**Practical No.05**

**Aim** : Program to illustrate Content Provider

**Performed By** : Yogesh Gavande

**Class :** BCA-III SEM-V

**Date :** 08/10/2024

**Code:**

**Main\_Activity.java**

package com.example.contentproviderexample;  
import android.content.ContentValues;  
import android.database.Cursor;  
import android.net.Uri;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MainActivity extends AppCompatActivity {  
 private EditText nameInput, phoneInput;  
 private TextView resultView;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 nameInput = findViewById(R.id.nameInput);  
 phoneInput = findViewById(R.id.phoneInput);  
 resultView = findViewById(R.id.resultView);  
  
 Button addButton = findViewById(R.id.addButton);  
 Button queryButton = findViewById(R.id.queryButton);  
  
 addButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 addContact();  
 }  
 });  
  
 queryButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 queryContacts();  
 }  
 });  
 }  
  
 private void addContact() {  
 ContentValues values = new ContentValues();  
 values.put(ContactProvider.NAME, nameInput.getText().toString());  
 values.put(ContactProvider.PHONE, phoneInput.getText().toString());  
  
 Uri uri = getContentResolver().insert(ContactProvider.CONTENT\_URI, values);  
 if (uri != null) {  
 resultView.setText("Contact added successfully");  
 } else {  
 resultView.setText("Failed to add contact");  
 }  
 }  
  
 private void queryContacts() {  
 StringBuilder result = new StringBuilder();  
 Cursor cursor = getContentResolver().query(ContactProvider.CONTENT\_URI, null, null, null, null);  
  
 if (cursor != null && cursor.moveToFirst()) {  
 do {  
 String name = cursor.getString(cursor.getColumnIndex(ContactProvider.NAME));  
 String phone = cursor.getString(cursor.getColumnIndex(ContactProvider.PHONE));  
 result.append("Name: ").append(name).append(", Phone: ").append(phone).append("\n");  
 } while (cursor.moveToNext());  
 cursor.close();  
 }  
  
 resultView.setText(result.toString());  
 }  
}

**ContentProvider.java**

package com.example.contentproviderexample;  
import android.content.ContentProvider;  
import android.content.ContentUris;  
import android.content.ContentValues;  
import android.content.Context;  
import android.content.UriMatcher;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
import android.net.Uri;  
  
public class ContactProvider extends ContentProvider {  
 public static final String AUTHORITY = "com.example.contentproviderexample.provider";  
 public static final Uri CONTENT\_URI = Uri.parse("content://" + AUTHORITY + "/contacts");  
  
 public static final String NAME = "name";  
 public static final String PHONE = "phone";  
  
 private static final int CONTACTS = 1;  
 private static final int CONTACT\_ID = 2;  
  
 private static final UriMatcher uriMatcher = new UriMatcher(UriMatcher.NO\_MATCH);  
  
 static {  
 uriMatcher.addURI(AUTHORITY, "contacts", CONTACTS);  
 uriMatcher.addURI(AUTHORITY, "contacts/#", CONTACT\_ID);  
 }  
  
 private SQLiteDatabase db;  
  
 @Override  
 public boolean onCreate() {  
 ContactDatabaseHelper dbHelper = new ContactDatabaseHelper(getContext());  
 db = dbHelper.getWritableDatabase();  
 return (db != null);  
 }  
  
 @Override  
 public String getType(Uri uri) {  
 switch (uriMatcher.match(uri)) {  
 case CONTACTS:  
 return "vnd.android.cursor.dir/vnd.example.contacts";  
 case CONTACT\_ID:  
 return "vnd.android.cursor.item/vnd.example.contacts";  
 default:  
 throw new IllegalArgumentException("Unsupported URI: " + uri);  
 }  
 }  
  
 @Override  
 public Uri insert(Uri uri, ContentValues values) {  
 long id = db.insert("contacts", null, values);  
 if (id > 0) {  
 Uri newUri = ContentUris.withAppendedId(CONTENT\_URI, id);  
 getContext().getContentResolver().notifyChange(newUri, null);  
 return newUri;  
 }  
 throw new android.database.SQLException("Failed to insert row into " + uri);  
 }  
  
 @Override  
 public Cursor query(Uri uri, String[] projection, String selection, String[] selectionArgs, String sortOrder) {  
 Cursor cursor;  
 switch (uriMatcher.match(uri)) {  
 case CONTACTS:  
 cursor = db.query("contacts", projection, selection, selectionArgs, null, null, sortOrder);  
 break;  
 case CONTACT\_ID:  
 selection = "\_id = ?";  
 selectionArgs = new String[]{uri.getLastPathSegment()};  
 cursor = db.query("contacts", projection, selection, selectionArgs, null, null, sortOrder);  
 break;  
 default:  
 throw new IllegalArgumentException("Unsupported URI: " + uri);  
 }  
 cursor.setNotificationUri(getContext().getContentResolver(), uri);  
 return cursor;  
 }  
  
 @Override  
 public int update(Uri uri, ContentValues values, String selection, String[] selectionArgs) {  
 int count;  
 switch (uriMatcher.match(uri)) {  
 case CONTACTS:  
 count = db.update("contacts", values, selection, selectionArgs);  
 break;  
 case CONTACT\_ID:  
 selection = "\_id = ?";  
 selectionArgs = new String[]{uri.getLastPathSegment()};  
 count = db.update("contacts", values, selection, selectionArgs);  
 break;  
 default:  
 throw new IllegalArgumentException("Unsupported URI: " + uri);  
 }  
 getContext().getContentResolver().notifyChange(uri, null);  
 return count;  
 }  
  
 @Override  
 public int delete(Uri uri, String selection, String[] selectionArgs) {  
 int count;  
 switch (uriMatcher.match(uri)) {  
 case CONTACTS:  
 count = db.delete("contacts", selection, selectionArgs);  
 break;  
 case CONTACT\_ID:  
 selection = "\_id = ?";  
 selectionArgs = new String[]{uri.getLastPathSegment()};  
 count = db.delete("contacts", selection, selectionArgs);  
 break;  
 default:  
 throw new IllegalArgumentException("Unsupported URI: " + uri);  
 }  
 getContext().getContentResolver().notifyChange(uri, null);  
 return count;  
 }  
  
 private static class ContactDatabaseHelper extends SQLiteOpenHelper {  
 private static final String DATABASE\_NAME = "contacts.db";  
 private static final int DATABASE\_VERSION = 1;  
  
 public ContactDatabaseHelper(Context context) {  
 super(context, DATABASE\_NAME, null, DATABASE\_VERSION);  
 }  
  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
 db.execSQL("CREATE TABLE contacts (\_id INTEGER PRIMARY KEY AUTOINCREMENT, name TEXT, phone TEXT);");  
 }  
  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
 db.execSQL("DROP TABLE IF EXISTS contacts");  
 onCreate(db);  
 }  
 }  
}

**Activity\_main.xml**

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:padding="16dp">  
  
 <EditText  
 android:id="@+id/nameInput"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="Name" />  
  
 <EditText  
 android:id="@+id/phoneInput"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="Phone" />  
  
 <Button  
 android:id="@+id/addButton"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Add Contact" />  
  
 <Button  
 android:id="@+id/queryButton"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Query Contacts" />  
  
 <TextView  
 android:id="@+id/resultView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="16dp" />  
  
</LinearLayout>

**Manifest.xml**

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools">  
  
 <application  
 android:allowBackup="true"  
 android:dataExtractionRules="@xml/data\_extraction\_rules"  
 android:fullBackupContent="@xml/backup\_rules"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/Theme.Contentproviderexample"  
 tools:targetApi="31">  
 <provider  
 android:name=".ContactProvider"  
 android:authorities="com.example.contentproviderexample.provider"  
 android:exported="false" />  
 <activity  
 android:name=".MainActivity"  
 android:exported="true">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 </application>  
  
</manifest>

**Output :**



