activitymain.xml

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

<RelativeLayout 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:padding="10dp" tools:context=".MainActivity">

<TextView android:id="@+id/texttitle"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:text="Please Enter the Details Below:" android:textSize="24dp" android:layout\_margin="20dp"

/>

<EditText android:id="@+id/name"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:inputType="textPersonName" android:hint="Name" android:textSize="24dp" android:layout\_below="@+id/texttitle"

/>

<EditText android:id="@+id/contact"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:inputType="number" android:hint="Contact" android:textSize="24dp" android:layout\_below="@+id/name"

/>

<EditText android:id="@+id/dob"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:inputType="number" android:hint="Date of Birth" android:textSize="24dp" android:layout\_below="@+id/contact"

/>

<Button

android:id="@+id/btnInsert" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:text="Insert New Data" android:textSize="24dp" android:layout\_marginTop="30dp" android:layout\_below="@+id/dob"

/>

<Button

android:id="@+id/btnUpdate" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:text="Update Existing Data" android:textSize="24dp" android:layout\_below="@+id/btnInsert"

/>

<Button

android:id="@+id/btnDelete" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:text="Delete Existing Data" android:textSize="24dp" android:layout\_below="@+id/btnUpdate"

/>

<Button

android:id="@+id/btnView" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:text="View Data" android:textSize="24dp" android:layout\_below="@+id/btnDelete"

/>

</RelativeLayout>

mainactivity.java

package com.example.myapplication;

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;

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);

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();

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

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();

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

if (checkupdatedata == true) Toast.makeText(MainActivity.this, "Entry Updated",

Toast.LENGTH\_SHORT).show(); else

Toast.makeText(MainActivity.this, "Entry Not Updated", Toast.LENGTH\_SHORT).show();

}

});

delete.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View view) {

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

Boolean checkdeletedata = DB.deletedata(nameTXT); if (checkdeletedata == 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 v) { 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("Date of Birth :" + res.getString(2) + "\n\n");

}

}

});

}

}

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

builder.setTitle("User Entries"); builder.setMessage(buffer.toString()); builder.show();

DBHelper.java

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;

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)");

}

@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) { SQLiteDatabase DB = this.getWritableDatabase();

ContentValues contentValues = new ContentValues(); contentValues.put("name", name); contentValues.put("contact", contact); contentValues.put("dob", dob);

long result = DB.insert("Userdetails", null, contentValues); if (result == -1) {

return false;

} else {

return true;

}

}

public Boolean updateuserdata(String name, String contact, String dob) {

SQLiteDatabase DB = this.getWritableDatabase(); ContentValues contentValues = new ContentValues();

contentValues.put("contact", contact); contentValues.put("dob", dob);

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;

}

}