모바일 프로그래밍 실습 보고서

학번	201411203	이름	박소영
순서	19	날짜	2018.05.31 목요일
강의자료	10 - SQLite		
주제	**안드로이드 내장 데이터베이스 이용하기**		

소스코드

java 소스코드 : MainActivity.class

```
public class MainActivity extends AppCompatActivity {
    EditText pname;
    EditText pquantity;
    EditText id;

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

        id = (EditText) findViewById(R.id._id);
        pname = (EditText) findViewById(R.id.productname);
        pquantity = (EditText) findViewById(R.id.quantity);
    }

    public void newProduct(View view) {
        MyDBHandler dbHandler = new MyDBHandler(this, null, null, 1);
        int quantity = Integer.parseInt(pquantity.getText().toString());
        Product product = new Product(pname.getText().toString(), quantity);
        Boolean result = dbHandler.addProduct(product);
        if (result) {
         Toast.makeText(this, "DB INSERT SUCCESS", Toast.LENGTH_SHORT).show();
        }
    }

    public void deleteProduct(View view) {
        MyDBHandler dbHandler = new MyDBHandler(this, null, null, 1);
```

```
        boolean result = dbHandler.deleteProduct(pname.getText().toString());
        if(result){
            id.setText("");
            pquantity.setText("");
            pname.setText("");
            Toast.makeText(this, "Record Deleted", Toast.LENGTH_SHORT).show();
        }else{
            Toast.makeText(this, "No Match Found", Toast.LENGTH_SHORT).show();
        }
    }

    public void findProduct(View view) {
        /*
        MyDBHandler dbHandler = new MyDBHandler(this, null, null, 1);
        Product product = dbHandler.findProduct(pname.getText().toString());
        if(product != null){
            id.setText(String.valueOf(product.getID()));
            pquantity.setText(String.valueOf(product.getQuantity()));
        }else{
            Toast.makeText(this, "No Match Found", Toast.LENGTH_SHORT).show();
        }
         */
        Intent intent = new Intent(this, FindActivity.class);
        startActivity(intent);
    }

    public void updateProduct(View view) {
        MyDBHandler dbHandler = new MyDBHandler(this, null, null, 1);
        int quantity = Integer.parseInt(pquantity.getText().toString());
        Product product = new Product(pname.getText().toString(), quantity);
        boolean result = dbHandler.updateProduct(product);
        if(result) {
            id.setText("");
            pquantity.setText("");
            pname.setText("");
            Toast.makeText(this, "Record Updated", Toast.LENGTH_SHORT).show();
        } else {
            Toast.makeText(this, "No Match Found", Toast.LENGTH_SHORT).show();
        }

    }
}
```

Layout : activity_main.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity" android:padding="20dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="ID : " />
        <EditText
            android:id="@+id/_id"
            android:layout_width="match_parent"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="NAME : " />
        <EditText
            android:id="@+id/productname"
            android:layout_width="match_parent"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <TextView
```

```
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="QUANTITY : " />
        <EditText
            android:id="@+id/quantity"
            android:layout_width="match_parent"
            android:layout_height="wrap_content" />
    </LinearLayout>

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="findProduct"
        android:text="FIND"
        android:layout_marginTop="10dp"
        android:layout_marginBottom="10dp"/>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center">
        <Button
            android:layout_width="0dp"
            android:layout_weight="1"
            android:layout_height="wrap_content"
            android:onClick="newProduct"
            android:text="INSERT" />
        <Button
            android:layout_width="0dp"
            android:layout_weight="1"
            android:layout_height="wrap_content"
            android:onClick="deleteProduct"
            android:text="DELETE"
            android:paddingLeft="15dp"
            android:paddingRight="15dp"/>
        <Button
            android:layout_width="0dp"
            android:layout_weight="1"
            android:layout_height="wrap_content"
            android:onClick="updateProduct"
            android:text="UPDATE" />
    </LinearLayout>
</LinearLayout>
```

소스코드

java 소스코드 : FindActivity.class

```java
public class FindActivity extends AppCompatActivity {
    EditText queryEditText;
    GridLayout gridLayout;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_find);

        init();
    }

    public void init() {
        queryEditText = (EditText) findViewById(R.id.queryEditText);
        gridLayout = (GridLayout) findViewById(R.id.gridLayout);
    }

    public void execQuery(View view) {
        MyDBHandler dbHandler = new MyDBHandler(this, null, null, 1);
        String sql = queryEditText.getText().toString();

        Cursor cursor = dbHandler.selectQuery(sql);
        cursor.moveToFirst();
        if (!cursor.isAfterLast()) {
            // last가 아니다 > 질의문의 답이 무언가 있다.
            // 답의 개수만큼 gridLayout을 구성해야 한다.

            int column = cursor.getColumnCount();
            int row = cursor.getCount();

            gridLayout.removeAllViewsInLayout(); // 새로 그리기 전에 일단 지운다.
            gridLayout.setColumnCount(column);
            gridLayout.setRowCount(row + 1);

            for(int i=0; i<column; i++) {
                View v = getLayoutInflater().inflate(R.layout.row, null);
                v.setMinimumWidth(130);
                TextView item = (TextView) v.findViewById(R.id.item);
                item.setTextSize(17);
```

```
                item.setGravity(Gravity.CENTER);
                item.setText(cursor.getColumnName(i));

                item.setBackgroundColor(Color.LTGRAY);
                gridLayout.addView(v);
            }

            while(!cursor.isAfterLast()) {
                for (int i=0; i<column; i++) {
                    View v = getLayoutInflater().inflate(R.layout.row, null);
                    TextView item = (TextView) v.findViewById(R.id.item);
                    item.setText(cursor.getString(i));
                    gridLayout.addView(v);
                }
                // 한줄 다 읽어들인 이후에는 커서를 다음 줄로 옮겨줘야 한다.
                cursor.moveToNext();
            }
        }
    }
}
```

Layout : activity_find.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:padding="20dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:layout_marginBottom="20dp">

        <EditText
            android:id="@+id/queryEditText"
            android:layout_width="0dp"
            android:layout_weight="3"
```

```
            android:layout_height="wrap_content" />

        <Button
            android:id="@+id/execQueryButton"
            android:layout_width="0dp"
            android:layout_weight="1"
            android:layout_height="wrap_content"
            android:onClick="execQuery"
            android:text="GO!" />
    </LinearLayout>

    <GridLayout
        android:id="@+id/gridLayout"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />
</LinearLayout>
```

java 소스코드 : MyDBHandler.class

```
public class MyDBHandler extends SQLiteOpenHelper {
    public static final String DATABASE_NAME = "productDB.db";
    public static final String DATABASE_TABLE = "products";
    public static final String COLUMN_ID = "_id";
    public static final String COLUMN_PRODUCTNAME = "productname";
    public static final String COLUMN_QUANTITY = "quantity";
    Context context;

    public MyDBHandler(Context context, String name,
                       SQLiteDatabase.CursorFactory factory, int version) {
        super(context, DATABASE_NAME, factory, version);

        this.context = context;
        onCreate(this.getWritableDatabase());
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String CREATE_TABLE = "create table if not exists " + DATABASE_TABLE
              + "("+ COLUMN_ID + " integer primary key autoincrement,"
              + COLUMN_PRODUCTNAME + " text,"+COLUMN_QUANTITY+" integer"+")";

        db.execSQL(CREATE_TABLE);
```

```
        Scanner scan = new Scanner(context.getResources()
            .openRawResource(R.raw.sql));
        String query = "";

        while (scan.hasNextLine()) {
            query += scan.nextLine()+"\n";

            if (query.trim().endsWith(";")) {
                db.execSQL(query);
                query = "";
            }
        }
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("drop table if exists " + DATABASE_TABLE);
        onCreate(db);
    }

    public boolean addProduct(Product product) {
        ContentValues values = new ContentValues();
        values.put(COLUMN_PRODUCTNAME, product.getProductName());
        values.put(COLUMN_QUANTITY, product.getQuantity());
        SQLiteDatabase db = this.getWritableDatabase();
        if (db.insert(DATABASE_TABLE, null, values) > 0) {
            db.close();
            return true;
        } else {
            db.close();
            return false;
        }
    }

    public boolean deleteProduct(String productname){
        boolean result = false;
        String query="select * from "+ DATABASE_TABLE +
                " 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(DATABASE_TABLE, COLUMN_ID + "=?",
                    new String[]{String.valueOf(product.getID())});
            cursor.close();
            db.close();
            return true;
        }
        db.close();
        return result;
    }

    public Product findProduct(String productname){
        String query="select * from "+ DATABASE_TABLE + " 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)));
            product.setProductName(cursor.getString(1));
            product.setQuantity(Integer.parseInt(cursor.getString(2)));
            cursor.close();
        } else {
            product = null;
        }
        db.close();
        return product;
    }

    public boolean updateProduct(Product argproduct){
        Product product = findProduct(argproduct.getProductName());
        if(product != null)
        {
            SQLiteDatabase db = this.getWritableDatabase();
            ContentValues values=new ContentValues();
            values.put(COLUMN_PRODUCTNAME, product.getProductName());
            values.put(COLUMN_QUANTITY, argproduct.getQuantity());
            db.update(DATABASE_TABLE, values, COLUMN_ID + "= \'" +
                    product.getID() + "\'", null );
            db.close();
            return true;
        }
        return false;
    }
```

```
    public Cursor selectQuery(String query) {
        SQLiteDatabase db = this.getWritableDatabase();
        return db.rawQuery(query, null); // 질의문안에 포함되어 있어 인자 필요 X
    }
}
```

실행 결과