LAB2

Q1

```java
package appledb;

import java.sql.*;
import java.io.*;

class User {
    public String username;
    public String password;
    public String role;
    User(String un, String pw,String urole){
        username = un;
        password = pw;
        role = urole;
    }
}

public class AppleDB {

    /**
     * @param args the command line arguments
     */
    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/shubhsin";
    static final String USER = "root";
    static final String PASS = "student";
    static User loggedUser;
    static boolean logged = false;
    static Connection conn = null;
    static Statement stmt =null;
    AppleDB() {

    }

    public static void main(String[] args) {

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

```java
    System.out.println("Connecting to database...");
    conn = DriverManager.getConnection(DB_URL,USER,PASS);
}
catch(ClassNotFoundException | SQLException e) { }

while(true) {
System.out.println("Select Option - ");
System.out.println("1. Login\n2. Register\n3. Logout\n4. Quit\n");
String ip = "";
BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
try {
ip = br.readLine();
}
catch(Exception e) {}

finally {
    int choice = Integer.parseInt(ip);
    switch(choice) {
        case 1:loginUser();
            break;
        case 2:registerUser();
            break;
        case 3:logoutUser();
            break;
        case 4:System.exit(0);
            break;

        default: System.exit(0);
            break;
    }
}
}
}


public static void loginUser(){
    System.out.print("Username : ");
    BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
    String un = "";
    String pw = "";
    try {
    un = br.readLine();
    }catch(Exception e) {}
    System.out.print("Password :");
    try{
```

```java
        pw = br.readLine();}
        catch(Exception e) {}

        try {
        stmt = conn.createStatement();
        String sql;
        sql = "SELECT * from users";
        ResultSet rs = stmt.executeQuery(sql);
        while(rs.next()) {
            String first = rs.getString("uname");
            String last = rs.getString("pwd");
            String role = rs.getString("role");
            if(un.equals(first) && pw.equals(last)) {
                System.out.println(first +" logged in as "+role );
                logged =true;
                loggedUser = new User(first,last,role);
            }

        }
        if(!logged) {
            System.out.println("Sorry Wrong Credentials.");
        }

        }
        catch(Exception e){

        }


    }

    public static void registerUser() {
        System.out.println("Chose a role");
        System.out.println("1. Student\n2. Teacher\n3. ADMIN\n");
        String ip = "";
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        try {
        ip = br.readLine();
        }
        catch(Exception e) {}
        finally{}
        String uname = "";
        String pwd = "";
        try {
            System.out.println("Username :");
```

```java
        uname = br.readLine();
        System.out.println("Password :");
        pwd = br.readLine();
    }
    catch(Exception e){}
    finally {
        String role = "";
        int ch = Integer.parseInt(ip);
        switch(ch) {
            case 1:role = "student";
                    break;
            case 2:role = "teacher";
                    break;
            case 3:role = "admin";
                    break;
            default:break;
        }
        try{
        stmt = conn.createStatement();
        String sql;
        PreparedStatement p = null;

        sql = "INSERT INTO users values(?,?,?)";
        p = conn.prepareStatement(sql);
        p.setString(1, uname);
        p.setString(2, pwd);
        p.setString(3, role);
        p.execute();
        p.close();

        }
        catch(Exception e) {}
        finally{
            System.out.println("Registered Successfully!");
        }

    }
}

public static void logoutUser() {
    if(logged) {
    loggedUser = null;
    logged = false;
    System.out.println("Logged out successfully");
    }
```

```
      }

}
```

Q2

```java
package bankdb;
import java.sql.*;
import java.io.*;

class User {
    public String username;
    public String password;
    public String role;
    public int balance;
    public String status;
    User(String un, String pw,int bal,String urole){
        username = un;
        password = pw;
        role = urole;
        balance = bal;
    }
}

public class BankDB {
    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost/shubhsin";
    static final String USER = "root";
    static final String PASS = "student";
    static User loggedUser;
    static boolean logged = false;
    static Connection conn = null;
    static Statement stmt =null;

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here

        try {
            //
```

```java
    Class.forName("com.mysql.jdbc.Driver");
    //
    System.out.println("Connecting to database...");
    conn = DriverManager.getConnection(DB_URL,USER,PASS);
}
catch(ClassNotFoundException | SQLException e) { }

 while(true) {
 System.out.println("Select Option - ");
 System.out.println("1. Login\n2. Register\n3. Logout\n4. Check Balance\n5. Show
Transactions\n6. Admin Panel\n7. Exit\n");
 String ip = "";
 BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
 try {
 ip = br.readLine();
 }
 catch(Exception e) {}

 finally {
    int choice = Integer.parseInt(ip);
    switch(choice) {
        case 1:loginUser();
            break;
        case 2:registerUser();
            break;
        case 3:logoutUser();
            break;
        case 4:{ if(logged)
                checkBalance();
             else
                System.out.println("You need to login");
             break;
             }
        case 5: { if(logged)
                showTransactions();
             else
                System.out.println("You need to login");
             break;
             }
        case 6: if(logged) {
                if(loggedUser.role.equals("admin"))
                   showAdminMenu();
                else
                   System.out.println("You are not logged in as an admin!");
                }
```

```java
                break;
            case 7:System.exit(0);
                    break;

            default: System.exit(0);
                    break;
        }
    }
}


}

public static void loginUser(){
    System.out.print("Username : ");
    BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
    String un = "";
    String pw = "";
    try {
    un = br.readLine();
    }catch(Exception e) {}
    System.out.print("Password :");
    try{
    pw = br.readLine();}
    catch(Exception e) {}

    try {
    stmt = conn.createStatement();
    String sql;
    sql = "SELECT * from bankdb";
    ResultSet rs = stmt.executeQuery(sql);
    while(rs.next()) {
        String first = rs.getString("username");
        String last = rs.getString("pass");
        int bal = rs.getInt("balance");
        String role = rs.getString("role");
        String stat = rs.getString("status");
        if(un.equals(first) && pw.equals(last)) {
            if(stat.equals("open")){
            System.out.println(first +" logged in as "+role );
            System.out.println("Current Balance - "+bal);
            logged =true;
            loggedUser = new User(first,last,bal,role);
            }
            else {
                System.out.println("Your account is frozen");
```

```java
                }
            }

        }
        if(!logged) {
            System.out.println("Sorry Wrong Credentials / Account Frozen");
        }
        }
        catch(Exception e){

        }
    }

    public static void registerUser(){
        System.out.println("Chose a role");
        System.out.println("1. User\n2. Admin\n");
        String ip = "";
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        try {
        ip = br.readLine();
        }
        catch(Exception e) {}
        finally{}
        String uname = "";
        String pwd = "";
        try {
            System.out.println("Username :");
            uname = br.readLine();
            System.out.println("Password :");
            pwd = br.readLine();
        }
        catch(Exception e){}
        finally {
            String role = "";
            int ch = Integer.parseInt(ip);
            switch(ch) {
                case 1:role = "user";
                        break;
                case 2:role = "admin";
                        break;
                default:break;
            }
            try{
            stmt = conn.createStatement();
            String sql;
```

```java
            PreparedStatement p = null;

            sql = "INSERT INTO bankdb values(?,?,?,?)";
            p = conn.prepareStatement(sql);
            p.setString(1, uname);
            p.setString(2, pwd);
            p.setInt(3, 0);
            p.setString(4,role);
            p.execute();
            p.close();

            }
            catch(Exception e) {}
            finally{
                System.out.println("Registered Successfully!");
            }

        }

    }

    public static void logoutUser() {
        if(logged) {
        loggedUser = null;
        logged = false;
        System.out.println("Logged out successfully");
        }
    }

    public static void checkBalance() {
        System.out.println("Current Balance - "+loggedUser.balance);
    }

    public static void showTransactions() {
        try {
        stmt = conn.createStatement();
        String sql;
        sql = "SELECT * from banktran";
//      PreparedStatement p = conn.prepareStatement(sql);
//      p.setString(1,loggedUser.username);
//      p.execute();

        ResultSet rs = stmt.executeQuery(sql);
        while(rs.next()) {
            String first = rs.getString("uname");
```

```java
        if(loggedUser.username.equals(first)) {
            int amt = rs.getInt("detail");
            System.out.println(amt);
        }

    }

    }
    catch(Exception e){

    }
}

/*

ADMIN OPERATIONS CODE

|| CONFIDENTIAL ||

*/
public static void showAdminMenu() {
    System.out.println("Enter your choice :");
    System.out.println("1. Freeze Account\n2. Unblock Account\n3. Transact");
    String ip = "";
    BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
    try {
    ip = br.readLine();
    }
    catch(Exception e) {}
    finally{}
    int ch = Integer.parseInt(ip);
    switch(ch){
        case 1:{
            System.out.println("Enter account's username : ");
            try{
                ip = br.readLine();
            }
            catch(Exception e){

            }
            finally{
                try {
                stmt = conn.createStatement();
                String sql;
```

```java
            PreparedStatement p = null;
            sql = "UPDATE BANKDB SET STATUS=? where USERNAME=?";
            p = conn.prepareStatement(sql);
            p.setString(2, ip);
            p.setString(1,"freeze");
            p.execute();
            p.close();
            }
            catch(Exception e){}
            finally{System.out.println("Successfully Froze "+ip+"'s Account");}
        }
    }
        break;
    case 2:{
        System.out.println("Enter account's username : ");
        try{
            ip = br.readLine();
        }
        catch(Exception e){

        }
        finally{
            try {
            stmt = conn.createStatement();
            String sql;
            PreparedStatement p = null;
            sql = "UPDATE BANKDB SET STATUS=? where USERNAME=?";
            p = conn.prepareStatement(sql);
            p.setString(2, ip);
            p.setString(1,"open");
            p.execute();
            p.close();
            }
            catch(Exception e){}
            finally{System.out.println("Successfully Unblocked "+ip+"'s
Account");}
        }
    }
        break;
    case 3: {
        System.out.println("Enter username :");
        String uname = "";
        int calAmt ;
        String amt="";
        try{
```

```java
            uname = br.readLine();
            }catch(Exception e){}
            System.out.println("Enter Amount :");
            try{
            amt = br.readLine();
            calAmt = Integer.parseInt(amt);
            }catch(Exception e){}
            try{
                stmt = conn.createStatement();
                String sql;
                PreparedStatement p = null;
                sql = "UPDATE BANKDB SET BALANCE=? where
USERNAME=?";
                p = conn.prepareStatement(sql);
                p.setString(2, uname);
                p.setString(1,amt);
                p.execute();
                p.close();
            }
            catch(Exception e){}

            try{
                stmt = conn.createStatement();
                String sql;
                PreparedStatement p = null;
                sql = "INSERT INTO BANKTRAN VALUES (?,?,?)";
                p = conn.prepareStatement(sql);
                p.setString(1, uname);
                p.setString(2,amt);
                p.setTimestamp(3, getCurrentTimeStamp());
                p.execute();
                p.close();
            }
            catch(Exception e){}

        }
        break;
        default:break;
    }
  }

    private static java.sql.Timestamp getCurrentTimeStamp() {

        java.util.Date today = new java.util.Date();
        return new java.sql.Timestamp(today.getTime());
```

```
    }
}
```