***What is the use of SQLite open helper class in SQLite?***

It is a helper class to manage database creation and version management.We can create a subclass implementing onCreate , onUpgrade or onOpen and this class takes care of opening the database if it exists, creating it if it does not, and upgrading it as necessary. Transactions are used to make sure the database is always in a sensible state.

This class makes it easy for ContentProvider implementations to defer opening and upgrading the database until first use, to avoid blocking application startup with long-running database upgrades.

***What is the use of OnUpgrade function in SQLiteOpenHelper class?***

This function is called when the database needs to be upgraded. The implementation should use this method to drop tables, add tables, or do anything else it needs to upgrade to the new schema version.

***Syntax***: void onUpgrade (SQLiteDatabase db,int oldVersion,int newVersion)

*Parameters:*

db SQliteDatabase : The database

oldversion int : The old database version

newVersion int : The new database version

***How to show SQLite database table information in Android application what is the best***

***way to do it?***

Use table layout for showing database information in android application. The TableLayout is a view group to display data in a table format. A table might have many rows and columns. The TableRow class is used to define a row of the table. In the row, you can have multiple columns. When you add views (e.g. TextView) to the row, by default, the first view is in the first column (column index 0); the second view is in the second column.

Example :

1. <TableLayout xmlns:android=" http://schemas.android.com/apk/res/android"  
      xmlns:tools= "http://schemas.android.com/tools"  
      android:id= "@+id/tablelayout"  
      android:layout\_width= "match\_parent"  
      android:layout\_height= "match\_parent"  
      android:stretchColumns="\*"  
     
      >  
   </TableLayout>

Is used in xml to create an empty table layout.

1. We create a class called DatabaseHelper that is used to create a database (outletdb), a table (tbloutletdata), and insert data to the table. The tbloutletdata table has three columns: outlet\_id, outlet\_name, and outlet\_type.

public class DatabaseHelper extends SQLiteOpenHelper {  
   public static final String DATABASE\_NAME = "outletdb";  
   public static final int DATABASE\_VERSION = 1;  
   public static final String TABLE\_OUTLET = "tbloutletdata";  
   public static final String CREATE\_TABLE\_OUTLET= "CREATE TABLE IF NOT EXISTS "+ TABLE\_OUTLET+ "(outlet\_id INTEGER PRIMARY KEY AUTOINCREMENT, outlet\_name TEXT NULL, outlet\_type TEXT NULL)";  
   public static final String DELETE\_TABLE\_OUTLET="DROP TABLE IF EXISTS " + TABLE\_OUTLET;  
  
   public DatabaseHelper(Context context) {  
     super(context, DATABASE\_NAME, null, DATABASE\_VERSION);  
  
   }  
   public void onCreate(SQLiteDatabase db) {  
  
     db.execSQL(CREATE\_TABLE\_OUTLET);  
  
   }  
   //Upgrading database  
   @Override  
   public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
  
     db.execSQL(DELETE\_TABLE\_OUTLET);  
     //Create tables again  
     onCreate(db);  
   }  
  
   public void insertData(String outlet\_name,String outlet\_type ){  
  
     // Open the database for writing  
     SQLiteDatabase db = this.getWritableDatabase();  
     // Start the transaction.  
     db.beginTransaction();  
     ContentValues values;  
  
     try  
     {  
        values = new ContentValues();  
        values.put("outlet\_name",outlet\_name);  
        values.put("outlet\_type",outlet\_type);  
        // Insert Row  
        long i = db.insert(TABLE\_OUTLET, null, values);  
        Log.i("Insert", i + "");  
        // Insert into database successfully.  
        db.setTransactionSuccessful();  
  
     }  
     catch (SQLiteException e)  
     {  
        e.printStackTrace();  
  
     }  
     finally  
     {  
        db.endTransaction();  
        // End the transaction.  
        db.close();  
        // Close database  
     }  
  
   }  
  
  
}

1. In the MainActivity class that hosts the TableLayout, you create an instance of DatabaseHelper, insert sample data to the database, add header row to the table, and add data rows to the table.

public class MainActivity extends Activity {  
   private Context context;  
   @Override  
   protected void onCreate(Bundle savedInstanceState) {  
     super.onCreate(savedInstanceState);  
     setContentView(R.layout.activity\_main);  
     context=this;  
     // Create DatabaseHelper instance  
     DatabaseHelper dataHelper=new DatabaseHelper(context);  
   // Insert sample data  
     dataHelper.insertData("Kent","HoReCa");  
     dataHelper.insertData("FineE","HoReCa");  
     dataHelper.insertData("MKent","GTrade");  
     dataHelper.insertData("MeviusLove","MTrade");  
     dataHelper.insertData("XEMo","HoReCa");  
     // Reference to TableLayout  
     TableLayout tableLayout=(TableLayout)findViewById(R.id.tablelayout);  
     // Add header row  
     TableRow rowHeader = new TableRow(context);  
     rowHeader.setBackgroundColor(Color.parseColor("#c0c0c0"));  
     rowHeader.setLayoutParams(new TableLayout.LayoutParams(TableLayout.LayoutParams.MATCH\_PARENT,  
           TableLayout.LayoutParams.WRAP\_CONTENT));  
     String[] headerText={"ID","NAME","TYPE"};  
     for(String c:headerText) {  
        TextView tv = new TextView(this);  
        tv.setLayoutParams(new TableRow.LayoutParams(TableRow.LayoutParams.WRAP\_CONTENT,  
             TableRow.LayoutParams.WRAP\_CONTENT));  
        tv.setGravity(Gravity.CENTER);  
        tv.setTextSize(18);  
        tv.setPadding(5, 5, 5, 5);  
        tv.setText(c);  
        rowHeader.addView(tv);  
     }  
     tableLayout.addView(rowHeader);  
  
     // Get data from sqlite database and add them to the table  
     // Open the database for reading  
     SQLiteDatabase db = dataHelper.getReadableDatabase();  
     // Start the transaction.  
     db.beginTransaction();  
  
     try  
     {  
        String selectQuery = "SELECT \* FROM "+ DatabaseHelper.TABLE\_OUTLET;  
        Cursor cursor = db.rawQuery(selectQuery,null);  
        if(cursor.getCount() >0)  
        {  
           while (cursor.moveToNext()) {  
             // Read columns data  
             int outlet\_id= cursor.getInt(cursor.getColumnIndex("outlet\_id"));  
             String outlet\_name= cursor.getString(cursor.getColumnIndex("outlet\_name"));  
             String outlet\_type= cursor.getString(cursor.getColumnIndex("outlet\_type"));  
  
             // dara rows  
             TableRow row = new TableRow(context);  
             row.setLayoutParams(new TableLayout.LayoutParams(TableLayout.LayoutParams.MATCH\_PARENT,  
                   TableLayout.LayoutParams.WRAP\_CONTENT));  
             String[] colText={outlet\_id+"",outlet\_name,outlet\_type};  
             for(String text:colText) {  
                TextView tv = new TextView(this);  
                tv.setLayoutParams(new TableRow.LayoutParams(TableRow.LayoutParams.WRAP\_CONTENT,  
                     TableRow.LayoutParams.WRAP\_CONTENT));  
                tv.setGravity(Gravity.CENTER);  
                tv.setTextSize(16);  
                tv.setPadding(5, 5, 5, 5);  
                tv.setText(text);  
                row.addView(tv);  
             }  
             tableLayout.addView(row);  
  
           }  
  
        }  
        db.setTransactionSuccessful();  
  
     }  
     catch (SQLiteException e)  
     {  
        e.printStackTrace();  
  
     }  
     finally  
     {  
        db.endTransaction();  
        // End the transaction.  
        db.close();  
        // Close database  
     }  
}