Program 12

10-11-22

Design a login form with username and password using linear layout and toast valid credentials

*Activitymain 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=".MainActivity">  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Please enter the details below!"  
 android:textSize="30dp" />  
 <EditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="Name"  
 android:id="@+id/et1"/>  
 <EditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="Contact"  
 android:id="@+id/et2"  
 />  
 <EditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="Dob"  
 android:id="@+id/et3"/>  
 <Button  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Insert"  
 android:id="@+id/bt1"/>  
 <Button  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Update"  
 android:id="@+id/bt2"/>  
 <Button  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Delete"  
 android:id="@+id/bt3"  
 />  
 <Button  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="View"  
 android:id="@+id/bt4"/>  
  
  
</LinearLayout>

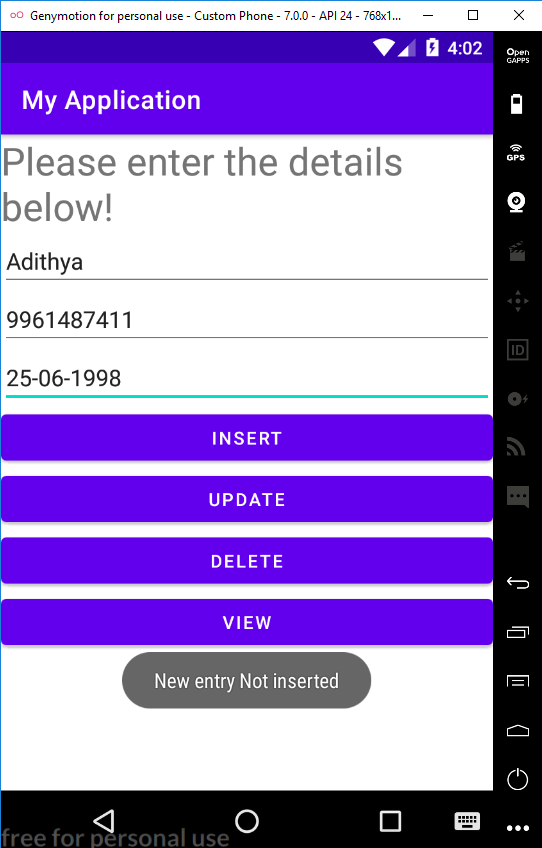
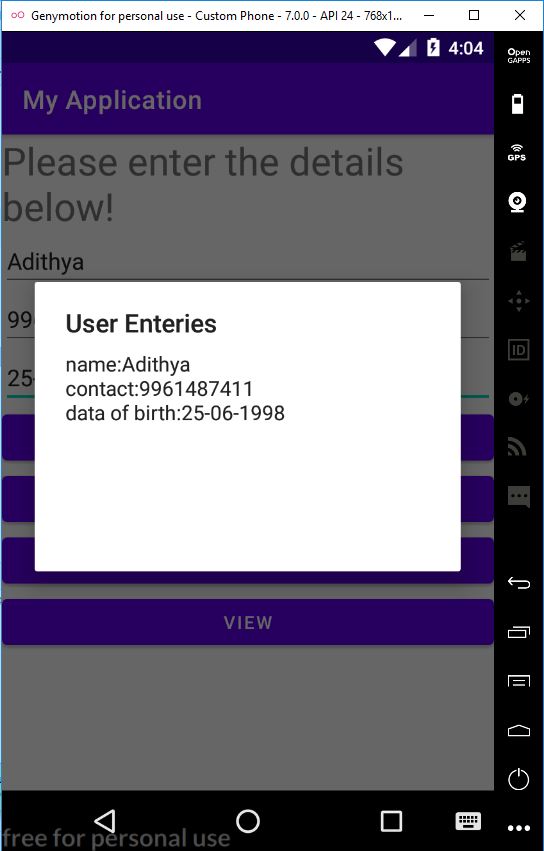
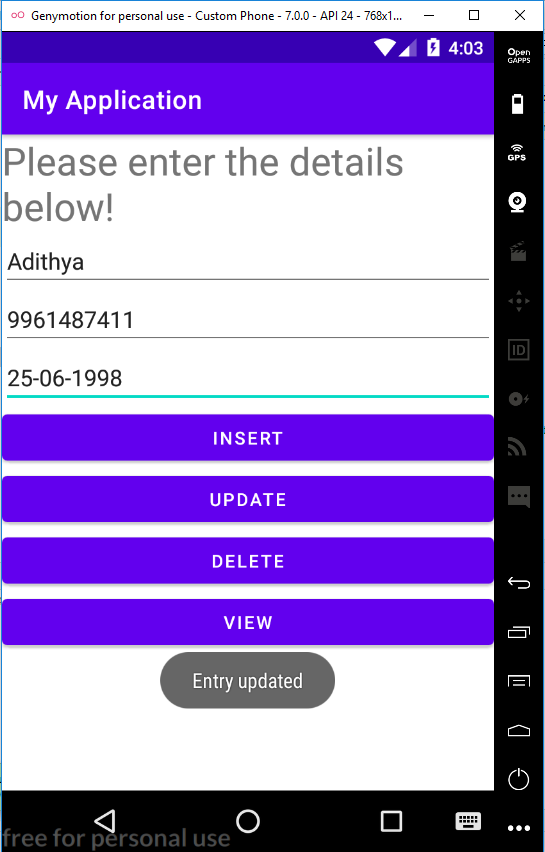
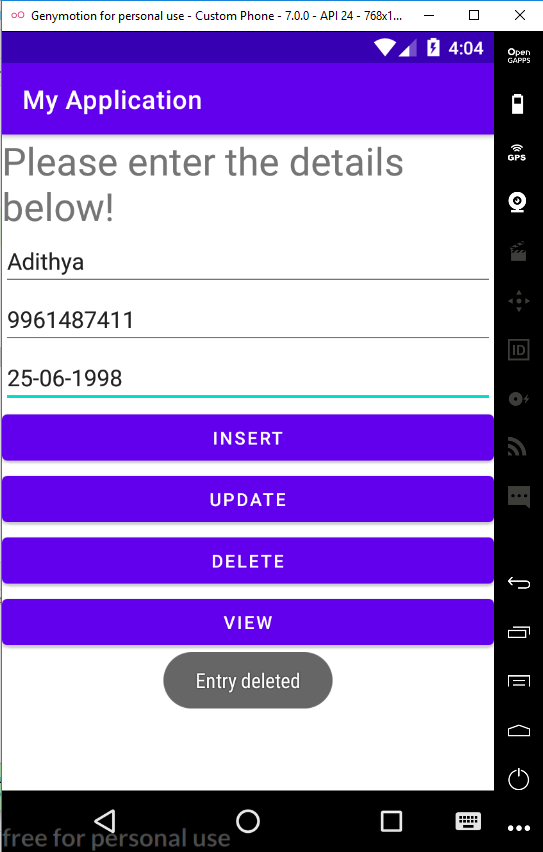
Main activity java

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 et1,et2,et3;  
 Button Insert,Update,Delete,View;  
 DBHelper db;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 et1=findViewById(R.id.*et1*);  
 et2=findViewById(R.id.*et2*);  
 et3=findViewById(R.id.*et3*);  
  
 Insert =findViewById(R.id.*bt1*);  
 Update = findViewById(R.id.*bt2*);  
 Delete = findViewById(R.id.*bt3*);  
 View =findViewById(R.id.*bt4*);  
 db= new DBHelper(this);  
  
  
 Insert.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(android.view.View view) {  
 String name= et1.getText().toString();  
 String contact=et2.getText().toString();  
 String dob= et3.getText().toString();  
  
 boolean checkInsertData= db.insertData(name,contact,dob);  
 if(checkInsertData==true)  
 Toast.*makeText*(MainActivity.this,"New entry inserted",Toast.*LENGTH\_LONG*).show();  
 else  
 Toast.*makeText*(MainActivity.this,"New entry Not inserted",Toast.*LENGTH\_LONG*).show();  
  
 }  
 });  
  
 Update.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(android.view.View view) {  
 String name= et1.getText().toString();  
 String contact=et2.getText().toString();  
 String dob= et3.getText().toString();  
  
 boolean checkUpdateData= db.updateData(name,contact,dob);  
 if(checkUpdateData==true)  
 Toast.*makeText*(MainActivity.this,"Entry updated",Toast.*LENGTH\_LONG*).show();  
 else  
 Toast.*makeText*(MainActivity.this,"Entry Not Updated",Toast.*LENGTH\_LONG*).show();  
  
 }  
 });  
 Delete.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(android.view.View view) {  
 String name= et1.getText().toString();  
  
  
 boolean checkDeleteData= db.deleteData(name);  
 if(checkDeleteData==true)  
 Toast.*makeText*(MainActivity.this,"Entry deleted",Toast.*LENGTH\_LONG*).show();  
 else  
 Toast.*makeText*(MainActivity.this,"Entry Not deleted",Toast.*LENGTH\_LONG*).show();  
  
 }  
 });  
 View.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(android.view.View view) {  
 Cursor res = db.getData();  
 if (res.getCount() == 0) {  
 Toast.*makeText*(MainActivity.this, "No entry exists", Toast.*LENGTH\_LONG*).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("data of birth:" + res.getString(2) + "\n\n\n\n\n");  
 }  
 AlertDialog.Builder builder=new AlertDialog.Builder(MainActivity.this);  
 builder.setCancelable(true);  
 builder.setTitle("User Enteries");  
 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(@Nullable 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 insertData(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 updateData(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;  
 }  
  
  
}

**OUTPUT**

** **