**SQLite Example**

**MainActivity.java**

package com.example.sqliteexample;  
  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.content.Intent;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Toast;  
  
public class MainActivity extends AppCompatActivity {  
 EditText name, loc, desig;  
 Button saveBtn;  
 Intent intent;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 name = (EditText)findViewById(R.id.*txtName*);  
 loc = (EditText)findViewById(R.id.*txtLocation*);  
 desig = (EditText)findViewById(R.id.*txtDesignation*);  
 saveBtn = (Button)findViewById(R.id.*btnSave*);  
 saveBtn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String username = name.getText().toString()+"\n";  
 String location = loc.getText().toString();  
 String designation = desig.getText().toString();  
 DbHandler dbHandler = new DbHandler(MainActivity.this);  
 dbHandler.insertUserDetails(username,location,designation);  
 intent = new Intent(MainActivity.this,DetailsActivity.class);  
 startActivity(intent);  
 Toast.*makeText*(getApplicationContext(), "Details Inserted Successfully",Toast.*LENGTH\_SHORT*).show();  
 }  
 });  
 }  
}

**DetailsActivity.java**

package com.example.sqliteexample;  
import android.content.Intent;  
import android.os.Bundle;  
import android.support.v7.app.AppCompatActivity;  
import android.view.View;  
import android.widget.Button;  
import android.widget.ListAdapter;  
import android.widget.ListView;  
import android.widget.SimpleAdapter;  
import java.util.ArrayList;  
import java.util.HashMap;  
  
public class DetailsActivity extends AppCompatActivity {  
 Intent intent;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*details*);  
 DbHandler db = new DbHandler(this);  
 ArrayList<HashMap<String, String>> userList = db.GetUsers();  
 ListView lv = (ListView) findViewById(R.id.*user\_list*);  
 ListAdapter adapter = new SimpleAdapter(DetailsActivity.this, userList, R.layout.*list\_row*,new String[]{"name","designation","location"}, new int[]{R.id.*name*, R.id.*designation*, R.id.*location*});  
 lv.setAdapter(adapter);  
 Button back = (Button)findViewById(R.id.*btnBack*);  
 back.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 intent = new Intent(DetailsActivity.this,MainActivity.class);  
 startActivity(intent);  
 }  
 });  
 }  
}

**DbHandler.java**

package com.example.sqliteexample;  
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;  
  
public class DbHandler extends SQLiteOpenHelper {  
 private static final int *DB\_VERSION* = 1;  
 private static final String *DB\_NAME* = "usersdb";  
 private static final String *TABLE\_Users* = "userdetails";  
 private static final String *KEY\_ID* = "id";  
 private static final String *KEY\_NAME* = "name";  
 private static final String *KEY\_LOC* = "location";  
 private static final String *KEY\_DESG* = "designation";  
 public DbHandler(Context context){  
 super(context,*DB\_NAME*, null, *DB\_VERSION*);  
 }  
 @Override  
 public void onCreate(SQLiteDatabase db){  
 String CREATE\_TABLE = "CREATE TABLE " + *TABLE\_Users* + "("  
 + *KEY\_ID* + " INTEGER PRIMARY KEY AUTOINCREMENT," + *KEY\_NAME* + " TEXT,"  
 + *KEY\_LOC* + " TEXT,"  
 + *KEY\_DESG* + " TEXT"+ ")";  
 db.execSQL(CREATE\_TABLE);  
 }  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion){  
 // Drop older table if exist  
 db.execSQL("DROP TABLE IF EXISTS " + *TABLE\_Users*);  
 // Create tables again  
 onCreate(db);  
 }  
 // \*\*\*\* CRUD (Create, Read, Update, Delete) Operations \*\*\*\*\* //  
  
 // Adding new User Details  
 void insertUserDetails(String name, String location, String designation){  
 //Get the Data Repository in write mode  
 SQLiteDatabase db = this.getWritableDatabase();  
 //Create a new map of values, where column names are the keys  
 ContentValues cValues = new ContentValues();  
 cValues.put(*KEY\_NAME*, name);  
 cValues.put(*KEY\_LOC*, location);  
 cValues.put(*KEY\_DESG*, designation);  
 // Insert the new row, returning the primary key value of the new row  
 long newRowId = db.insert(*TABLE\_Users*,null, cValues);  
 db.close();  
 }  
 // Get User Details  
 public ArrayList<HashMap<String, String>> GetUsers(){  
 SQLiteDatabase db = this.getWritableDatabase();  
 ArrayList<HashMap<String, String>> userList = new ArrayList<>();  
 String query = "SELECT name, location, designation FROM "+ *TABLE\_Users*;  
 Cursor cursor = db.rawQuery(query,null);  
 while (cursor.moveToNext()){  
 HashMap<String,String> user = new HashMap<>();  
 user.put("name",cursor.getString(cursor.getColumnIndex(*KEY\_NAME*)));  
 user.put("designation",cursor.getString(cursor.getColumnIndex(*KEY\_DESG*)));  
 user.put("location",cursor.getString(cursor.getColumnIndex(*KEY\_LOC*)));  
 userList.add(user);  
 }  
 return userList;  
 }  
 // Get User Details based on userid  
 public ArrayList<HashMap<String, String>> GetUserByUserId(int userid){  
 SQLiteDatabase db = this.getWritableDatabase();  
 ArrayList<HashMap<String, String>> userList = new ArrayList<>();  
 String query = "SELECT name, location, designation FROM "+ *TABLE\_Users*;  
 Cursor cursor = db.query(*TABLE\_Users*, new String[]{*KEY\_NAME*, *KEY\_LOC*, *KEY\_DESG*}, *KEY\_ID*+ "=?",new String[]{String.*valueOf*(userid)},null, null, null, null);  
 if (cursor.moveToNext()){  
 HashMap<String,String> user = new HashMap<>();  
 user.put("name",cursor.getString(cursor.getColumnIndex(*KEY\_NAME*)));  
 user.put("designation",cursor.getString(cursor.getColumnIndex(*KEY\_DESG*)));  
 user.put("location",cursor.getString(cursor.getColumnIndex(*KEY\_LOC*)));  
 userList.add(user);  
 }  
 return userList;  
 }  
 // Delete User Details  
 public void DeleteUser(int userid){  
 SQLiteDatabase db = this.getWritableDatabase();  
 db.delete(*TABLE\_Users*, *KEY\_ID*+" = ?",new String[]{String.*valueOf*(userid)});  
 db.close();  
 }  
 // Update User Details  
 public int UpdateUserDetails(String location, String designation, int id){  
 SQLiteDatabase db = this.getWritableDatabase();  
 ContentValues cVals = new ContentValues();  
 cVals.put(*KEY\_LOC*, location);  
 cVals.put(*KEY\_DESG*, designation);  
 int count = db.update(*TABLE\_Users*, cVals, *KEY\_ID*+" = ?",new String[]{String.*valueOf*(id)});  
 return count;  
 }  
}

**activity\_main.xml**

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:orientation="vertical" android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
 <TextView  
 android:id="@+id/fstTxt"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="100dp"  
 android:layout\_marginTop="150dp"  
 android:text="Name" />  
 <EditText  
 android:id="@+id/txtName"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="100dp"  
 android:ems="10"/>  
 <TextView  
 android:id="@+id/secTxt"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Location"  
 android:layout\_marginLeft="100dp" />  
 <EditText  
 android:id="@+id/txtLocation"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="100dp"  
 android:ems="10" />  
 <TextView  
 android:id="@+id/thirdTxt"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Designation"  
 android:layout\_marginLeft="100dp" />  
 <EditText  
 android:id="@+id/txtDesignation"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="100dp"  
 android:ems="10" />  
 <Button  
 android:id="@+id/btnSave"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="100dp"  
 android:text="Save" />  
</LinearLayout>

**Details.xml**

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="fill\_parent"  
 android:orientation="vertical" >  
 <ListView  
 android:id="@+id/user\_list"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:dividerHeight="1dp" />  
 <Button  
 android:id="@+id/btnBack"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:layout\_marginTop="20dp"  
 android:text="Back" />  
</LinearLayout>

**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/name"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textStyle="bold"  
 android:textSize="17dp" />  
 <TextView  
 android:id="@+id/designation"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/name"  
 android:layout\_marginTop="7dp"  
 android:textColor="#343434"  
 android:textSize="14dp" />  
 <TextView  
 android:id="@+id/location"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignBaseline="@+id/designation"  
 android:layout\_alignBottom="@+id/designation"  
 android:layout\_alignParentRight="true"  
 android:textColor="#343434"  
 android:textSize="14dp" />  
</RelativeLayout>