Using an SQLite Database

In this tutorial we will see how to populate spinner data from an SQLite Database.

Step 1: Create new project in Eclipse IDE by going to **File -> Android Project** and fill the required details. Call the project **Database** or another name of your choice.

Creating the SQLite Database Handler Class

Step 2: Create a new class file and name it as **DatabaseHandler.java**. This SQLite handler class has following primary function to handle database operations.

Open your SQLite handler class (**DatabaseHandler.java**) file and paste the following code:

```java
import java.util.ArrayList;
import java.util.List;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class DatabaseHandler extends SQLiteOpenHelper {
    // Database Version
    private static final int DATABASE_VERSION = 1;

    // Database Name
    private static final String DATABASE_NAME =
"spinnerExample";

    // Labels table name
    private static final String TABLE_LABELS = "labels";

    // Labels Table Columns names
    private static final String KEY_ID = "id";
    private static final String KEY_NAME = "name";

    public DatabaseHandler(Context context) {
        super(context, DATABASE_NAME, null,
DATABASE_VERSION);
    }

    // Creating Tables
    @Override
    public void onCreate(SQLiteDatabase db) {
        // Category table create query
        String CREATE_CATEGORIES_TABLE = "CREATE TABLE " +
TABLE_LABELS + "("
                + KEY_ID + " INTEGER PRIMARY KEY," +
KEY_NAME + " TEXT)";
        db.execSQL(CREATE_CATEGORIES_TABLE);
    }

    // Upgrading database
    @Override
    public void onUpgrade(SQLiteDatabase db, int
oldVersion, int newVersion) {
        // Drop older table if existed
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_LABELS);

        // Create tables again
        onCreate(db);
    }

    /**
     * Inserting new label into labels table
     * */
    public void insertLabel(String label){
        SQLiteDatabase db = this.getWritableDatabase();

        ContentValues values = new ContentValues();
        values.put(KEY_NAME, label);

        // Inserting Row
        db.insert(TABLE_LABELS, null, values);
        db.close(); // Closing database connection
    }

    /**
     * Getting all labels
     * returns list of labels
     * */
    public List<String> getAllLabels(){
        List<String> labels = new ArrayList<String>();
        // Select All Query
        String selectQuery = "SELECT  * FROM " +
TABLE_LABELS;

        SQLiteDatabase db = this.getReadableDatabase();
        Cursor cursor = db.rawQuery(selectQuery, null);

        // looping through all rows and adding to list
        if (cursor.moveToFirst()) {
            do {
                labels.add(cursor.getString(1));
            } while (cursor.moveToNext());
        }

        // closing connection
        cursor.close();
        db.close();

        // returning labels
        return labels;
    }
}
```

Creating Spinner Dropdown

Step 3: Open your **activity_main.xml** and create a spinner dropdown using following xml code. In the following code a simple form is created with an EditText and a Spinner.

```
<?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" >

    <!-- Label -->
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="Add New Label"
        android:padding="8dip" />
    <!-- Input Text -->
    <EditText android:id="@+id/input_label"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="8dip"
        android:layout_marginRight="8dip"/>

    <!-- Add Button -->
    <Button android:id="@+id/btn_add"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Add Label"
        android:layout_marginLeft="8dip"
        android:layout_marginTop="8dip"/>

    <!-- Select Label -->
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="Select Label"
        android:padding="8dip" />

    <!-- Spinner Dropdown -->
    <Spinner
        android:id="@+id/spinner"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dip"
        android:layout_marginLeft="8dip"
        android:layout_marginRight="8dip"
    />
</LinearLayout>
```

Step 4: Open your main activity class and try the following code.

- **loadSpinnerData()** is called on **onCreate()** method to load the spinner data from SQLite database
- Once Add button is clicked, new label is inserted into database.
- After new label inserted into database, again **loadSpinnerData()** is called to load the spinner with newly added data from SQLite database

Paste the following code into your main_activity class:

```
import java.util.List;
import android.app.Activity;
import android.content.Context;
import android.os.Bundle;
import android.view.View;
import android.view.inputmethod.InputMethodManager;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;

public class MainActivity extends Activity implements
  OnItemSelectedListener {

    // Spinner element
    Spinner spinner;

    // Add button
    Button btnAdd;

    // Input text
    EditText inputLabel;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        // Spinner element
        spinner = (Spinner) findViewById(R.id.spinner);

        // add button
        btnAdd = (Button) findViewById(R.id.btn_add);

        // new label input field
        inputLabel = (EditText) findViewById(R.id.input_label);

        // Spinner click listener
        spinner.setOnItemSelectedListener(this);
    // Loading spinner data from database
    loadSpinnerData();

    /**
     * Add new label button click listener
     * */
    btnAdd.setOnClickListener(new View.OnClickListener() {

      @Override
      public void onClick(View arg0) {
        String label = inputLabel.getText().toString();

        if (label.trim().length() > 0) {
          // database handler
          DatabaseHandler db = new DatabaseHandler(
              getApplicationContext());

          // inserting new label into database
          db.insertLabel(label);

          // making input filed text to blank
          inputLabel.setText("");

          // Hiding the keyboard
          InputMethodManager imm = (InputMethodManager)
getSystemService(Context.INPUT_METHOD_SERVICE);
          imm.hideSoftInputFromWindow(inputLabel.getWindowToken(), 0);

          // loading spinner with newly added data
          loadSpinnerData();
        } else {
          Toast.makeText(getApplicationContext(), "Please enter label name",
              Toast.LENGTH_SHORT).show();
        }

      }
    });
  }

  /**
   * Function to load the spinner data from SQLite database
   * */
  private void loadSpinnerData() {
    // database handler
    DatabaseHandler db = new DatabaseHandler(getApplicationContext());
```

```
        // Spinner Drop down elements
        List<String> lables = db.getAllLabels();

        // Creating adapter for spinner
        ArrayAdapter<String> dataAdapter = new ArrayAdapter<String>(this,
                android.R.layout.simple_spinner_item, lables);

        // Drop down layout style - list view with radio button
        dataAdapter

.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);

        // attaching data adapter to spinner
        spinner.setAdapter(dataAdapter);
    }

    @Override
    public void onItemSelected(AdapterView<?> parent, View view, int position,
            long id) {
        // On selecting a spinner item
        String label = parent.getItemAtPosition(position).toString();

        // Showing selected spinner item
        Toast.makeText(parent.getContext(), "You selected: " + label,
                Toast.LENGTH_LONG).show();

    }

    @Override
    public void onNothingSelected(AdapterView<?> arg0) {
        // TODO Auto-generated method stub

    }
}
```

Step 5: Run your project add new entries into the database.