Experiment 6

Experiment 6: Retrieve restaurant data from the local database and populate the ListView with restaurant names and categories.

Lab Objective:
To retrieve restaurant data from a local SQLite database and display it in a ListView.

Prerequisites:

1. Basic knowledge of databases
2. Programming fundamentals
3. Understanding of ListView
4. Development environment setup

Outcome:
- Connect to a local database and execute SQL queries.
- Populate a ListView with restaurant names and categories.
- Understand data flow from database to UI.
- Gain hands-on experience with database and ListView.

Step 1: activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:padding="16dp"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <ListView
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>
</LinearLayout>
```

Step 2: DatabaseHelper.kt

```
import android.content.Context
import android.database.sqlite.SQLiteDatabase
import android.database.sqlite.SQLiteOpenHelper

class DatabaseHelper(context: Context) :
    SQLiteOpenHelper(context, "restaurant.db", null, 1) {

    override fun onCreate(db: SQLiteDatabase) {
        db.execSQL(
            "CREATE TABLE restaurants(" +
                    "id INTEGER PRIMARY KEY AUTOINCREMENT, " +
                    "name TEXT, category TEXT);"
        )

        db.execSQL("INSERT INTO restaurants(name, category) VALUES('Spice Garden',
'Indian');")
        db.execSQL("INSERT INTO restaurants(name, category) VALUES('Burger King', 'Fast
Food');")
        db.execSQL("INSERT INTO restaurants(name, category) VALUES('Sushi House',
'Japanese');")
        db.execSQL("INSERT INTO restaurants(name, category) VALUES('Green Bowl',
'Vegan');")
    }

    override fun onUpgrade(db: SQLiteDatabase, oldVersion: Int, newVersion: Int) { }
}
```

Step 3: MainActivity.kt

```
import android.database.Cursor
import android.os.Bundle
import android.widget.ListView
import android.widget.SimpleCursorAdapter
import androidx.appcompat.app.AppCompatActivity

class MainActivity : AppCompatActivity() {

    private lateinit var listView: ListView
    private lateinit var dbHelper: DatabaseHelper

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        listView = findViewById(R.id.listView)
        dbHelper = DatabaseHelper(this)

        val db = dbHelper.readableDatabase
        val cursor: Cursor = db.rawQuery(
            "SELECT id AS _id, name, category FROM restaurants",
            null
        )

        val adapter = SimpleCursorAdapter(
            this,
            android.R.layout.simple_list_item_2,
            cursor,
            arrayOf("name", "category"),
            intArrayOf(android.R.id.text1, android.R.id.text2),
            0
        )

        listView.adapter = adapter
    }
}
```

Explanation:

1. The SQLite database is created using SQLiteOpenHelper.

2. Four restaurants are inserted when the database is first created.

3. A SELECT query retrieves id, name, and category.

4. SimpleCursorAdapter maps database columns to ListView rows.

5. ListView displays restaurant names and categories.

Test Cases:

1. Launch app → ListView shows four restaurants.

2. Database empty → ListView displays no data.

3. Add a restaurant row → Appears on next launch.

4. Update restaurant name → Updated name appears in ListView.

5. Insert many rows → ListView scrolls and loads all rows correctly.