**Assignment-10**

**Aim:Create an Android application to implement the SQL Database storage system.**

**DbHelper.java file:**

package com.example.myapplication;

import android.content.ContentValues;

import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.database.sqlite.SQLiteOpenHelper;

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 users(id TEXT PRIMARY KEY, name TEXT, surname TEXT, marks TEXT)");

}

@Override

public void onUpgrade(SQLiteDatabase DB, int oldVersion, int newVersion) {

DB.execSQL("DROP TABLE IF EXISTS users");

onCreate(DB);

}

public Boolean InsertData(String id, String name, String surname, String marks) {

SQLiteDatabase DB = this.getWritableDatabase();

ContentValues contentValues = new ContentValues();

contentValues.put("id", id);

contentValues.put("name", name);

contentValues.put("surname", surname);

contentValues.put("marks", marks);

long res = DB.insert("users", null, contentValues);

return res != -1;

}

public Boolean UpdateData(String id, String name, String surname, String marks) {

SQLiteDatabase DB = this.getWritableDatabase();

ContentValues contentValues = new ContentValues();

contentValues.put("name", name);

contentValues.put("surname", surname);

contentValues.put("marks", marks);

Cursor cursor = DB.rawQuery("SELECT \* FROM users WHERE id=?", new String[]{id});

if (cursor.getCount() > 0) {

long res = DB.update("users", contentValues, "id=?", new String[]{id});

cursor.close(); // Close cursor to prevent memory leaks

return res != -1;

}

cursor.close();

return false;

}

public Boolean DeleteData(String id) {

SQLiteDatabase DB = this.getWritableDatabase();

Cursor cursor = DB.rawQuery("SELECT \* FROM users WHERE id=?", new String[]{id});

if (cursor.getCount() > 0) {

long result = DB.delete("users", "id=?", new String[]{id});

cursor.close();

return result != -1;

}

cursor.close();

return false;

}

public Cursor ViewData() {

SQLiteDatabase DB = this.getWritableDatabase();

return DB.rawQuery("SELECT \* FROM users", null);

}

}

**.java file code:**

package com.example.myapplication;

import android.database.Cursor;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import androidx.appcompat.app.AlertDialog;

import androidx.appcompat.app.AppCompatActivity;

public class SqlLite extends AppCompatActivity {

Button insert, update, view, delete;

EditText id, name, surname, marks;

DBHelper DB;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_sql\_lite);

// Initializing UI elements

insert = findViewById(R.id.insert);

update = findViewById(R.id.update);

view = findViewById(R.id.view);

delete = findViewById(R.id.delete);

id = findViewById(R.id.id);

name = findViewById(R.id.name);

surname = findViewById(R.id.surname);

marks = findViewById(R.id.marks);

DB = new DBHelper(this);

// Insert Data

insert.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String Id = id.getText().toString();

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

String Surname = surname.getText().toString();

String Marks = marks.getText().toString();

Boolean checkInsert = DB.InsertData(Id, Name, Surname, Marks);

if (checkInsert) {

Toast.makeText(SqlLite.this, "Data inserted successfully!", Toast.LENGTH\_SHORT).show();

} else {

Toast.makeText(SqlLite.this, "Cannot insert data", Toast.LENGTH\_SHORT).show();

}

}

});

// Update Data

update.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String Id = id.getText().toString();

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

String Surname = surname.getText().toString();

String Marks = marks.getText().toString();

Boolean checkUpdate = DB.UpdateData(Id, Name, Surname, Marks);

if (checkUpdate) {

Toast.makeText(SqlLite.this, "Data updated successfully", Toast.LENGTH\_SHORT).show();

} else {

Toast.makeText(SqlLite.this, "Unable to update data!", Toast.LENGTH\_SHORT).show();

}

}

});

// View Data

view.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Cursor res = DB.ViewData();

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

Toast.makeText(SqlLite.this, "No data exists", Toast.LENGTH\_SHORT).show();

return;

}

StringBuffer buffer = new StringBuffer();

while (res.moveToNext()) {

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

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

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

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

}

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

builder.setCancelable(true);

builder.setTitle("User Data");

builder.setMessage(buffer.toString());

builder.show();

}

});

// Delete Data

delete.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String Id = id.getText().toString();

if (Id.equals("")) {

Toast.makeText(SqlLite.this, "Please enter ID", Toast.LENGTH\_SHORT).show();

} else {

Boolean checkDelete = DB.DeleteData(Id);

if (checkDelete) {

Toast.makeText(SqlLite.this, "Data deleted!", Toast.LENGTH\_SHORT).show();

} else {

Toast.makeText(SqlLite.this, "Unable to delete", Toast.LENGTH\_SHORT).show();

}

}

}

});

}

}

**.xml file code:**

<?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"

android:padding="20dp"

tools:context=".SqlLite">

<!-- ID Input -->

<EditText

android:id="@+id/id"

android:layout\_width="match\_parent"

android:layout\_height="50dp"

android:hint="Enter ID" />

<!-- Name Input -->

<EditText

android:id="@+id/name"

android:layout\_width="match\_parent"

android:layout\_height="50dp"

android:layout\_marginTop="10dp"

android:hint="Enter name" />

<!-- Surname Input -->

<EditText

android:id="@+id/surname"

android:layout\_width="match\_parent"

android:layout\_height="50dp"

android:layout\_marginTop="10dp"

android:hint="Enter surname" />

<!-- Marks Input -->

<EditText

android:id="@+id/marks"

android:layout\_width="match\_parent"

android:layout\_height="50dp"

android:layout\_marginTop="10dp"

android:hint="Enter marks" />

<!-- First Row of Buttons -->

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:gravity="center"

android:layout\_marginTop="20dp">

<Button

android:id="@+id/insert"

android:layout\_width="0dp"

android:layout\_height="50dp"

android:layout\_weight="1"

android:text="Insert"

android:layout\_margin="5dp"/>

<Button

android:id="@+id/view"

android:layout\_width="0dp"

android:layout\_height="50dp"

android:layout\_weight="1"

android:text="View"

android:layout\_margin="5dp"/>

</LinearLayout>

<!-- Second Row of Buttons -->

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:gravity="center"

android:layout\_marginTop="20dp">

<Button

android:id="@+id/update"

android:layout\_width="0dp"

android:layout\_height="50dp"

android:layout\_weight="1"

android:text="Update"

android:layout\_margin="5dp"/>

<Button

android:id="@+id/delete"

android:layout\_width="0dp"

android:layout\_height="50dp"

android:layout\_weight="1"

android:text="Delete"

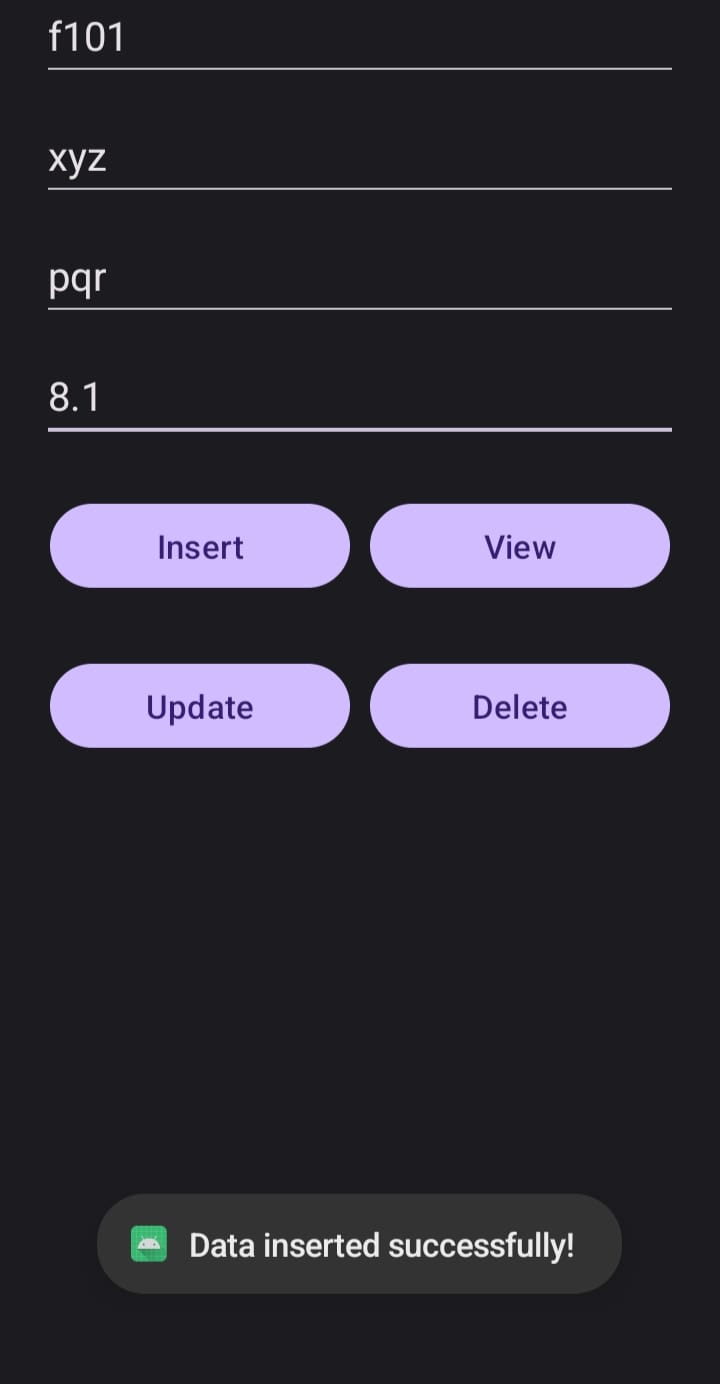
android:layout\_margin="5dp"/>

</LinearLayout>

</LinearLayout>

**OUTPUT:**

**insert:**

****