Assignment – 22

Prakash Manikanta Irrinki
Prakashnaidu9494@gmail.com

Task 1:

Establishing Database Connections:

Write a Java program that connects to a SQLite database and prints out the connection object to confirm successful connection.

Program:

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

public class ConnectionEstablish {

	public static void main(String[] args) {

		Connection con=null;
		try {
			DriverManager.registerDriver(new
					com.mysql.cj.jdbc.Driver());

	con=DriverManager.getConnection("jdbc:mysql://localhost:3306/
					practicedb", "root", "Sys@123");
			System.out.println("Successfully connected: "+con);
		} catch (SQLException e) {
			// TODO Auto-generated catch block
			e.printStackTrace();
		}
		finally {
			try {
				con.close();
			} catch (SQLException e) {
				// TODO Auto-generated catch block
				e.printStackTrace();
			}
		}
	}
}
```

Output:

```
Successfully connected: com.mysql.cj.jdbc.ConnectionImpl@747edf66
```

Task 2:

SQL Queries using JDBC:

Create a table 'User' with a following schema 'User ID' and 'Password' stored as hash format (note you have research on how to generate hash from a string), accept ""User ID"" and ""Password"" as input and check in the table if they match to confirm whether user access is allowed or not.

Program:

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
public class SQLQueriesUsingJDBC {
	public static void main(String[] args) {
		String actualhashpassword = null;
		String expectedhashpassword = null;
		try {
			DriverManager.registerDriver(
					new com.mysql.cj.jdbc.Driver());
			Connection con = DriverManager.getConnection(
					"jdbc:mysql://localhost:3306/practicedb",
						"root", "Sys@123");
			String select = "select md5(userid),md5(password)
				from user where userid=? AND password=?";
			String select1 = "select md5(password) from user
						where userid=?";
			PreparedStatement pstmt = con.prepareStatement
							(select);
			PreparedStatement pstmt1 = con.prepareStatement
							(select1);
			pstmt.setString(1, "teja");
			pstmt.setString(2, "teja001");
			pstmt1.setString(1, "teja");
			ResultSet rs = pstmt.executeQuery();
			ResultSet rs1 = pstmt1.executeQuery();
			while (rs1.next()) {
			expectedhashpassword=rs1.getString("md5(password)");
			}
			while (rs.next()) {
				actualhashpassword =
						rs.getString("md5(password)");
				break;
			}

		} catch (SQLException e) {
			e.printStackTrace();
		}
		try {
			if (expectedhashpassword.equals(actualhashpassword))
			{
			System.out.println("User access is Allowed..");
			}
			if(!expectedhashpassword.equals(actualhashpassword))
			{
			System.out.println("User access is  Not Allowed..");
```

```
                }
            } catch (NullPointerException e) {
            }
      }
}
```

Output:

```
User access is Allowed..
```

Task 3:

Prepared Statement:

Modify the SELECT query program to use PreparedStatement to parameterize the query and prevent SQL injection.

Program:

```
import java.sql.Connection;
import java.sql.Date;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
public class ConnectionEstablish {
      public static void main(String[] args) {
            Connection con=null;
            try {
                  DriverManager.registerDriver(
                        new com.mysql.cj.jdbc.Driver());
                  con=DriverManager.getConnection
                        ("jdbc:mysql://localhost:3306/
                                    practicedb", "root", "Sys@123");
                  String query="select * from customer where cid= ? ";
                  PreparedStatement pstmt=con.prepareStatement(query);
                  pstmt.setInt(1, 102);
                  ResultSet rs=pstmt.executeQuery();
                  while(rs.next()) {
                        int cid=rs.getInt("cid");
                        String cname=rs.getString("cname");
                        String address=rs.getString("address");
                        Date date=rs.getDate("dob");
                        System.out.println(cid+" "+cname+" "+address+"
                                                "+date);
                  }
            } catch (SQLException e) {
                  e.printStackTrace();
            }
            finally {
                  try {
                        con.close();
                  } catch (SQLException e) {
                        e.printStackTrace();
                  }
            }
      }
}
```

Output:

```
102 teja Akividu 2001-05-18
```