

	Class.forName("com.mysql.jdbc.Driver");
Connection con = DriverManager.getConnection(
	"jdbc:mysql://localhost:3306/test1", "root",
"root");
			stmt = con.createStatement();
String sql = "DELETE FROM Person WHERE id =
1";
			stmt.executeUpdate(sql);
sql = "SELECT id, firstName, lastName,age FROM
Person";
	ResultSet rs = stmt.executeQuery(sql);
			while (rs.next()) {
				// Retrieve by column name
				int id = rs.getInt("id");
				int age = rs.getInt("age");
		String first = rs.getString("firstName");
		String last = rs.getString("lastName");
```

```
// Display values
        System.out.print("ID is : " + id + "  ||  ");
    System.out.print("  Age is:  " + age + "  ||  ");
System.out.print("FirstName is:  " + first +"  || ");
    System.out.println("  LastName is:  " + last);
            } } catch (Exception e) {
            e.printStackTrace();
        } } }
```

MySQL Queries

Inserting Rows

```
-- Insert a row with all the column values
 INSERT INTO products VALUES (1001, 'PEN', 'Pen Red', 5000, 1.23);

-- Insert multiple rows in one command
-- Inserting NULL to the auto_increment column results in max_value + 1
 INSERT INTO products VALUES
      (NULL, 'PEN', 'Pen Blue',  8000, 1.25),
      (NULL, 'PEN', 'Pen Black', 2000, 1.25);

-- Insert value to selected columns
-- Missing value for the auto_increment column also results in max_value + 1
 INSERT INTO products (productCode, name, quantity, price) VALUES
      ('PEC', 'Pencil 2B', 10000, 0.48),
      ('PEC', 'Pencil 2H', 8000, 0.49) ;
```

```
-- Missing columns get their default values
INSERT INTO products (productCode, name) VALUES ('PEC', 'Pencil HB');

-- 2nd column (productCode) is defined to be NOT NULL
INSERT INTO products values (NULL, NULL, NULL, NULL, NULL);

--show table
SELECT * FROM products;

-- Remove the specific row
DELETE FROM products WHERE productID = 1006;

-- List all rows for the specified columns
SELECT name, price FROM products;
```

SELECT without Table

You can also issue SELECT without a table. For example, you can SELECT an expression or evaluate a built-in function.