Authentication.java

```java
package org.main;

import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

import org.model.Contact;
import org.model.User;
import org.util.DBFactory;

public class Authentication {
    public static boolean check(User user) throws
ClassNotFoundException, SQLException {
        boolean status = false;
        Connection con = DBFactory.getConn();
        Statement stmt = con.createStatement();
        ResultSet rs = stmt.executeQuery("select * from
contact_details"
                + " where username='" + user.getUsername()
                + "' and password='" + user.getPassword() +
"'");
        if(rs.next()) {
            status = true;
        }
        rs.close();
        stmt.close();
        return status;
    }
}
```

Contact.java

```java
package org.main;

import java.sql.Connection;
import java.sql.SQLException;
import java.sql.Statement;

import org.model.Contact;
import org.util.DBFactory;

public class ContactDAO {

    public static void delete(Integer phnNumber) throws
ClassNotFoundException, SQLException {
        Connection con = DBFactory.getConn();
        Statement stmt = con.createStatement();
        String query = "delete from contact where phnoneNumber =
" + phnNumber;
        int count = stmt.executeUpdate(query);
        if (count > 0)
            System.out.println("Contact removed");
        else
            System.out.println("contact found with this number
#" + phnNumber);
        stmt.close();
    }

    public static void insert(Contact contact) throws
ClassNotFoundException, SQLException {
        Connection con = DBFactory.getConn();
        Statement stmt = con.createStatement();
        String query = "insert into contact values(" +
contact.getPhnNumber() + ", '" + contact.getFirstName() + "', "
                    + contact.getLastName() + ")";
        stmt.executeUpdate(query);
        System.out.println("Values Inserted");
        stmt.close();
    }
}
```

Solution.java

```java
package org.main;

import java.io.BufferedReader;
import java.io.Console;
import java.io.IOException;
import java.io.InputStreamReader;
import java.sql.SQLException;

import org.model.Contact;
import org.model.User;

public class Solution {

    public static void main(String[] args) throws IOException {
        BufferedReader cons = new BufferedReader(new
InputStreamReader(System.in));
        System.out.print("Username :");
        String username = cons.readLine();
        System.out.print("Password :");
        String password = cons.readLine();
        User user = new User( username, password);
        try {
            if (Authentication.check(user)) {
                System.out.println("Login Success");
                boolean flag = false;
                while (true) {
                    System.out.print("1.Create  2.Delete \n
Choice:");
                    switch
(Integer.parseInt(cons.readLine())) {
                    case 1:
                        Contact contact1 = new Contact();

    contact1.setPhnNumber(Double.parseDouble(cons.readLine()));

    contact1.setFirstName(cons.readLine());

    contact1.setLastName(cons.readLine());
                        ContactDAO.insert(contact1);
                        break;
                    case 2:
                        Integer phnNumber =
Integer.parseInt(cons.readLine());
                        ContactDAO.delete(phnNumber);
                        break;
                    default:
                        flag = true;
                        break;
                    }
                    if(flag)
                        break;
                }
            }
```

```java
                else {
                        System.out.println("Invalid Login Details");
                }
        }
        catch (ClassNotFoundException | SQLException e) {
                e.printStackTrace();
        }
    }

}
```

Contact.java

```java
package org.model;

public class Contact {

    double phnNumber;
    String firstName;
    String lastName;
    String mailId;
    public String getMailId() {
        return mailId;
    }
    public void setMailId(String mailId) {
        this.mailId = mailId;
    }
    public double getPhnNumber() {
        return phnNumber;
    }
    public void setPhnNumber(double phnNumber) {
        this.phnNumber = phnNumber;
    }
    public String getFirstName() {
        return firstName;
    }
    public void setFirstName(String firstName) {
        this.firstName = firstName;
    }
    public String getLastName() {
        return lastName;
    }
    public void setLastName(String lastName) {
        this.lastName = lastName;
    }


    public Contact(Integer phnNumber, String firstName, String
lastName) {
        super();
        this.phnNumber = phnNumber;
        this.firstName = firstName;
        this.lastName = lastName;
    }

    public Contact() {
        super();
    }


}
```

User.java

```java
package org.model;

public class User {
      private String username;
      private String password;

      public String getUsername() {
            return username;
      }

      public void setUsername(String username) {
            this.username = username;
      }

      public String getPassword() {
            return password;
      }

      public void setPassword(String password) {
            this.password = password;
      }

      public User(String username, String password) {
            super();
            this.username = username;
            this.password = password;
      }

      public User() {
            super();
      }

}
```

DBFactory.java

```java
package org.util;

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

public class DBFactory {
    public static Connection getConn() throws
ClassNotFoundException, SQLException {
        if (connection == null) {
            Class.forName("com.mysql.jdbc.Driver");
            connection =
DriverManager.getConnection("jdbc:mysql://localhost:3306/test",
"root", "");
        }
        return connection;
    }

    private static Connection connection = null;
}
```