SSN College of Engineering, Kalavakkam

Department of Computer Science and Engineering

UCS1711-Mobile Application Development Lab

Exercise 5:Andriod Application for CRUD Operations on Database

Name : S.NACHAMMAI DEVI POOJA

Class :CSE Sec: B

Reg no: 185001096

AIM:

To develop an android application for CRUD operations in Database.

CODE:

MainActivity.java:

```
package com.example.database;


import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
```

```
public class MainActivity extends AppCompatActivity {
    EditText name, contact, dob, email;
    Button insert, update, delete, view;
    DBHelper DB;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        name = findViewById(R.id.name);
        contact = findViewById(R.id.contact);
        dob = findViewById(R.id.dob);
        email = findViewById(R.id.email);

        insert = findViewById(R.id.btnInsert);
        update = findViewById(R.id.btnUpdate);
        delete = findViewById(R.id.btnDelete);
        view = findViewById(R.id.btnView);
        DB = new DBHelper(this);
        insert.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
```

```
                String nameTXT = name.getText().toString();

                String contactTXT = contact.getText().toString();

                String dobTXT = dob.getText().toString();

                String emailTXT = email.getText().toString();


                Boolean checkinsertdata =
DB.insertuserdata(nameTXT, contactTXT, dobTXT, emailTXT);

                if(checkinsertdata==true)

                    Toast.makeText(MainActivity.this, "New Entry
Inserted", Toast.LENGTH_SHORT).show();

                else

                    Toast.makeText(MainActivity.this, "New Entry
Not Inserted", Toast.LENGTH_SHORT).show();

            }         });

        update.setOnClickListener(new View.OnClickListener() {

            @Override

            public void onClick(View view) {

                String nameTXT = name.getText().toString();

                String contactTXT = contact.getText().toString();

                String dobTXT = dob.getText().toString();

                String emailTXT = email.getText().toString();


                Boolean checkupdatedata =
DB.updateuserdata(nameTXT, contactTXT, dobTXT, emailTXT);

                if(checkupdatedata==true)
```

```
                    Toast.makeText(MainActivity.this, "Entry
Updated", Toast.LENGTH_SHORT).show();

                else

                    Toast.makeText(MainActivity.this, "New Entry
Not Updated", Toast.LENGTH_SHORT).show();

            }         });

        delete.setOnClickListener(new View.OnClickListener() {

            @Override

            public void onClick(View view) {

                String nameTXT = name.getText().toString();

                Boolean checkudeletedata =
DB.deletedata(nameTXT);

                if(checkudeletedata==true)

                    Toast.makeText(MainActivity.this, "Entry
Deleted", Toast.LENGTH_SHORT).show();

                else

                    Toast.makeText(MainActivity.this, "Entry Not
Deleted", Toast.LENGTH_SHORT).show();

            }         });


        view.setOnClickListener(new View.OnClickListener() {

            @Override

            public void onClick(View view) {

                Cursor res = DB.getdata();

                if(res.getCount()==0){
```

```
                    Toast.makeText(MainActivity.this, "No Entry
Exists", Toast.LENGTH_SHORT).show();

                    return;

                }

                StringBuffer buffer = new StringBuffer();

                while(res.moveToNext()){

                    buffer.append("Name
:"+res.getString(0)+"\n");

                    buffer.append("Contact
:"+res.getString(1)+"\n");

                    buffer.append("Address
:"+res.getString(2)+"\n");

                    buffer.append("Email
:"+res.getString(3)+"\n\n");

                }


                AlertDialog.Builder builder = new
AlertDialog.Builder(MainActivity.this);

                builder.setCancelable(true);

                builder.setTitle("User Entries");

                builder.setMessage(buffer.toString());

                builder.show();

            }         });

    }}
```

activity_main.xml :

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<RelativeLayout tools:context=".MainActivity"
android:padding="10dp" android:layout_height="match_parent"
android:layout_width="match_parent"
xmlns:tools="http://schemas.android.com/tools"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:android="http://schemas.android.com/apk/res/android">


<TextView android:layout_height="wrap_content"
android:layout_width="match_parent"
android:textColor="@color/colorPrimary"
android:layout_marginTop="20dp" android:textSize="28dp"
android:text="         Fill the required details"
android:id="@+id/texttitle"/>


<EditText android:layout_height="wrap_content"
android:layout_width="match_parent"
android:textColor="@color/colorPrimary" android:textSize="24dp"
android:id="@+id/name" android:inputType="textPersonName"
android:layout_below="@+id/texttitle" android:hint="Enter
Name"/>


<EditText android:layout_height="wrap_content"
android:layout_width="match_parent"
android:textColor="@color/colorPrimary" android:textSize="24dp"
android:id="@+id/contact" android:inputType="number"
android:layout_below="@+id/name" android:hint="Enter Contact"/>


<EditText android:layout_height="wrap_content"
android:layout_width="match_parent"
android:textColor="@color/colorPrimary" android:textSize="24dp"
android:id="@+id/dob" android:inputType="text"
android:layout_below="@+id/contact" android:hint="Enter
Address"/>
```

```
<EditText android:layout_height="wrap_content"
android:layout_width="match_parent"
android:textColor="@color/colorPrimary" android:textSize="24dp"
android:id="@+id/email" android:inputType="text"
android:layout_below="@+id/dob" android:hint="Enter Email"/>
```

```
<Button android:layout_height="wrap_content"
android:textColor="@color/colorPrimary"
android:background="@color/colorAccent"
android:layout_width="match_parent"
android:layout_marginTop="65dp" android:textSize="24dp"
android:text="Insert New Data" android:id="@+id/btnInsert"
android:layout_below="@id/email"/>
```

```
<Button android:layout_height="wrap_content"
android:textColor="@color/colorPrimary"
android:background="@color/colorAccent"
android:layout_width="match_parent"
android:layout_marginTop="15dp" android:textSize="24dp"
android:text="Update Data" android:id="@+id/btnUpdate"
android:layout_below="@id/btnInsert"/>
```

```
<Button android:layout_height="wrap_content"
android:textColor="@color/colorPrimary"
android:background="@color/colorAccent"
android:layout_width="match_parent"
android:layout_marginTop="15dp" android:textSize="24dp"
android:text="Delete Existing Data" android:id="@+id/btnDelete"
android:layout_below="@id/btnUpdate"/>
```

```
<Button android:layout_height="wrap_content"
android:textColor="@color/colorPrimary"
android:background="@color/colorAccent"
```

```
android:layout_width="match_parent"
android:layout_marginTop="15dp" android:textSize="24dp"
android:text="View Data" android:id="@+id/btnView"
android:layout_below="@id/btnDelete"/>

</RelativeLayout>
```

Colors.xml:

```
<?xml version="1.0" encoding="utf-8"?>

<resources>

    <color name="purple_200">#FFBB86FC</color>

    <color name="purple_500">#13299E</color>

    <color name="purple_700">#FF3700B3</color>

    <color name="teal_200">#FF03DAC5</color>

    <color name="teal_700">#FF018786</color>

    <color name="black">#FF000000</color>

    <color name="white">#FFFFFFFF</color>

    <color name="colorPrimary">#9E77E3</color>

    <color name="colorAccent">#0B7369</color>

</resources>
```

DBHelper.java:

```
package com.example.database;



        import android.content.ContentValues;
```

```
        import android.content.Context;
        import android.database.Cursor;
        import android.database.sqlite.SQLiteDatabase;
        import android.database.sqlite.SQLiteOpenHelper;

        import androidx.annotation.Nullable;

public class DBHelper extends SQLiteOpenHelper {
    public DBHelper(Context context) {
        super(context, "Userdata.db", null, 1);
    }

    @Override
    public void onCreate(SQLiteDatabase DB) {
        DB.execSQL("create Table Userdetails(name TEXT primary
key, contact TEXT, dob TEXT, email TEXT)");
    }

    @Override
    public void onUpgrade(SQLiteDatabase DB, int i, int i1) {
        DB.execSQL("drop Table if exists Userdetails");
    }
```

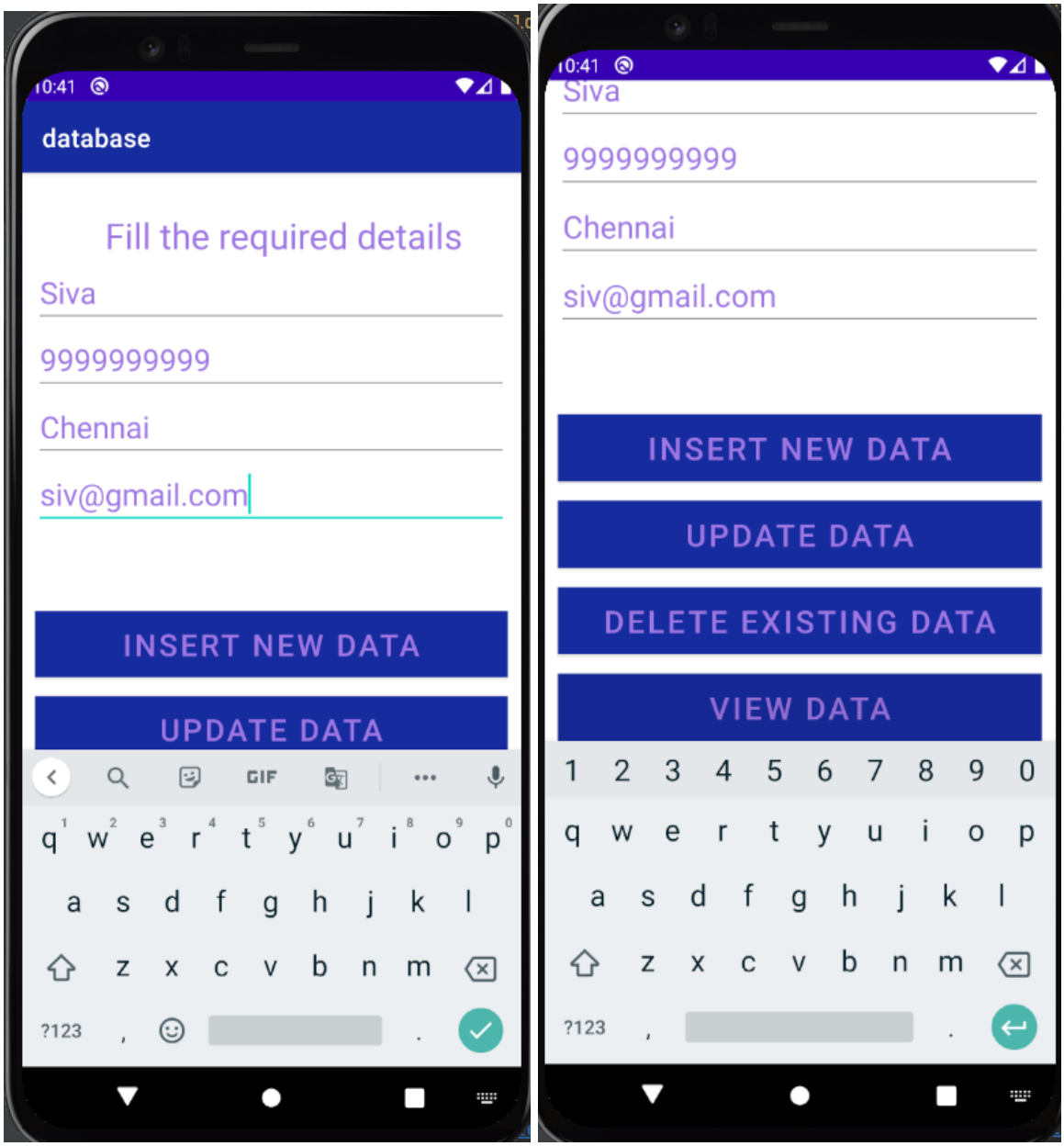
```
    public Boolean insertuserdata(String name, String contact,
String dob, String email)
    {
        SQLiteDatabase DB = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put("name", name);
        contentValues.put("contact", contact);
        contentValues.put("dob", dob);
        contentValues.put("email", email);

        long result=DB.insert("Userdetails", null,
contentValues);
        if(result==-1){
            return false;
        }else{
            return true;
        }
    }

    public Boolean updateuserdata(String name, String contact,
String dob, String email) {
        SQLiteDatabase DB = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put("contact", contact);
```

```
        contentValues.put("dob", dob);
        contentValues.put("email", email);


        Cursor cursor = DB.rawQuery("Select * from Userdetails
where name = ?", new String[]{name});
        if (cursor.getCount() > 0) {
            long result = DB.update("Userdetails", contentValues,
"name=?", new String[]{name});
            if (result == -1) {
                return false;
            } else {
                return true;
            }
        } else {
            return false;
        }}


    public Boolean deletedata (String name)
    {
        SQLiteDatabase DB = this.getWritableDatabase();
        Cursor cursor = DB.rawQuery("Select * from Userdetails
where name = ?", new String[]{name});
        if (cursor.getCount() > 0) {
```

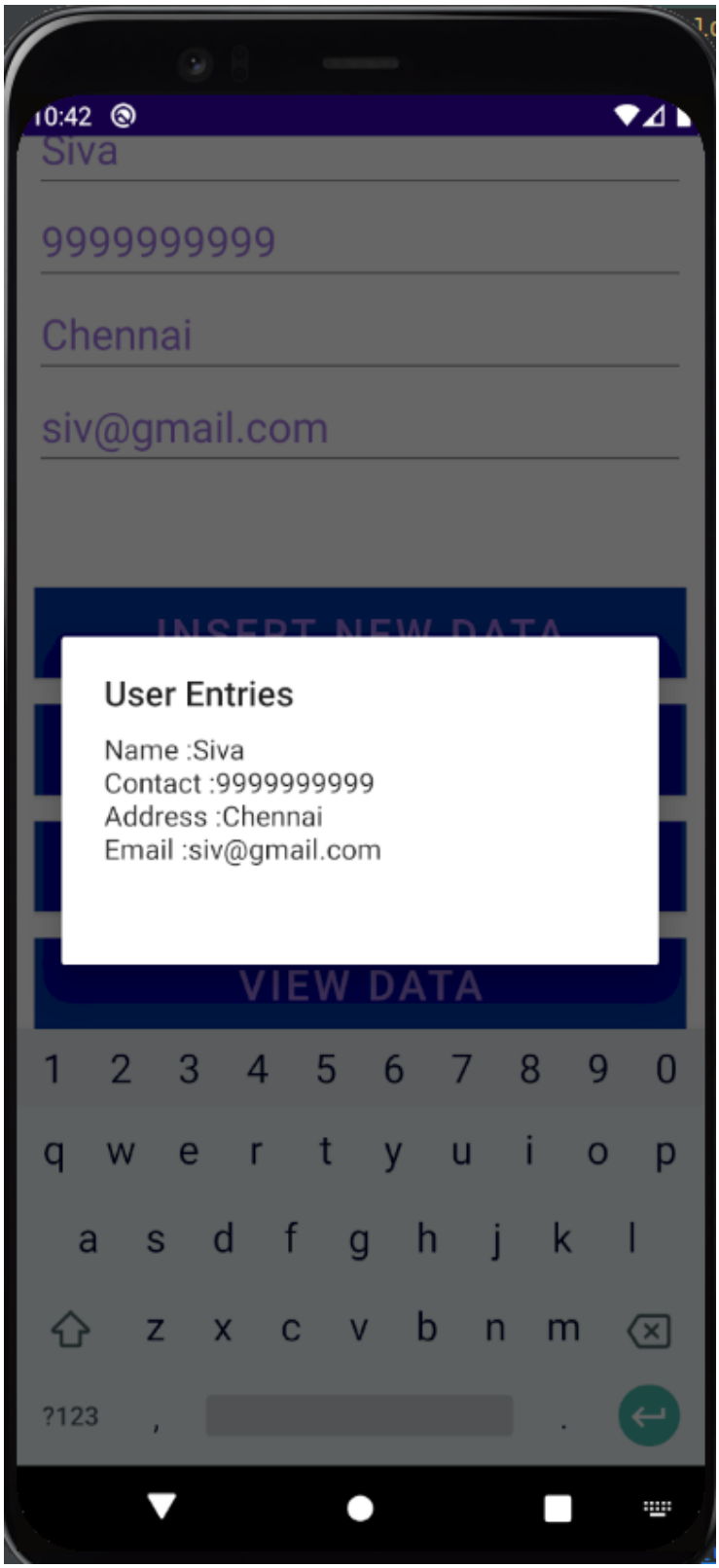
```
            long result = DB.delete("Userdetails", "name=?", new
String[]{name});
            if (result == -1) {
                return false;
            } else {
                return true;
            }
        } else {
            return false;
        }
    }
    public Cursor getdata ()
    {
        SQLiteDatabase DB = this.getWritableDatabase();
        Cursor cursor = DB.rawQuery("Select * from Userdetails",
null);
        return cursor;
    }
}
```

OUTPUT:

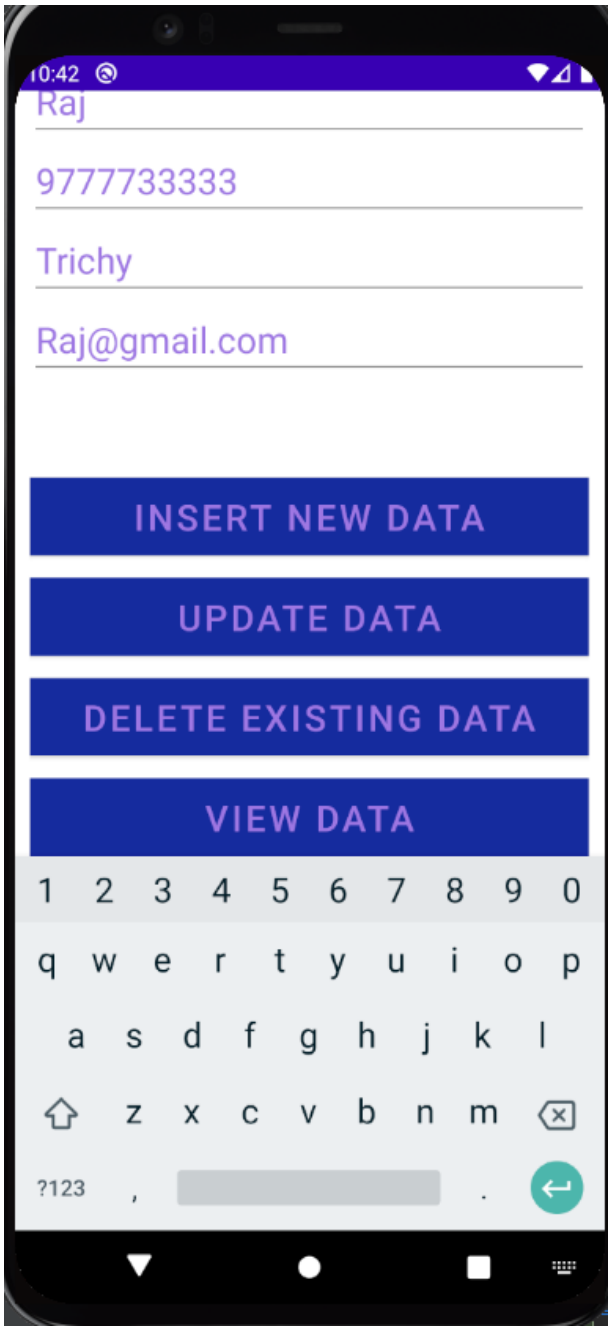
Insert Data:

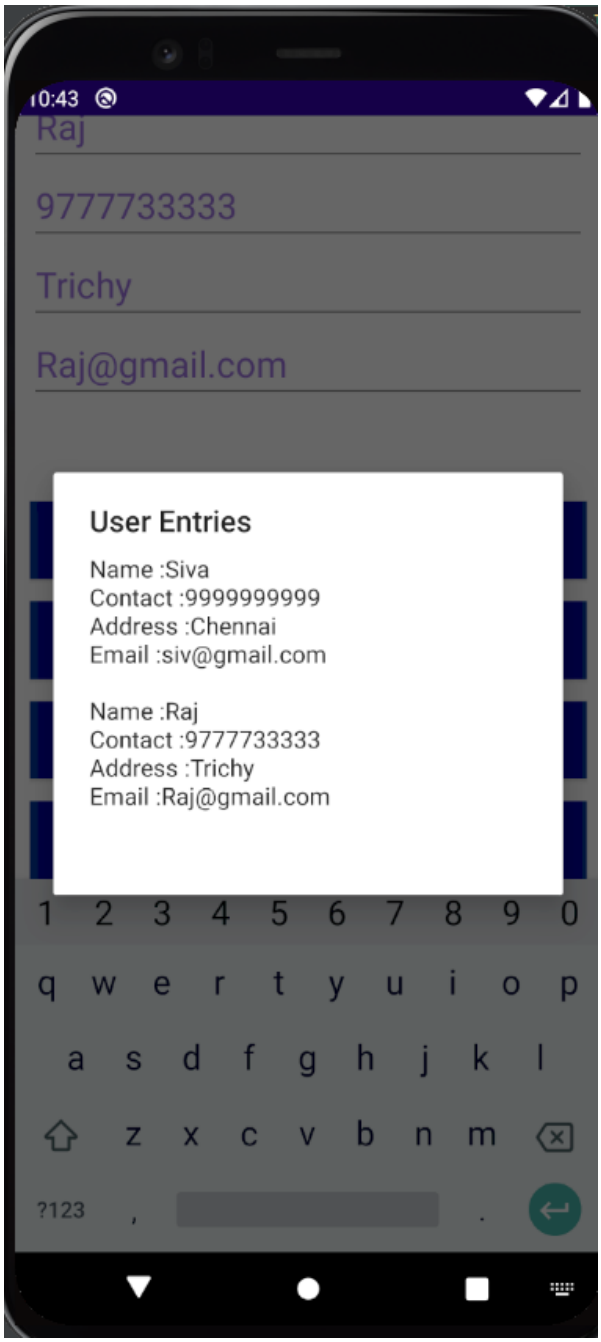

10:41
database
Fill the required details
Siva
9999999999
Chennai
siv@gmail.com
INSERT NEW DATA
UPDATE DATA
10:41
Siva
9999999999
Chennai
siv@gmail.com
INSERT NEW DATA
UPDATE DATA
DELETE EXISTING DATA
VIEW DATA

View Data:

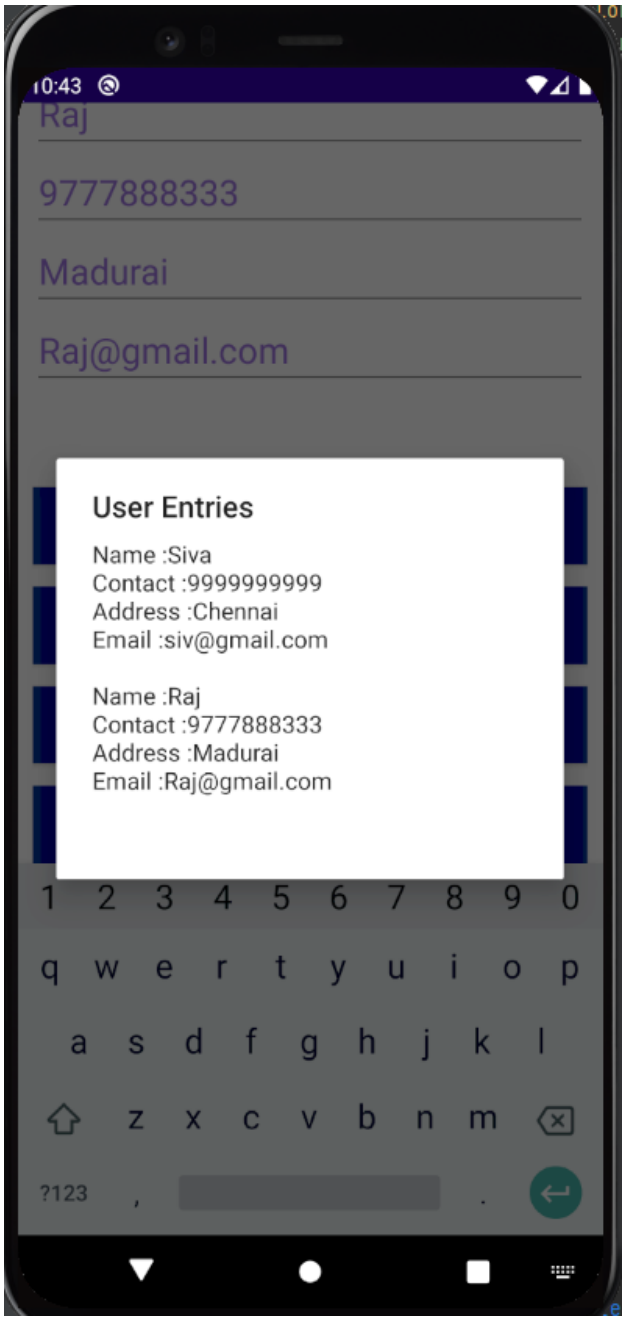

10:42
Siva
9999999999
Chennai
siv@gmail.com
User Entries
Name :Siva
Contact :9999999999
Address :Chennai
Email :siv@gmail.com
VIEW DATA

Insert new data:

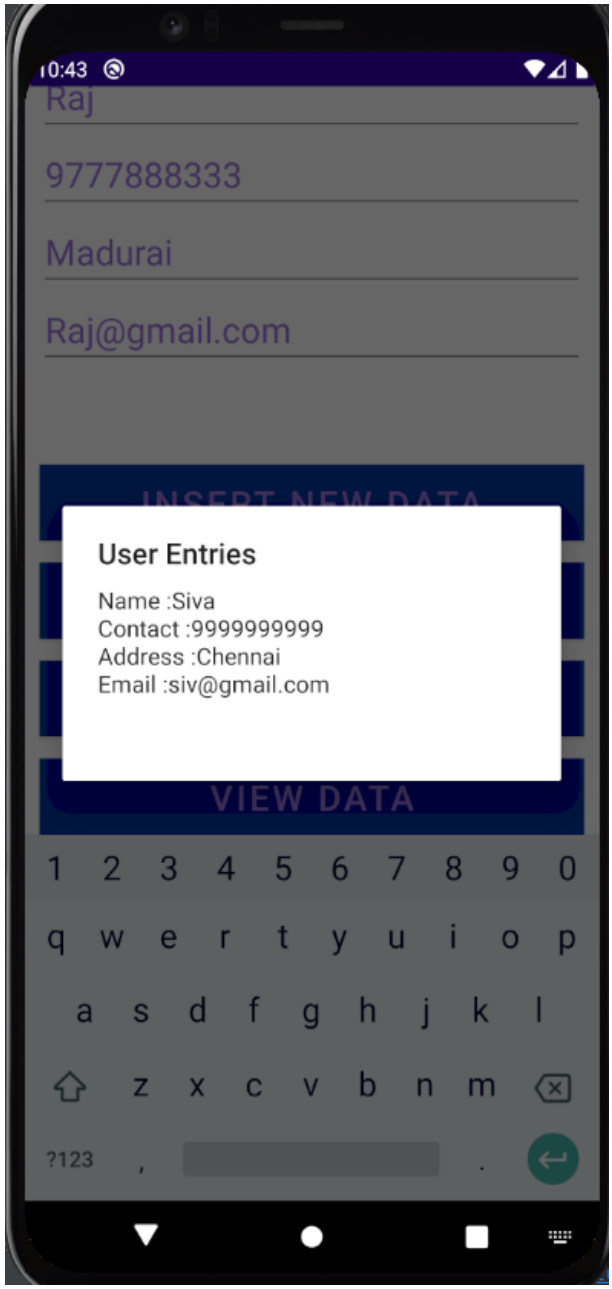




Update data:



Delete Data:



RESULT:

An android application for CRUD operations in Database has been successfully developed.