* Managing data using Sqlite

SQLite is a opensource SQL database that stores data to a text file on a device. Android comes in with built in SQLite database implementation.

SQLite supports all the relational database features. In order to access this database, you don't need to establish any kind of connections for it like JDBC,ODBC e.t.c

**Database - Package**

The main package is android.database.sqlite that contains the classes to manage your own databases

**Database - Creation**

In order to create a database you just need to call this method openOrCreateDatabase with your database name and mode as a parameter. It returns an instance of SQLite database which you have to receive in your own object.

Its syntax is given below

SQLiteDatabase mydatabase = openOrCreateDatabase("your database name",MODE\_PRIVATE,null);

Apart from this , there are other functions available in the database package , that does this job.

**Database - Insertion**

we can create table or insert data into table using execSQL method defined in SQLiteDatabase class.

Its syntax is given below

mydatabase.execSQL("CREATE TABLE IF NOT EXISTS Customer(Username VARCHAR,Password VARCHAR);");

mydatabase.execSQL("INSERT INTO Customer VALUES('ramu','ram@123');");

**Database - Fetching**

We can retrieve anything from database using an object of the Cursor class. We will call a method of this class called rawQuery and it will return a resultset with the cursor pointing to the table. We can move the cursor forward and retrieve the data.

Cursor resultSet = mydatbase.rawQuery("Select \* from Customer",null);

resultSet.moveToFirst();

String username = resultSet.getString(0);

String password = resultSet.getString(1);

There are other functions available in the Cursor class that allows us to effectively retrieve the data. That includes

|  |  |
| --- | --- |
| **Sr.No** | **Method & Description** |
| 1 | **getColumnCount()**  This method return the total number of columns of the table. |
| 2 | **getColumnIndex(String columnName)**  This method returns the index number of a column by specifying the name of the column |
| 3 | **getColumnName(int columnIndex)**  This method returns the name of the column by specifying the index of the column |
| 4 | **getColumnNames()**  This method returns the array of all the column names of the table. |
| 5 | **getCount()**  This method returns the total number of rows in the cursor |
| 6 | **getPosition()**  This method returns the current position of the cursor in the table |
| 7 | **isClosed()**  This method returns true if the cursor is closed and return false otherwise |

Database - Helper class

For managing all the operations related to the database , an helper class has been given and is called SQLiteOpenHelper. It automatically manages the creation and update of the database. Its syntax is given below

public class DBHelper extends SQLiteOpenHelper {

public DBHelper(){

super(context,DATABASE\_NAME,null,1);

}

public void onCreate(SQLiteDatabase db) {}

public void onUpgrade(SQLiteDatabase database, int oldVersion, int newVersion) {}

}