| **Name of Student:** Ajay Karthikesan | | | |
| --- | --- | --- | --- |
| **Roll Number:** 57 | | **Practical Number:** 4 | |
| **Aim of Practical:**  Android program to perform CRUD operation using SQLite DB | | | |
| **DOP:** 9.10.23 | | **DOS:** 31.10.23 | |
| **CO Mapped:** - | **PO Mapped:** - | **Faculty Signature:** | **Marks:** |

**Aim: Android program to perform CRUD operation using SQLite DB**

**Theory:**

**What is SQLite?**

SQLite is an SQL Database. I am assuming here that you are familiar with SQL databases. So in SQL database, we store data in tables. The tables are the structure of storing data consisting of rows and columns. We are not going in depth of what is an SQL database and how to work in SQL database. If you are going through this post, then you must know the [**Basics of SQL**.](https://www.w3schools.com/sql/)

**What is CRUD?**

As the heading tells you here, we are going to learn the CRUD operation in SQLite Database. **CRUD** is nothing but an abbreviation for the basic operations that we perform in any database. And the operations are

• **Create**

• **Read**

• **Update**

• **Delete**

**McaDept.java:**

package com.example.sqlite;

public class McaDept {

int id;

String profName;

String profJobRole;

int experience;

String expertise;

public McaDept() {

super();

}

public McaDept(int id) {

super();

this.id = id;

}

public McaDept(int id, String profName, String profJobRole, int experience, String expertise) {

super();

this.id = id;

this.profName = profName;

this.profJobRole = profJobRole;

this.experience = experience;

this.expertise = expertise;

}

public McaDept(String profName, String profJobRole, int experience, String expertise) {

super();

this.profName = profName;

this.profJobRole = profJobRole;

this.experience = experience;

this.expertise = expertise;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getProfName() {

return profName;

}

public void setProfName(String profName) {

this.profName = profName;

}

public String getProfJobRole() {

return profJobRole;

}

public void setProfJobRole(String profJobRole) {

this.profJobRole = profJobRole;

}

public int getExperience() {

return experience;

}

public void setExperience(int experience) {

this.experience = experience;

}

public String getExpertise() {

return expertise;

}

public void setExpertise(String expertise) {

this.expertise = expertise;

}

}

**SQLiteDatabaseHandler.java:**

package com.example.sqlite;

import android.content.ContentValues;

import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.database.sqlite.SQLiteOpenHelper;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.List;

public class SQLiteDatabaseHandler extends SQLiteOpenHelper {

// All Static variables

// Database Version

private static final int DATABASE\_VERSION = 1;

// Database Name

private static final String DATABASE\_NAME = "mcaDeptData";

// TABLE\_PROF table name

private static final String TABLE\_PROF= "profData";

// TABLE\_PROF Table Columns names

private static final String KEY\_ID = "id";

private static final String PROF\_NAME = "profName";

private static final String PROF\_JOB\_ROLE = "profRole";

private static final String EXPERIENCE = "experience";

private static final String EXPERTISE = "expertise";

public SQLiteDatabaseHandler(Context context) {

super(context, DATABASE\_NAME, null, DATABASE\_VERSION);

}

// Creating Tables

@Override

public void onCreate(SQLiteDatabase sqLiteDatabase) {

String CREATE\_TABLE = "CREATE TABLE " + TABLE\_PROF + "("

+ KEY\_ID + " INTEGER PRIMARY KEY," + PROF\_NAME + " TEXT," + PROF\_JOB\_ROLE + " TEXT,"

+ EXPERIENCE + " INT," + EXPERTISE + " TEXT" + ")";

sqLiteDatabase.execSQL(CREATE\_TABLE);

}

@Override

public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {

// Drop older table if existed

sqLiteDatabase.execSQL("DROP TABLE IF EXISTS " + TABLE\_PROF);

// Create tables again

onCreate(sqLiteDatabase);

}

/\*\*

\* All CRUD(Create, Read, Update, Delete) Operations

\*/

// Adding new entry

void addProfInDB(McaDept mcaDept) {

SQLiteDatabase db = this.getWritableDatabase();

ContentValues values = new ContentValues();

values.put(PROF\_NAME, mcaDept.getProfName());

values.put(PROF\_JOB\_ROLE, mcaDept.getProfJobRole());

values.put(EXPERIENCE, mcaDept.getExperience());

values.put(EXPERTISE, mcaDept.getExpertise());

// Inserting Row

db.insert(TABLE\_PROF, null, values);

db.close(); // Closing database connection

}

// Getting single record

McaDept getSingleRecord(int id) {

SQLiteDatabase db = this.getReadableDatabase();

Cursor cursor = db.query(TABLE\_PROF, new String[] { KEY\_ID, PROF\_NAME, PROF\_JOB\_ROLE, EXPERIENCE, EXPERTISE}, KEY\_ID + "=?",

new String[] { String.valueOf(id) }, null, null, null, null);

if (cursor != null) {

cursor.moveToFirst();

}

McaDept mcaDept = new McaDept(cursor.getInt(0), cursor.getString(1), cursor.getString(2), cursor.getInt(3), cursor.getString(4));

return mcaDept;

}

// Getting All Records

public ArrayList<HashMap<String, String>> getAllRecords() {

List recordList = new ArrayList();

SQLiteDatabase db = this.getWritableDatabase();

// Select All Query

ArrayList<HashMap<String, String>> userList = new ArrayList<>();

String query = "SELECT \* FROM " + TABLE\_PROF;

Cursor cursor = db.rawQuery(query,null);

while (cursor.moveToNext()){

HashMap<String,String> user = new HashMap<>();

user.put("Name","Name: " + cursor.getString(1));

user.put("Job Role","Job Role: " + cursor.getString(2));

user.put("Experience", "Experience: " +String.valueOf(cursor.getInt(3)));

user.put("Expertise","Expertise: "+cursor.getString(4));

userList.add(user);

}

return userList;

}

// Updating single record

public int updateRecord(McaDept mcaDept) {

SQLiteDatabase db = this.getWritableDatabase();

ContentValues values = new ContentValues();

values.put(PROF\_NAME, mcaDept.getProfName());

values.put(PROF\_JOB\_ROLE, mcaDept.getProfJobRole());

values.put(EXPERIENCE, mcaDept.getExperience());

values.put(EXPERTISE, mcaDept.getExpertise());

// updating row

return db.update(TABLE\_PROF, values, KEY\_ID + " = ?", new String[] { String.valueOf(mcaDept.getId()) });

}

// Deleting single record

public void deleteRecord(McaDept mcaDept) {

SQLiteDatabase db = this.getWritableDatabase();

db.delete(TABLE\_PROF, KEY\_ID + " = ?", new String[] { String.valueOf(mcaDept.getId()) });

db.close();

}

// Deleting all records

public void deleteAllRecords() {

SQLiteDatabase db = this.getWritableDatabase();

db.delete(TABLE\_PROF,null,null);

db.close();

}

// Getting Records Count

public int getRecordsCount() {

String countQuery = "SELECT \* FROM " + TABLE\_PROF;

SQLiteDatabase db = this.getReadableDatabase();

Cursor cursor = db.rawQuery(countQuery, null);

cursor.close();

// return count

return cursor.getCount();

}

}

**activity\_main.xml:**

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

<androidx.constraintlayout.widget.ConstraintLayout 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"

tools:context=".MainActivity">

<ListView

android:id="@+id/listview"

android:layout\_width="405dp"

android:layout\_height="656dp"

android:layout\_marginTop="1dp"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.428"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent">

</ListView>

<com.google.android.material.floatingactionbutton.FloatingActionButton

android:id="@+id/floatingActionButton"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="339dp"

android:layout\_marginEnd="16dp"

android:layout\_marginBottom="16dp"

android:clickable="true"

android:contentDescription="@string/app\_name"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:srcCompat="@android:drawable/ic\_input\_add" />

</androidx.constraintlayout.widget.ConstraintLayout>

**MainActivity.java:**

package com.example.sqlite;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.database.sqlite.SQLiteDatabase;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.ListAdapter;

import android.widget.ListView;

import android.widget.SimpleAdapter;

import com.google.android.material.floatingactionbutton.FloatingActionButton;

import java.util.ArrayList;

import java.util.HashMap;

public class MainActivity extends AppCompatActivity {

FloatingActionButton fab;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

fab = (FloatingActionButton) findViewById(R.id.floatingActionButton);

SQLiteDatabaseHandler db = new SQLiteDatabaseHandler(this);

ArrayList<HashMap<String, String>> userList = (ArrayList<HashMap<String, String>>) db.getAllRecords();

ListView lv = (ListView) findViewById(R.id.listview);

ListAdapter adapter = new SimpleAdapter(this, userList, R.layout.list\_row,new String[]{"Name", "Job Role", "Experience", "Expertise"}, new int[]{R.id.profNameTV, R.id.profJobRoleTV, R.id.experienceTV, R.id.expertiseTV});

lv.setAdapter(adapter);

lv.setOnItemClickListener(new AdapterView.OnItemClickListener() {

@Override

public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {

McaDept stu = db.getSingleRecord(i+1);

Intent i1 = new Intent(getApplicationContext(),EditRecordActivity.class);

i1.putExtra("id",stu.getId());

i1.putExtra("profName",stu.getProfName());

i1.putExtra("profJobRole",stu.getProfJobRole());

i1.putExtra("experience",stu.getExperience());

i1.putExtra("expertise",stu.getExpertise());

startActivity(i1);

}

});

fab.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(getApplicationContext(), InsertRecordsActivity.class);

startActivity(intent);

}

});

}

}

**list\_row.xml:**

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

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:padding="5dip" >

<TextView

android:id="@+id/profNameTV"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textStyle="bold"

android:textSize="17dp" />

<TextView

android:id="@+id/profJobRoleTV"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@id/profNameTV"

android:textStyle="bold"

android:textSize="17dp" />

<TextView

android:id="@+id/experienceTV"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@id/profJobRoleTV"

android:textStyle="bold"

android:textSize="17dp" />

<TextView

android:id="@+id/expertiseTV"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@id/experienceTV"

android:textStyle="bold"

android:textSize="17dp" />

</RelativeLayout>

**activity\_insert\_records.xml:**

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

<ScrollView

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\_height="match\_parent"

android:layout\_width="match\_parent"

android:fillViewport="true"

android:orientation="vertical"

tools:context=".InsertRecordsActivity">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

tools:context=".InsertRecordsActivity">

<TextView

android:id="@+id/textView"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:gravity="center"

android:text="MCA Department Record "

android:textColor="@color/black"

android:textSize="24sp"

android:textStyle="bold" />

<ImageView

android:id="@+id/imageView"

android:layout\_width="match\_parent"

android:layout\_height="350px"

android:layout\_gravity="center"

app:srcCompat="@drawable/vesitlogo" />

<EditText

android:id="@+id/name"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="50px"

android:layout\_marginRight="50px"

android:ems="10"

android:hint="Name of Professor"

android:inputType="textPersonName"

android:minHeight="48dp" />

<EditText

android:id="@+id/role"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="50px"

android:layout\_marginRight="50px"

android:ems="10"

android:inputType="textPersonName"

android:minHeight="48dp"

android:hint="Job Role of Professor" />

<EditText

android:id="@+id/exp"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="50px"

android:layout\_marginRight="50px"

android:ems="10"

android:hint="Experience of Professor"

android:inputType="textPersonName"

android:minHeight="48dp" />

<EditText

android:id="@+id/expertise"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="50px"

android:layout\_marginRight="50px"

android:ems="10"

android:inputType="textPersonName"

android:minHeight="48dp"

android:hint="Expertise of Professor" />

<Button

android:id="@+id/button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="50px"

android:layout\_marginRight="50px"

android:layout\_gravity="center"

android:gravity="center"

android:text="INSERT RECORD" />

<Button

android:id="@+id/back"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="50px"

android:layout\_marginRight="50px"

android:layout\_gravity="center"

android:gravity="center"

android:text="VIEW RECORDS" />

<TextView

android:id="@+id/textView3"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:gravity="center"

android:textColor="@color/black"

android:textSize="18sp"

android:textStyle="bold" />

<FrameLayout

android:layout\_width="match\_parent"

android:layout\_height="0dp"

android:layout\_weight="1">

</FrameLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal">

<TextView

android:id="@+id/textView2"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:gravity="center"

android:text="Developed by Ajay Karthikesan"

android:textColor="@color/black"

android:textSize="18sp"

android:textStyle="bold" />

</LinearLayout>

</LinearLayout>

</ScrollView>

**InsertRecordsActivity.java:**

package com.example.sqlite;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

public class InsertRecordsActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_insert\_records);

final EditText ETprofName, ETprofRole, ETexperience, ETexpertise;

final Button submit, back;

final TextView status;

SQLiteDatabaseHandler db= new SQLiteDatabaseHandler(this);

ETprofName = findViewById(R.id.name);

ETprofRole = findViewById(R.id.role);

ETexperience = findViewById(R.id.exp);

ETexpertise = findViewById(R.id.expertise);

submit = findViewById(R.id.button);

back = findViewById(R.id.back);

status = findViewById(R.id.textView3);

back.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Intent i = new Intent(getApplicationContext(), MainActivity.class);

startActivity(i);

finish();

}

});

submit.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

final String name = ETprofName.getText().toString();

final String role = ETprofRole.getText().toString();

final Integer exp = Integer.parseInt(ETexperience.getText().toString());

final String expertise = ETexpertise.getText().toString();

if (name.isEmpty())

{

ETprofName.setError("This field can't be empty!");

}

if (role.isEmpty())

{

ETprofRole.setError("This field can't be empty!");

}

if (exp==null)

{

ETexperience.setError("This field can't be empty!");

}

if (expertise.isEmpty())

{

ETexpertise.setError("This field can't be empty!");

}

if(!name.isEmpty() && !role.isEmpty() && exp!=null && !expertise.isEmpty() ) {

McaDept mcaDept = new McaDept(name, role, exp, expertise);

db.addProfInDB(mcaDept);

status.setText("Record added sucessfully!");

}

}

});

}

}

**activity\_edit\_record.xml:**

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

<ScrollView

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\_height="match\_parent"

android:layout\_width="match\_parent"

android:fillViewport="true"

android:orientation="vertical"

tools:context=".EditRecordActivity">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

tools:context=".InsertRecordsActivity">

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:gravity="center"

android:text="MCA Department Record "

android:textColor="@color/black"

android:textSize="24sp"

android:textStyle="bold" />

<ImageView

android:id="@+id/imageView1"

android:layout\_width="match\_parent"

android:layout\_height="350px"

android:layout\_gravity="center"

app:srcCompat="@drawable/vesitlogo" />

<EditText

android:id="@+id/uname"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="50px"

android:layout\_marginRight="50px"

android:ems="10"

android:hint="Name of Professor"

android:inputType="textPersonName"

android:minHeight="48dp" />

<EditText

android:id="@+id/urole"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="50px"

android:layout\_marginRight="50px"

android:ems="10"

android:inputType="textPersonName"

android:minHeight="48dp"

android:hint="Job Role of Professor" />

<EditText

android:id="@+id/uexp"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="50px"

android:layout\_marginRight="50px"

android:ems="10"

android:hint="Experience of Professor"

android:inputType="textPersonName"

android:minHeight="48dp" />

<EditText

android:id="@+id/uexpertise"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="50px"

android:layout\_marginRight="50px"

android:ems="10"

android:inputType="textPersonName"

android:minHeight="48dp"

android:hint="Expertise of Professor" />

<Button

android:id="@+id/ubutton"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="50px"

android:layout\_marginRight="50px"

android:layout\_gravity="center"

android:gravity="center"

android:text="UPDATE RECORD" />

<Button

android:id="@+id/dbutton"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="50px"

android:layout\_marginRight="50px"

android:layout\_gravity="center"

android:gravity="center"

android:text="DELETE RECORD" />

<Button

android:id="@+id/uback"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="50px"

android:layout\_marginRight="50px"

android:layout\_gravity="center"

android:gravity="center"

android:text="VIEW RECORDS" />

<TextView

android:id="@+id/textView31"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:gravity="center"

android:textColor="@color/black"

android:textSize="18sp"

android:textStyle="bold" />

<FrameLayout

android:layout\_width="match\_parent"

android:layout\_height="0dp"

android:layout\_weight="1">

</FrameLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal">

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:gravity="center"

android:text="Developed by Ajay Karthikesan"

android:textColor="@color/black"

android:textSize="18sp"

android:textStyle="bold" />

</LinearLayout>

</LinearLayout>

</ScrollView>

**EditRecordActivity.java:**

package com.example.sqlite;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

public class EditRecordActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_edit\_record);

final EditText ETprofName, ETprofRole, ETexperience, ETexpertise;

final Button update, delete, back;

final TextView status;

SQLiteDatabaseHandler db= new SQLiteDatabaseHandler(this);

ETprofName = findViewById(R.id.uname);

ETprofRole = findViewById(R.id.urole);

ETexperience = findViewById(R.id.uexp);

ETexpertise = findViewById(R.id.uexpertise);

update = findViewById(R.id.ubutton);

delete = findViewById(R.id.dbutton);

back = findViewById(R.id.uback);

status = findViewById(R.id.textView31);

Intent i = getIntent();

int t1 = i.getIntExtra("id",0);

String t2 = i.getStringExtra("profName").toString();

String t3 = i.getStringExtra("profJobRole").toString();

int t4 = i.getIntExtra("experience",0);

String t5 = i.getStringExtra("expertise").toString();

ETprofName.setText(t2);

ETprofRole.setText(t3);

ETexperience.setText(""+t4);

ETexpertise.setText(t5);

back.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Intent i = new Intent(getApplicationContext(), MainActivity.class);

startActivity(i);

finish();

}

});

update.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

final String name = ETprofName.getText().toString();

final String role = ETprofRole.getText().toString();

final Integer exp = Integer.parseInt(ETexperience.getText().toString());

final String expertise = ETexpertise.getText().toString();

if (name.isEmpty())

{

ETprofName.setError("This field can't be empty!");

}

if (role.isEmpty())

{

ETprofRole.setError("This field can't be empty!");

}

if (exp==null)

{

ETexperience.setError("This field can't be empty!");

}

if (expertise.isEmpty())

{

ETexpertise.setError("This field can't be empty!");

}

if(!name.isEmpty() && !role.isEmpty() && exp!=null && !expertise.isEmpty() ) {

McaDept mcaDept = new McaDept(t1, name, role, exp, expertise);

db.updateRecord(mcaDept);

status.setText("Record updated sucessfully!");

}

}

});

delete.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

McaDept mcaDept = new McaDept(t1);

db.deleteRecord(mcaDept);

status.setText("Record deleted sucessfully!");

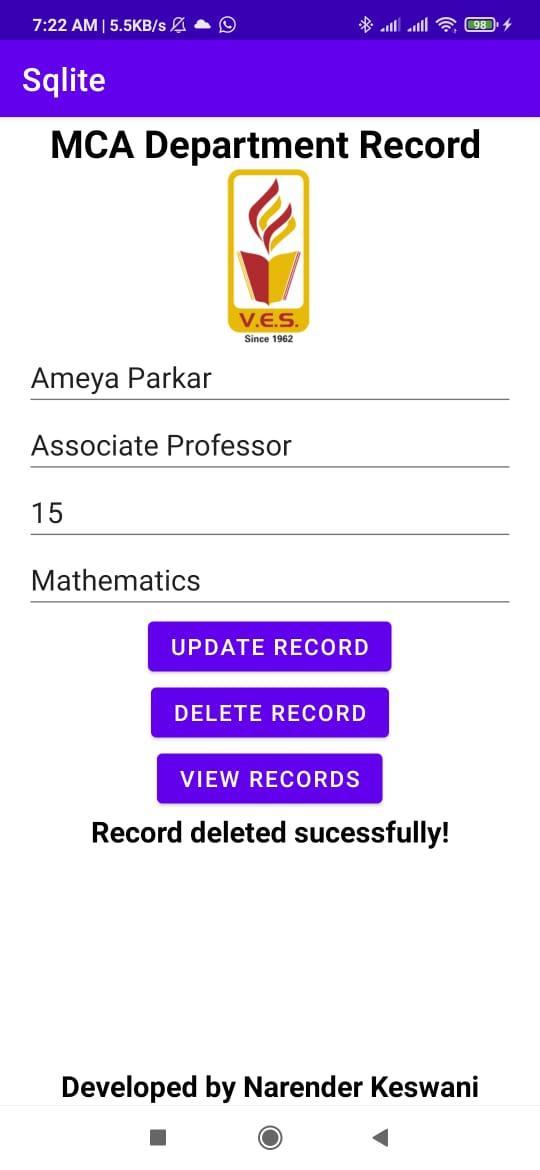
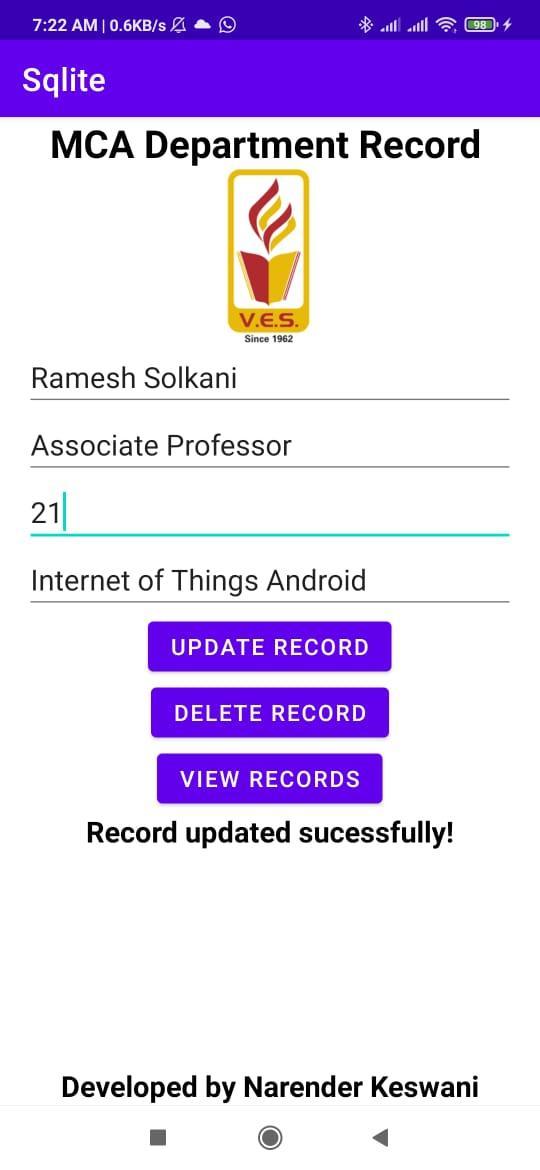
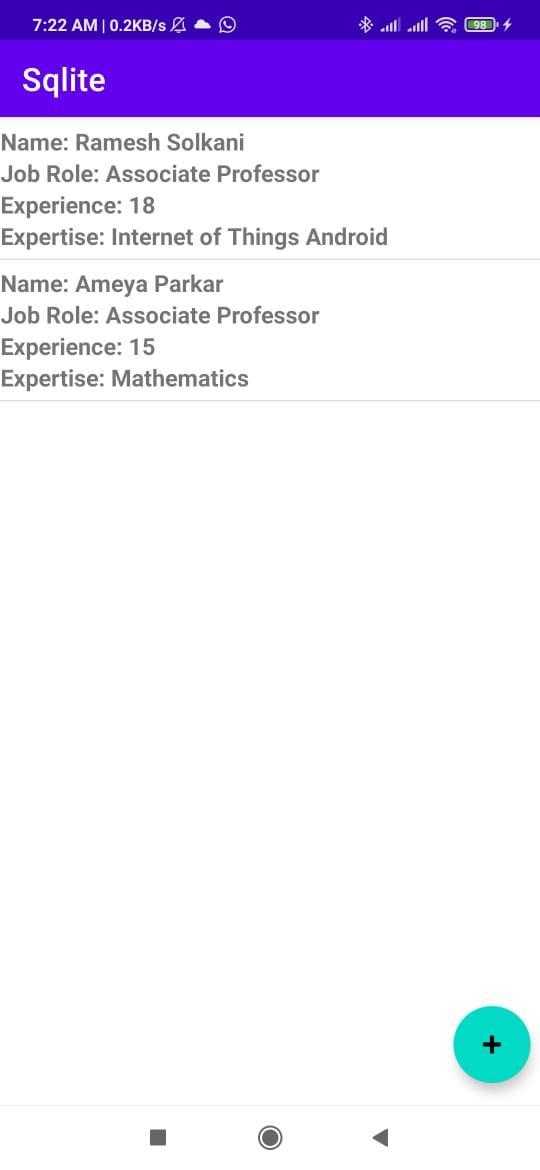
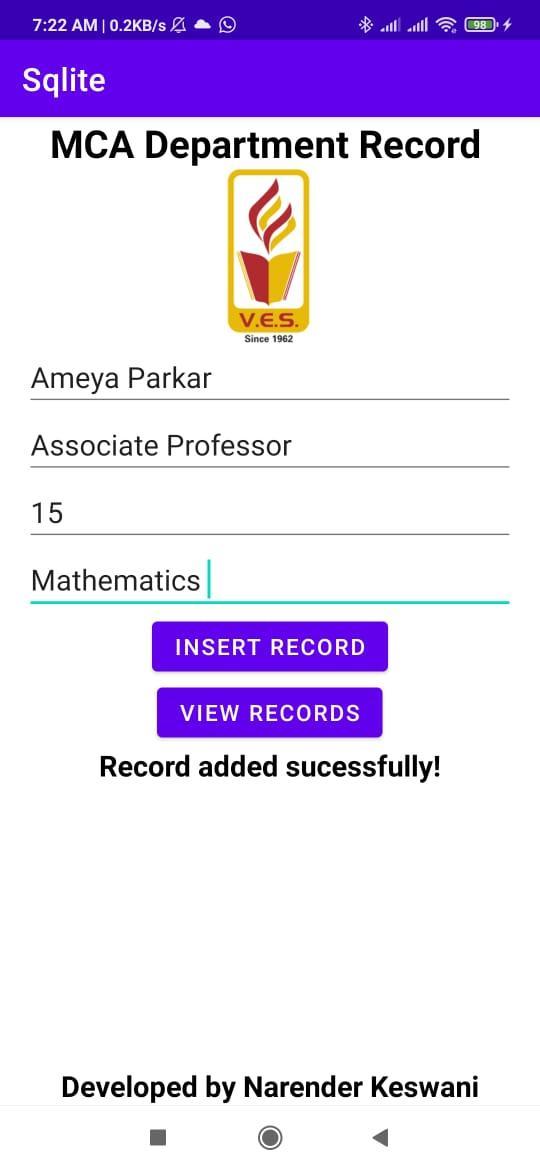
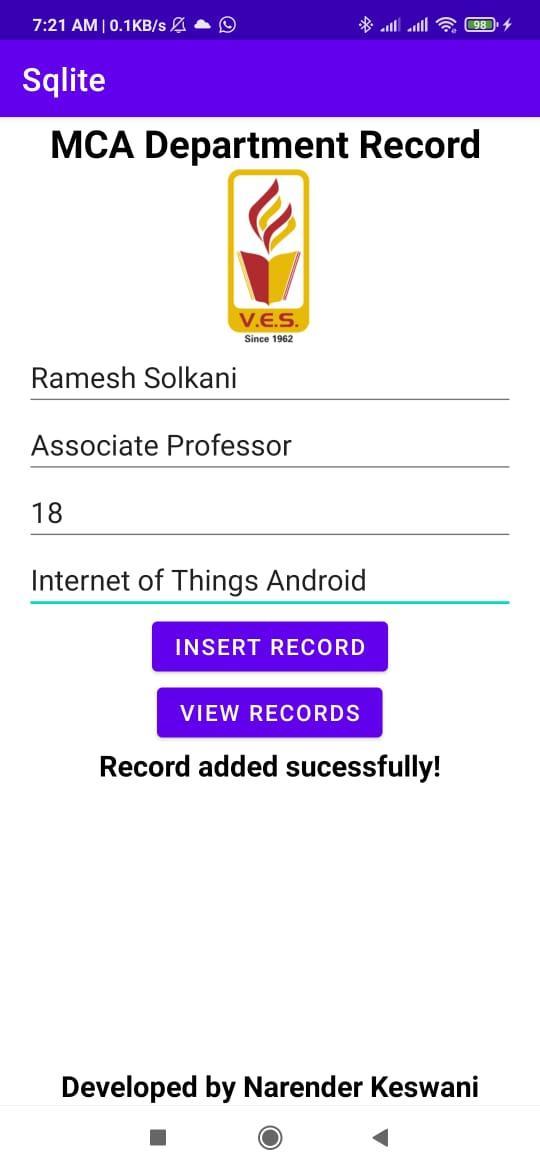
}

});

}

}

**OUTPUT:**

****

**CONCLUSION:**

From this practical, I have learned & implemented SQLite CRUD Operations in Android.