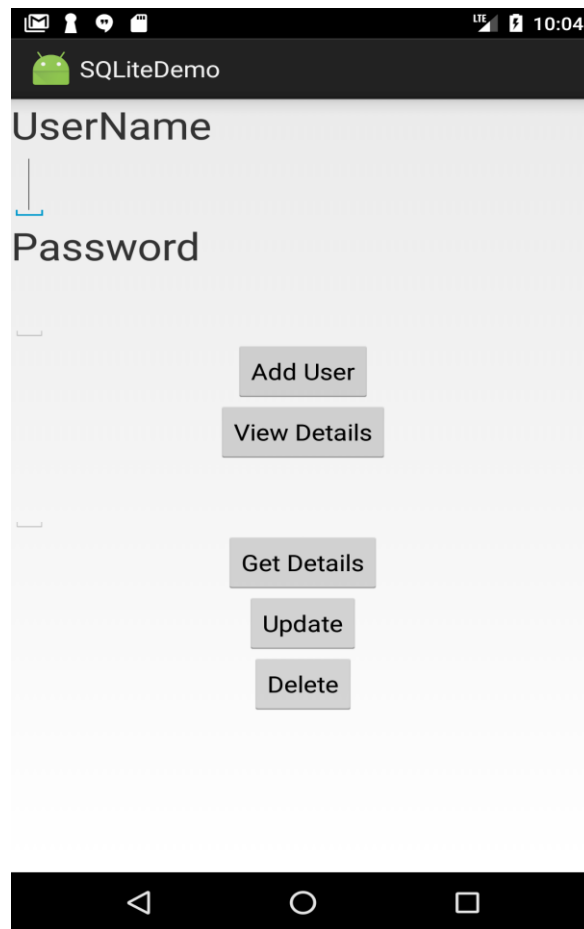


SQLiteDemo CRUD operations



activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context="com.example.girish.sqlitedemo.MainActivity">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="30dp"
        android:text="UserName" />
    <EditText
        android:id="@+id/userNameValue"
        android:textSize="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="30dp"
        android:text="Password" />
```

```

<EditText
    android:id="@+id/passwordValue"
    android:textSize="30dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:inputType="textPassword"/>
<Button
    android:id="@+id/button"
    android:layout_gravity="center"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:onClick="addUser"
    android:text="Add User" />
<Button
    android:id="@+id/button2"
    android:layout_gravity="center"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:onClick="viewDetails"
    android:text="View Details" />
<EditText
    android:id="@+id/name"
    android:textSize="30dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
<Button
    android:id="@+id/button3"
    android:layout_gravity="center"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:onClick="getDetails"
    android:text="Get Details" />
<Button
    android:id="@+id/button4"
    android:layout_gravity="center"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:onClick="update"
    android:text="Update" />
<Button
    android:id="@+id/button5"
    android:layout_gravity="center"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:onClick="delete"
    android:text="Delete" />
</LinearLayout>

```

Message.java

```

package com.example.girish.sqlitedemo;
/**
 * Created by girish on 13/8/17.
 */
import android.content.Context;
import android.widget.Toast;
public class Message {
    public static void message(Context context, String message) {
        Toast.makeText(context, message, Toast.LENGTH_LONG).show();
    }
}

```

DCSDatabaseAdapter.java

```

package com.example.girish.sqlitedemo;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.SQLException;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
/**
 * Created by girish on 13/8/17.
 */
public class DCSDatabaseAdapter {
    DCSHelper helper;
    public DCSDatabaseAdapter(Context context) {
        helper = new DCSHelper(context);
    }
    public long insertData(String name, String password) {
        SQLiteDatabase db = helper.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put(DCSHelper.NAME, name);
        contentValues.put(DCSHelper.PASSWORD, password);
        long id = db.insert(DCSHelper.TABLE_NAME, null, contentValues);
        return id;
    }
    public String getAllData() {
        SQLiteDatabase db = helper.getWritableDatabase();
        //SELECT _id,Name,Password from DCSTABLE;
        String[] columns = {DCSHelper.UID, DCSHelper.NAME, DCSHelper.PASSWORD};
        Cursor cursor = db.query(DCSHelper.TABLE_NAME, columns, null, null, null, null, null);
        StringBuffer buffer = new StringBuffer();
        while (cursor.moveToNext()) {
            int cid = cursor.getInt(0);
            String name = cursor.getString(1);
            String password = cursor.getString(2);
            buffer.append(cid + " " + name + " " + password + "\n");
        }
        return buffer.toString();
    }
    public String getData(String name, String password) {
        SQLiteDatabase db = helper.getWritableDatabase();
        //SELECT name, password from DCSTABLE WHERE NAME='Raj';
        //SELECT _id from DCSTABLE WHERE NAME=? AND password=?;
        String[] columns = {DCSHelper.UID, DCSHelper.NAME, DCSHelper.PASSWORD};
        String[] selectionArgs = {name, password};
        //Cursor cursor = db.query(DCSHelper.TABLE_NAME, columns, DCSHelper.NAME + "='"+name+"'", null, null,
null, null);
        Cursor cursor = db.query(DCSHelper.TABLE_NAME, columns, DCSHelper.NAME + "=? AND " +
DCSHelper.PASSWORD + "=?", selectionArgs, null, null, null);
        StringBuffer buffer = new StringBuffer();
        while (cursor.moveToNext()) {
            int index0 = cursor.getColumnIndex(DCSHelper.UID);
            int personId = cursor.getInt(index0);
            //int index1 = cursor.getColumnIndex(DCSHelper.NAME);
            //int index2 = cursor.getColumnIndex(DCSHelper.PASSWORD);
            //String password = cursor.getString(index2);
            buffer.append(personId + "\n");
            //buffer.append(name + " " + password + "\n");
        }
        return buffer.toString();
    }
    public int updateName(String oldName, String newName) {
        //UPDATE DCSTABLE SET Name='Raj from TY' WHERE Name='Raj';
        SQLiteDatabase db = helper.getWritableDatabase();
        ContentValues contentValues = new ContentValues();

```

```

        contentValues.put(DCSHelper.NAME, newName);
        String[] whereArgs = {oldName};
        int count = db.update(DCSHelper.TABLE_NAME, contentValues, DCSHelper.NAME + "=?", whereArgs);
        return count;
    }

    public int deleteRow() {
        //DELETE * FROM DCSTABLE WHERE Name='Girish';
        SQLiteDatabase db = helper.getWritableDatabase();
        String[] whereArgs = {"Girish"};
        int count = db.delete(DCSHelper.TABLE_NAME, DCSHelper.NAME + "=?", whereArgs);
        return count;
    }

    static class DCSHelper extends SQLiteOpenHelper {
        private static final String DATABASE_NAME = "dcsdatabase7";
        private static final String TABLE_NAME = "DCSTABLE";
        private static final int DATABASE_VERSION = 1;
        private static final String UID = "_id";
        private static final String NAME = "Name";
        private static final String PASSWORD = "Password";
        private static final String CREATE_TABLE = "CREATE TABLE " + TABLE_NAME + " (" + UID + "
INTEGER PRIMARY KEY AUTOINCREMENT, " + NAME + " VARCHAR(255), " + PASSWORD + "
VARCHAR(255));";
        private static final String DROP_TABLE = "DROP TABLE IF EXISTS " + TABLE_NAME + ";";
        private Context context;
        public DCSHelper(Context context) {
            super(context, DATABASE_NAME, null, DATABASE_VERSION);
            this.context = context;
            Message.message(context, "constructor called...");
        }
        @Override
        public void onCreate(SQLiteDatabase db) {
            try {
                db.execSQL(CREATE_TABLE);
                Message.message(context, "onCreate() called...");
            } catch (SQLException e) {
                Message.message(context, "" + e);
            }
        }
        @Override
        public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
            try {
                Message.message(context, "onUpgrade() called...");
                db.execSQL(DROP_TABLE);
                onCreate(db);
            } catch (SQLException e) {
                Message.message(context, "" + e);
            }
        }
    }
}

```

MainActivity.java

```

package com.example.girish.sqlitedemo;
import android.app.Activity;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
public class MainActivity extends Activity {
    EditText userName, password, name;
    DCSDatabaseAdapter dcshelper;
    @Override

```

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    userName = (EditText) findViewById(R.id.userNameValue);
    password = (EditText) findViewById(R.id.passwordValue);
    name = (EditText) findViewById(R.id.name);
    dcshelper = new DCSDatabaseAdapter(this);
}

public void addUser(View view) {
    String user = userName.getText().toString();
    String pass = password.getText().toString();
    long id = dcshelper.insertData(user, pass);
    if (id < 0)
        Message.message(this, "Unsuccessful...");
    else
        Message.message(this, "Row inserted successfully...");
}

public void viewDetails(View view) {
    String data = dcshelper.getAllData();
    Message.message(this, data);
}

/*public void getDetails(View view) {
    String s1 = name.getText().toString();
    String s2 = dcshelper.getData(s1);
    Message.message(this, s2);
}*/

public void getDetails(View view) {
    //Raj pass1
    String s1 = name.getText().toString();
    String sub1 = s1.substring(0, s1.indexOf(" "));
    String sub2 = s1.substring(s1.indexOf(" ") + 1);
    String s2 = dcshelper.getData(sub1, sub2);
    Message.message(this, s2);
}

public void update(View view) {
    dcshelper.updateName("Raj", "Raj from TYBSC");
}

public void delete(View view) {
    int count = dcshelper.deleteRow();
    Message.message(this, "" + count);
}
}

```