

EX:NO:05

SQLITE

ROLL.NO:210701290

DATE:08/03/2024

AIM:-

Create a Database table with the following structure using SQLite: Student (Register Number, Name, CGPA). Develop an android application to perform the following operation using SQLite developer classes. 1. Insert student Details 2. Update the student Record 3. Delete a specified record. 4. View the details.

PROCEDURE:-

Step 1: Create activities.

Step 2: Define UI.

Step 3: Implement Functionalty.

Step 4: Handle Navigation.

Step 5: Create a database helper classes.

Step 6: Define database schema.

Step 7: Implement CRUD operations.

Step 8: Use database in Activities.

PROGRAM CODE:-

AndroidManifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.studentdatabase">
    <application
        android:allowBackup="true"
```

```
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp"
    tools:context=".MainActivity">
    <!-- UI elements for inserting student details -->
    <EditText
```

```
android:id="@+id/registerNumberEditText"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Register Number"
android:inputType="number" />
```

```
<EditText
```

```
android:id="@+id/nameEditText"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_below="@id/registerNumberEditText"
android:layout_marginTop="16dp"
android:hint="Name" />
```

```
<EditText
```

```
android:id="@+id/cgpaEditText"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_below="@id/nameEditText"
android:layout_marginTop="16dp"
android:hint="CGPA"
android:inputType="numberDecimal" />
```

```
<Button
```

```
android:id="@+id/insertButton"
android:layout_width="wrap_content"
```

```

        android:layout_height="wrap_content"

        android:layout_below="@id/cgpaEditText"

        android:layout_centerHorizontal="true"

        android:layout_marginTop="16dp"

        android:text="Insert" />

<!-- UI elements for viewing student details -->

<ListView

        android:id="@+id/studentListView"

        android:layout_width="match_parent"

        android:layout_height="wrap_content"

        android:layout_below="@id/insertButton"

        android:layout_marginTop="16dp" />

</RelativeLayout>

```

MainActivity.kt:

```

package com.example.studentdatabase

import android.os.Bundle
import android.widget.ArrayAdapter
import android.widget.Button
import android.widget.EditText
import android.widget.ListView
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

    private lateinit var registerNumberEditText: EditText

```

```
private lateinit var nameEditText: EditText
private lateinit var cgpaEditText: EditText
private lateinit var insertButton: Button
private lateinit var studentListView: ListView
private lateinit var dbHelper: UserDBHelper
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_main)
    registerNumberEditText = findViewById(R.id.registerNumberEditText)
    nameEditText = findViewById(R.id.nameEditText)
    cgpaEditText = findViewById(R.id.cgpaEditText)
    insertButton = findViewById(R.id.insertButton)
    studentListView = findViewById(R.id.studentListView)
    dbHelper = UserDBHelper(this)
    insertButton.setOnClickListener {
        val registerNumber = registerNumberEditText.text.toString().toInt()
        val name = nameEditText.text.toString()
        val cgpa = cgpaEditText.text.toString().toDouble()
        val user = UserModel(registerNumber, name, cgpa)
        dbHelper.insertData(user)
        displayStudents()
    }
    displayStudents()
}
```

```
    }  
    private fun displayStudents() {  
        val users = dbHelper.getAllData()  
        val adapter = ArrayAdapter(this, android.R.layout.simple_list_item_1, users)  
        studentListView.adapter = adapter  
    }  
}
```

DBContract.kt:

```
package com.example.studentdatabase  
import android.provider.BaseColumns  
object DBContract {  
    class UserEntry : BaseColumns {  
        companion object {  
            const val TABLE_NAME = "Student"  
            const val COLUMN_REGISTER_NUMBER = "RegisterNumber"  
            const val COLUMN_NAME = "Name"  
            const val COLUMN_CGPA = "CGPA"  
        }  
    }  
}
```

UserModel.kt:

```
package com.example.studentdatabase

data class UserModel(val registerNumber: Int, val name: String, val cgpa: Double)
{
    override fun toString(): String {
        return "$registerNumber - $name - $cgpa"
    }
}
```

UserDBHelper.kt:

```
package com.example.studentdatabase

import android.content.ContentValues
import android.content.Context
import android.database.sqlite.SQLiteDatabase
import android.database.sqlite.SQLiteOpenHelper

class UserDBHelper(context: Context) :
    SQLiteOpenHelper(context, DATABASE_NAME, null,
        DATABASE_VERSION) {

    override fun onCreate(db: SQLiteDatabase) {

        val CREATE_TABLE =

            "CREATE TABLE ${DBContract.UserEntry.TABLE_NAME} (" +

                "${DBContract.UserEntry.COLUMN_REGISTER_NUMBER} " +
                INTEGER PRIMARY KEY," +

                "${DBContract.UserEntry.COLUMN_NAME} TEXT," +

                "${DBContract.UserEntry.COLUMN_CGPA} REAL)"
```

```

        db.execSQL(CREATE_TABLE)
    }

    override fun onUpgrade(db: SQLiteDatabase, oldVersion: Int, newVersion: Int)
    {
        db.execSQL("DROP TABLE IF EXISTS
${DBContract.UserEntry.TABLE_NAME}")

        onCreate(db)
    }

    fun insertData(user: UserModel) {
        val db = this.writableDatabase

        val values = ContentValues().apply {

            put(DBContract.UserEntry.COLUMN_REGISTER_NUMBER,
user.registerNumber)

            put(DBContract.UserEntry.COLUMN_NAME, user.name)

            put(DBContract.UserEntry.COLUMN_CGPA, user.cgpa)
        }

        db.insert(DBContract.UserEntry.TABLE_NAME, null, values)

        db.close()
    }

    fun getAllData(): List<UserModel> {
        val userList = mutableListOf<UserModel>()

        val db = this.readableDatabase

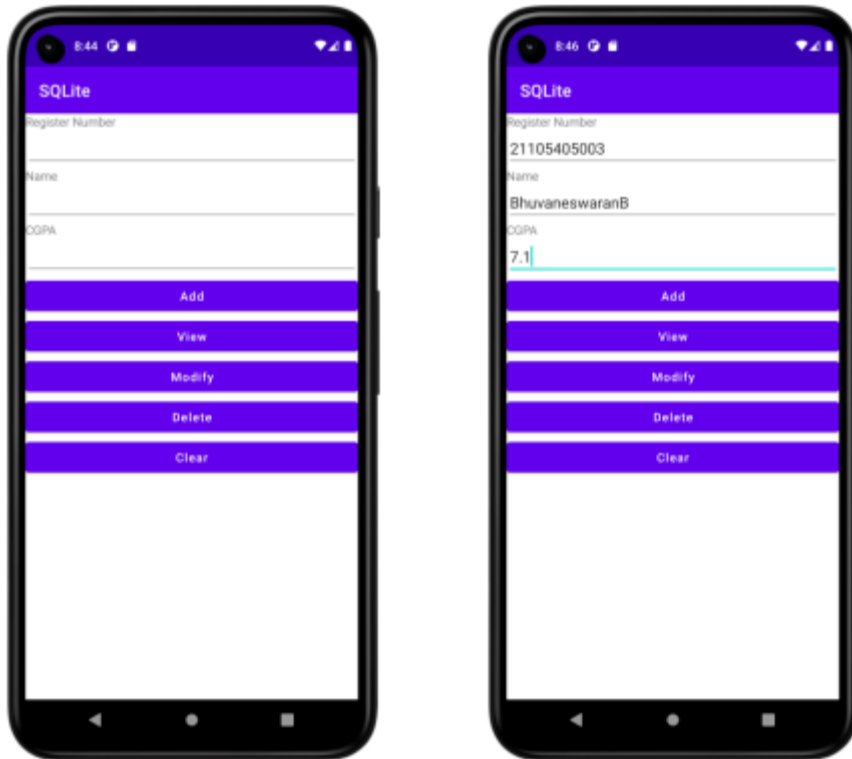
        val cursor = db.rawQuery("SELECT * FROM
${DBContract.UserEntry.TABLE_NAME}", null)

```



```
        if (cursor.moveToFirst()) {  
            do {  
                val registerNumber =  
cursor.getInt(cursor.getColumnIndex(DBContract.UserEntry.COLUMN_REGIST  
ER_NUMBER))  
  
                val name =  
cursor.getString(cursor.getColumnIndex(DBContract.UserEntry.COLUMN_NAM  
E))  
  
                val cgpa =  
cursor.getDouble(cursor.getColumnIndex(DBContract.UserEntry.COLUMN_CGP  
A))  
  
                userList.add(UserModel(registerNumber, name, cgpa))  
            } while (cursor.moveToNext())  
        }  
        cursor.close()  
        db.close()  
        return userList  
    }  
    companion object {  
        private const val DATABASE_VERSION = 1  
        private const val DATABASE_NAME = "StudentDatabase"  
    }  
}
```

OUTPUT:-



RESULT:-

Thus to develop an android application to perform the following operation using SQLite developer classes is implemented and executed successfully.