SQLite Database

1 About the Android Studio Database Example

2 Creating the Data Model

3 Implementing the Data Handler

3.1 The Add Handler Method

3.2 The Query Handler Method

3.3 The Delete Handler Method

4 Implementing the Activity Event Methods

5 Testing the Application

6 Summary

The database schema for the products table is outlined as follows:

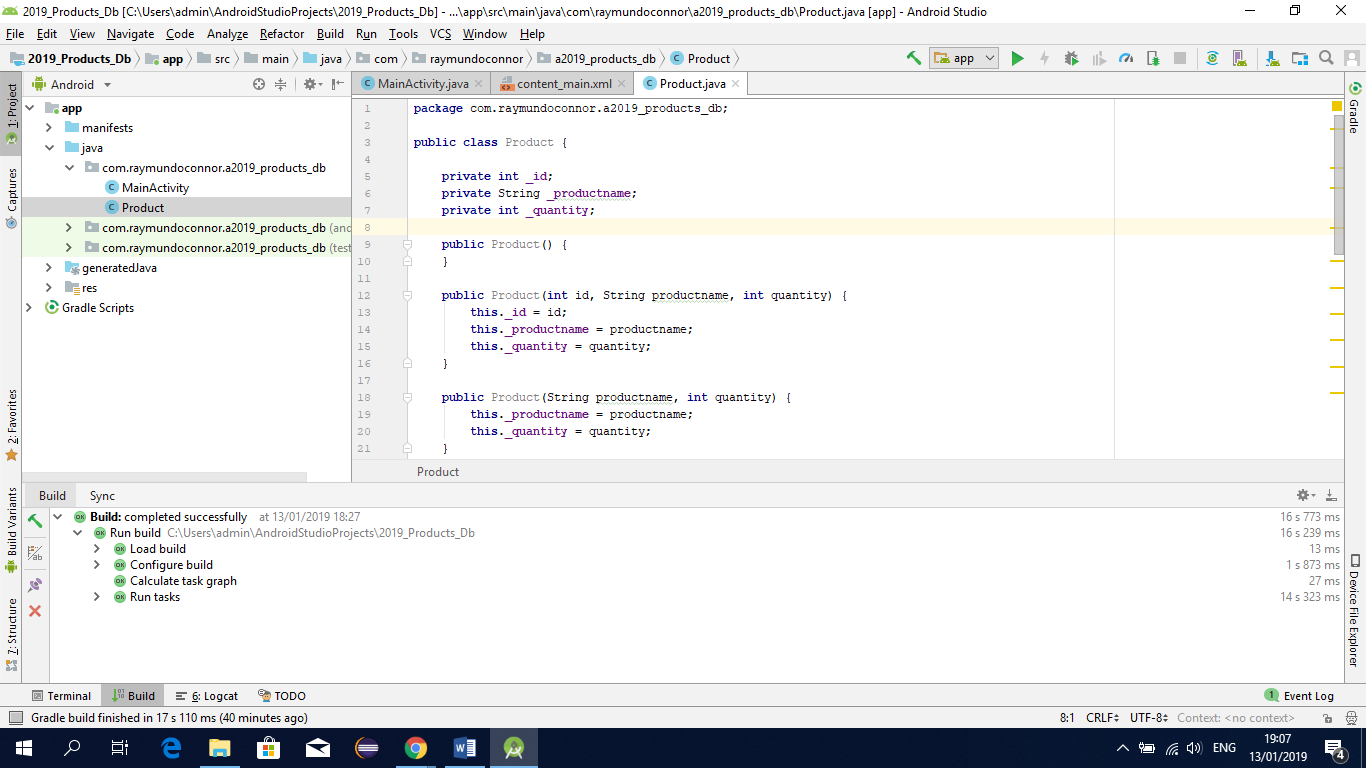
|  |  |
| --- | --- |
| **Column** | **Data Type** |
| productid | Integer / Primary Key/ Auto Increment |
| productname | Text |
| productquantity | Integer |

**Creating the Data Model**

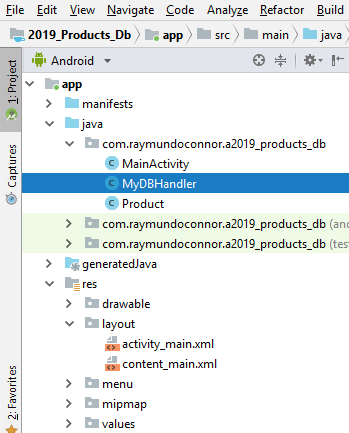
The application will consist of

* an activity
* a database handler class (MyDBHandler a subclass of SQLiteOpenHelper)
* a third class(Product) will need to be implemented to hold the database entry data as it is passed between the activity and the handler.

This is actually a very simple class capable of holding product ID, product name and product quantity values, together with getter and setter methods for accessing these values. Instances of this class can then be created within the activity and database handler and passed back and forth as needed. Essentially, this class can be thought of as representing the database model.



When finished your classes should look like this (see below)



**Product Class**

package com.raymundoconnor.a2019\_products\_db;

public class Product {  
 private int \_id;  
 private String \_productname;  
 private int \_quantity;

public Product() {  
 }

public Product(int id, String productname, int quantity) {  
 this.\_id = id;  
 this.\_productname = productname;  
 this.\_quantity = quantity;  
 }

public Product(String productname, int quantity) {  
 this.\_productname = productname;  
 this.\_quantity = quantity;  
 }

public void setID(int id) {  
 this.\_id = id;  
 }

public int getID() {  
 return this.\_id;  
 }

public void setProductName(String productname) {  
 this.\_productname = productname;  
 }

public String getProductName() {  
 return this.\_productname;  
 }

public void setQuantity(int quantity) {  
 this.\_quantity = quantity;  
 }

public int getQuantity() {  
 return this.\_quantity;  
 }

}

**MyDbHandler Class**

**package** com.raymundoconnor.a2019\_products\_db;  
  
**import** android.content.Context;  
**import** android.database.sqlite.SQLiteDatabase;  
**import** android.database.sqlite.SQLiteOpenHelper;  
**import** android.content.Context;  
**import** android.content.ContentValues;  
**import** android.database.Cursor;  
  
**public class** MyDBHandler **extends** SQLiteOpenHelper{  
 *// when you type SQLiteOpenHelper as above you must import the onCreate, onUpgrade methods* **private static final int *DATABASE\_VERSION*** = 1;  
 **private static final** String ***DATABASE\_NAME*** = **"productDB.db"**; **private static final** String ***TABLE\_PRODUCTS*** = **"products"**;  
  
 **public static final** String ***COLUMN\_ID*** = **"\_id"**;  
 **public static final** String ***COLUMN\_PRODUCTNAME*** = **"productname"**;  
 **public static final** String ***COLUMN\_QUANTITY*** = **"quantity"**;  
  
 **public** MyDBHandler(Context context, String name,  
 SQLiteDatabase.CursorFactory factory, **int** version) {  
 **super**(context, ***DATABASE\_NAME***, factory, ***DATABASE\_VERSION***);  
 }  
  
 @Override  
 **public void** onCreate(SQLiteDatabase db) {  
 String CREATE\_PRODUCTS\_TABLE = **"CREATE TABLE "** +  
 ***TABLE\_PRODUCTS*** + **"("** + ***COLUMN\_ID*** + **" INTEGER PRIMARY KEY,"** + ***COLUMN\_PRODUCTNAME*** + **" TEXT,"** + ***COLUMN\_QUANTITY*** + **" INTEGER"** + **")"**;  
 db.execSQL(CREATE\_PRODUCTS\_TABLE);  
 }  
  
 @Override  
 **public void** onUpgrade(SQLiteDatabase db, **int** oldVersion, **int** newVersion) {  
 db.execSQL(**"DROP TABLE IF EXISTS "** + ***TABLE\_PRODUCTS***);  
 onCreate(db);  
 }  
  
 **public void** addProduct(Product product) {  
  
 ContentValues values = **new** ContentValues();  
 values.put(***COLUMN\_PRODUCTNAME***, product.getProductName());  
 values.put(***COLUMN\_QUANTITY***, product.getQuantity());  
  
 SQLiteDatabase db = **this**.getWritableDatabase();  
  
 db.insert(***TABLE\_PRODUCTS***, **null**, values);  
 db.close();  
 }  
  
  
 **public** Product findProduct(String productname) {  
 String query = **"Select \* FROM "** + ***TABLE\_PRODUCTS*** + **" WHERE "** + ***COLUMN\_PRODUCTNAME*** + **" = \""** + productname + **"\""**;  
  
 SQLiteDatabase db = **this**.getWritableDatabase();  
  
 Cursor cursor = db.rawQuery(query, **null**);  
  
 Product product = **new** Product();  
  
 **if** (cursor.moveToFirst()) {  
 cursor.moveToFirst();  
 product.setID(Integer.*parseInt*(cursor.getString(0)));  
 product.setProductName(cursor.getString(1));  
 product.setQuantity(Integer.*parseInt*(cursor.getString(2)));  
 cursor.close();  
 } **else** {  
 product = **null**;  
 }  
 db.close();  
 **return** product;  
 }  
  
  
 **public boolean** deleteProduct(String productname) {  
  
 **boolean** result = **false**;  
  
 String query = **"Select \* FROM "** + ***TABLE\_PRODUCTS*** + **" WHERE "** + ***COLUMN\_PRODUCTNAME*** + **" = \""** + productname + **"\""**;  
  
 SQLiteDatabase db = **this**.getWritableDatabase();  
  
 Cursor cursor = db.rawQuery(query, **null**);  
  
 Product product = **new** Product();  
  
 **if** (cursor.moveToFirst()) {  
 product.setID(Integer.*parseInt*(cursor.getString(0)));  
 db.delete(***TABLE\_PRODUCTS***, ***COLUMN\_ID*** + **" = ?"**,  
 **new** String[] { String.*valueOf*(product.getID()) });  
 cursor.close();  
 result = **true**;  
 }  
 db.close();  
 **return** result;  
 }  
  
}

**MainActivity Class**

**package** com.raymundoconnor.a2019\_products\_db;  
**import** android.os.Bundle;  
**import** android.support.design.widget.FloatingActionButton;  
**import** android.support.design.widget.Snackbar;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.support.v7.widget.Toolbar;  
**import** android.view.View;  
**import** android.view.Menu;  
**import** android.view.MenuItem;  
**import** android.widget.EditText;  
**import** android.widget.TextView;  
  
**public class** MainActivity **extends** AppCompatActivity {  
 TextView **idView**;  
 EditText **productBox**;  
 EditText **quantityBox**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
  
 **idView** = (TextView) findViewById(R.id.***textViewID***);  
 **productBox** = (EditText) findViewById(R.id.***editTextProduct***);  
 **quantityBox** = (EditText) findViewById(R.id.***editTextQuantity***);  
 }  
  
 **public void** newProduct (View view) {  
 MyDBHandler dbHandler = **new** MyDBHandler(**this**, **null**, **null**, 1);  
  
 **int** quantity =  
 Integer.*parseInt*(**quantityBox**.getText().toString());  
  
 Product product =  
 **new** Product(**productBox**.getText().toString(), quantity);  
  
 dbHandler.addProduct(product);  
 **productBox**.setText(**""**);  
 **quantityBox**.setText(**""**);  
 }  
  
 **public void** lookupProduct (View view) {  
 MyDBHandler dbHandler = **new** MyDBHandler(**this**, **null**, **null**, 1);  
  
 Product product =  
 dbHandler.findProduct(**productBox**.getText().toString());  
  
 **if** (product != **null**) {  
 **idView**.setText(String.*valueOf*(product.getID()));  
  
 **quantityBox**.setText(String.*valueOf*(product.getQuantity()));  
 } **else** {  
 **idView**.setText(**"No Match Found"**);  
 }  
 }  
  
 **public void** removeProduct (View view) {  
 MyDBHandler dbHandler = **new** MyDBHandler(**this**, **null**,  
 **null**, 1);  
  
 **boolean** result = dbHandler.deleteProduct(  
 **productBox**.getText().toString());  
  
 **if** (result)  
 {  
 **idView**.setText(**"Record Deleted"**);  
 **productBox**.setText(**""**);  
 **quantityBox**.setText(**""**);  
 }  
 **else  
 idView**.setText(**"No Match Found"**);  
 }  
}

**Testing the app**

**To add a product**

* Enter a product name and quantity and click ADD

**To find a product**

* Click in the product name edit text and enter the product name you require and click FIND

**To delete a product**

* Click in the product name edit text and enter the product name you require and click DELETE

