

# 스마트폰 프로그래밍

## Assign #7

(제목 : SQLite Test)

학번 : 2016305078

성명 : 최 영 환

제출일자 : 2021.10.20

본인은 타인이나 다른 매체를 도용하지 않았음을 확인합니다 <본인사인>

<실행화면>

11:34

수업 관리 DB

수업명 :

수강인원 :

초기 입력 수정 삭제 조회

수업명	인원
-----	----

11:36

수업 관리 DB

수업명 : Android

수강인원 : 35

초기 입력 수정 삭제 조회

수업명	인원
Android	35

성공적으로 추가되었음

11:36

수업 관리 DB

수업명 : Python

수강인원 : 50

초기 입력 수정 삭제 조회

수업명	인원
Android	35
Python	50

성공적으로 추가되었음

11:37

수업 관리 DB

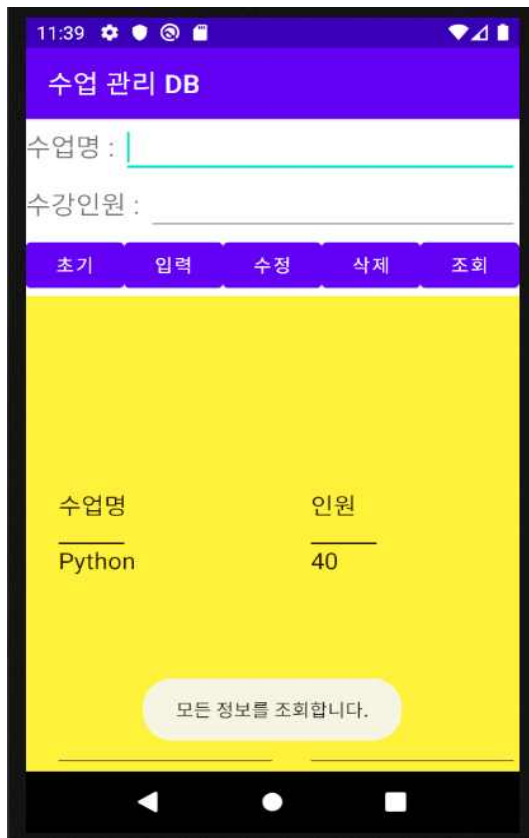
수업명 : Python

수강인원 : 40

초기 입력 수정 삭제 조회

수업명	인원
Android	35
Python	40

성공적으로 수정되었음



## <MainActivity.java>

```
1 package com.example.sqlitetest;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.content.Context;
6 import android.database.Cursor;
7 import android.database.SQLException;
8 import android.database.sqlite.SQLiteDatabase;
9 import android.database.sqlite.SQLiteOpenHelper;
10 import android.os.Bundle;
11 import android.view.View;
12 import android.widget.EditText;
13 import android.widget.Toast;
14
15 class DBHelper extends SQLiteOpenHelper {
16     private static final String DATABASE_NAME = "MyLecture.db";
17     private static final int DATABASE_VERSION = 2;
18
19     public DBHelper(Context context) { super(context, DATABASE_NAME, factory: null, DATABASE_VERSION); }
20
21     public void onCreate(SQLiteDatabase db) {
22         db.execSQL("CREATE TABLE lecture ( _id INTEGER PRIMARY KEY AUTOINCREMENT, subject TEXT, people TEXT);");
23     }
24
25     public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
26         db.execSQL("DROP TABLE IF EXISTS lecture");
27         onCreate(db);
28     }
29 }
30
31 }
```

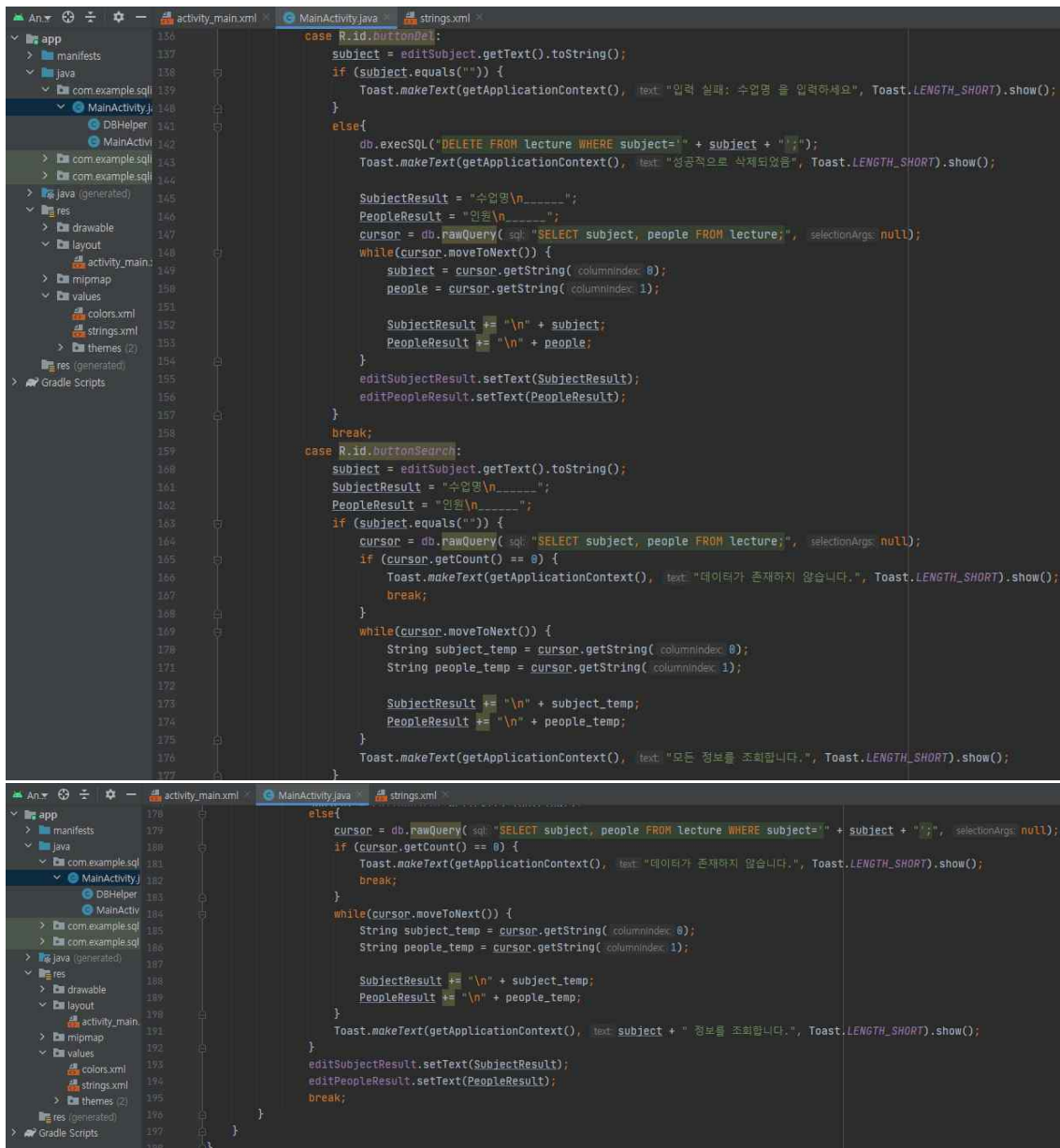
```
33 public class MainActivity extends AppCompatActivity {
34     DBHelper helper;
35     SQLiteDatabase db;
36     EditText editSubject;
37     EditText editPeople;
38     EditText editSubjectResult;
39     EditText editPeopleResult;
40
41     @Override
42     protected void onCreate(Bundle savedInstanceState) {
43         super.onCreate(savedInstanceState);
44         setContentView(R.layout.activity_main);
45
46         helper = new DBHelper( context this);
47         try{
48             db = helper.getWritableDatabase();
49         } catch (SQLException ex) {
50             db = helper.getReadableDatabase();
51         }
52
53         editSubject = (EditText) findViewById(R.id.editSubject);
54         editPeople = (EditText) findViewById(R.id.editPeople);
55         editSubjectResult = (EditText) findViewById(R.id.editSubjectResult);
56         editPeopleResult = (EditText) findViewById(R.id.editPeopleResult);
57     }
58 }
```

```
activity_main.xml MainActivity.java strings.xml
app
├── manifests
├── java
│   └── com.example.sql
│       ├── MainActivity.java
│       ├── DBHelper
│       └── MainActivity
├── com.example.sql
├── com.example.sql
├── java (generated)
├── res
│   ├── drawable
│   ├── layout
│   ├── activity_main
│   ├── mipmap
│   └── values
│       ├── colors.xml
│       ├── strings.xml
│       └── themes (2)
└── res (generated)
Gradle Scripts

59 public void onClick(View view) {
60     String subject;
61     String people;
62     String SubjectResult;
63     String PeopleResult;
64     Cursor cursor;
65
66     switch (view.getId()) {
67         case R.id.buttonInit:
68             db.execSQL("DROP TABLE IF EXISTS lecture");
69             db.execSQL("CREATE TABLE lecture ( _id INTEGER PRIMARY KEY AUTOINCREMENT, subject TEXT, people TEXT);");
70             Toast.makeText(getApplicationContext(), "성공적으로 초기화되었음", Toast.LENGTH_SHORT).show();
71             editSubjectResult.setText("");
72             editPeopleResult.setText("");
73             break;
74         case R.id.buttonInput:
75             subject = editSubject.getText().toString();
76             people = editPeople.getText().toString();
77             if (subject.equals("") || people.equals("")) {
78                 Toast.makeText(getApplicationContext(), "입력 실패: 빈칸을 입력하세요.", Toast.LENGTH_SHORT).show();
79             }
80             else {
81                 SubjectResult = "수업명\n_____";
82                 PeopleResult = "인원\n_____";
83
84                 boolean activation = true;
85                 cursor = db.rawQuery("SELECT subject, people FROM lecture", selectionArgs, null);
86                 while(cursor.moveToNext()) {
87                     String subject_temp = cursor.getString(columnindex 0);
88                     if (subject.equals(subject_temp)) {
89                         activation = false;
90                     }
91                 }
92                 if(activation == true){
93                     db.execSQL("INSERT INTO lecture VALUES (null, " + subject + ", " + people + ");");
94                     Toast.makeText(getApplicationContext(), "성공적으로 추가되었음", Toast.LENGTH_SHORT).show();
95                 }
96                 else {
97                     Toast.makeText(getApplicationContext(), "이미 존재하는 과목입니다.", Toast.LENGTH_SHORT).show();
98                 }
99             }
100     }
101 }
```

```
activity_main.xml MainActivity.java strings.xml
app
├── manifests
├── java
│   └── com.example.sql
│       ├── MainActivity.java
│       ├── DBHelper
│       └── MainActivity
├── com.example.sql
├── com.example.sql
├── java (generated)
├── res
│   ├── drawable
│   ├── layout
│   ├── activity_main
│   ├── mipmap
│   └── values
│       ├── colors.xml
│       ├── strings.xml
│       └── themes (2)
└── res (generated)
Gradle Scripts

102 cursor = db.rawQuery("SELECT subject, people FROM lecture", selectionArgs, null);
103 while(cursor.moveToNext()) {
104     String subject_temp = cursor.getString(columnindex 0);
105     String people_temp = cursor.getString(columnindex 1);
106
107     SubjectResult += "\n" + subject_temp;
108     PeopleResult += "\n" + people_temp;
109 }
110 editSubjectResult.setText(SubjectResult);
111 editPeopleResult.setText(PeopleResult);
112 break;
113 case R.id.buttonEdit:
114     subject = editSubject.getText().toString();
115     people = editPeople.getText().toString();
116     if (subject.equals("") || people.equals("")) {
117         Toast.makeText(getApplicationContext(), "입력 실패: 빈칸을 입력하세요.", Toast.LENGTH_SHORT).show();
118     }
119     else {
120         db.execSQL("UPDATE lecture SET people=" + people + " WHERE subject=" + subject + ";");
121         Toast.makeText(getApplicationContext(), "성공적으로 수정되었음", Toast.LENGTH_SHORT).show();
122
123         SubjectResult = "수업명\n_____";
124         PeopleResult = "인원\n_____";
125         cursor = db.rawQuery("SELECT subject, people FROM lecture;", selectionArgs, null);
126         while(cursor.moveToNext()) {
127             subject = cursor.getString(columnindex 0);
128             people = cursor.getString(columnindex 1);
129
130             SubjectResult += "\n" + subject;
131             PeopleResult += "\n" + people;
132         }
133         editSubjectResult.setText(SubjectResult);
134         editPeopleResult.setText(PeopleResult);
135     }
136 break;
137 }
```



```
package com.example.sqlitetest;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Context;
import android.database.Cursor;
import android.database.SQLException;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;
```

```

class DBHelper extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "MyLecture.db";
    private static final int DATABASE_VERSION = 2;
    public DBHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }
    public void onCreate(SQLiteDatabase db) {
        db.execSQL("CREATE TABLE lecture ( _id INTEGER PRIMARY KEY AUTOINCREMENT,
subject TEXT, people Text);");
    }
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS lecture");
        onCreate(db);
    }
}

public class MainActivity extends AppCompatActivity {
    DBHelper helper;
    SQLiteDatabase db;
    EditText editSubject;
    EditText editPeople;
    EditText editSubjectResult;
    EditText editPeopleResult;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        helper = new DBHelper(this);
        try{
            db = helper.getWritableDatabase();
        } catch (SQLException ex) {
            db = helper.getReadableDatabase();
        }
        editSubject = (EditText) findViewById(R.id.editSubject);
        editPeople = (EditText) findViewById(R.id.editPeople);
        editSubjectResult = (EditText) findViewById(R.id.editSubjectResult);
        editPeopleResult = (EditText) findViewById(R.id.editPeopleResult);
    }
    public void onClick(View view) {
        String subject;
        String people;
        String SubjectResult;
    }
}

```



```

String PeopleResult;
Cursor cursor;
switch (view.getId()) {
    case R.id.buttonInit:
        db.execSQL("DROP TABLE IF EXISTS lecture");
        db.execSQL("CREATE TABLE lecture ( _id INTEGER PRIMARY KEY
AUTOINCREMENT, subject TEXT, people Text);");
        Toast.makeText(getApplicationContext(), "성공적으로 초기화되었음",
Toast.LENGTH_SHORT).show();
        editSubjectResult.setText("");
        editPeopleResult.setText("");
        break;
    case R.id.buttonInput:
        subject = editSubject.getText().toString();
        people = editPeople.getText().toString();
        if (subject.equals("") || people.equals("")) {
            Toast.makeText(getApplicationContext(), "입력 실패: 빈칸을 입력하세
요", Toast.LENGTH_SHORT).show();
        }
        else {
            SubjectResult ="수업명₩n_____";
            PeopleResult ="인원₩n_____";
            boolean activation =true;
            cursor = db.rawQuery("SELECT subject, people FROM lecture", null);
            while(cursor.moveToNext()) {
                String subject_temp = cursor.getString(0);
                if (subject.equals(subject_temp)) {
                    activation =false;
                }
            }
            if(activation ==true){
                db.execSQL("INSERT INTO lecture VALUES (null, '"+ subject +"',
''+ people + '');"");
                Toast.makeText(getApplicationContext(), "성공적으로 추가되었음",
Toast.LENGTH_SHORT).show();
            }
            else {
                Toast.makeText(getApplicationContext(), "이미 존재하는 과목입니
다.", Toast.LENGTH_SHORT).show();
            }
            cursor = db.rawQuery("SELECT subject, people FROM lecture", null);

```



```

        while(cursor.moveToNext()) {
            String subject_temp = cursor.getString(0);
            String people_temp = cursor.getString(1);
            SubjectResult += "\n" + subject_temp;
            PeopleResult += "\n" + people_temp;
        }
        editSubjectResult.setText(SubjectResult);
        editPeopleResult.setText(PeopleResult);
    }
    break;
case R.id.buttonEdit:
    subject = editSubject.getText().toString();
    people = editPeople.getText().toString();
    if (subject.equals("") || people.equals("")) {
        Toast.makeText(getApplicationContext(), "입력 실패: 빈칸을 입력하세요", Toast.LENGTH_SHORT).show();
    }
    else {
        db.execSQL("UPDATE lecture SET people='"+ people +"' WHERE subject='"+ subject +"'");
        Toast.makeText(getApplicationContext(), "성공적으로 수정되었음", Toast.LENGTH_SHORT).show();
        SubjectResult ="수업명\n_____";
        PeopleResult ="인원\n_____";
        cursor = db.rawQuery("SELECT subject, people FROM lecture;", null);
        while(cursor.moveToNext()) {
            subject = cursor.getString(0);
            people = cursor.getString(1);
            SubjectResult += "\n" + subject;
            PeopleResult += "\n" + people;
        }
        editSubjectResult.setText(SubjectResult);
        editPeopleResult.setText(PeopleResult);
    }
    break;
case R.id.buttonDel:
    subject = editSubject.getText().toString();
    if (subject.equals("")) {
        Toast.makeText(getApplicationContext(), "입력 실패: 수업명 을 입력하세요", Toast.LENGTH_SHORT).show();
    }

```

```

else{
    db.execSQL("DELETE FROM lecture WHERE subject='"+ subject + "';");
    Toast.makeText(getApplicationContext(), "성공적으로 삭제되었음",
Toast.LENGTH_SHORT).show();
    SubjectResult ="수업명\n_____";
    PeopleResult ="인원\n_____";
    cursor = db.rawQuery("SELECT subject, people FROM lecture;", null);
    while(cursor.moveToNext()) {
        subject = cursor.getString(0);
        people = cursor.getString(1);
        SubjectResult += "\n" + subject;
        PeopleResult += "\n" + people;
    }
    editSubjectResult.setText(SubjectResult);
    editPeopleResult.setText(PeopleResult);
}
break;
case R.id.buttonSearch:
    subject = editSubject.getText().toString();
    SubjectResult ="수업명\n_____";
    PeopleResult ="인원\n_____";
    if (subject.equals("")) {
        cursor = db.rawQuery("SELECT subject, people FROM lecture;", null);
        if (cursor.getCount() ==0) {
            Toast.makeText(getApplicationContext(), "데이터가 존재하지 않습
니다.", Toast.LENGTH_SHORT).show();
            break;
        }
        while(cursor.moveToNext()) {
            String subject_temp = cursor.getString(0);
            String people_temp = cursor.getString(1);
            SubjectResult += "\n" + subject_temp;
            PeopleResult += "\n" + people_temp;
        }
        Toast.makeText(getApplicationContext(), "모든 정보를 조회합니다.",
Toast.LENGTH_SHORT).show();
    }
    else{
        cursor = db.rawQuery("SELECT subject, people FROM lecture WHERE
subject='"+ subject + "';", null);
        if (cursor.getCount() ==0) {

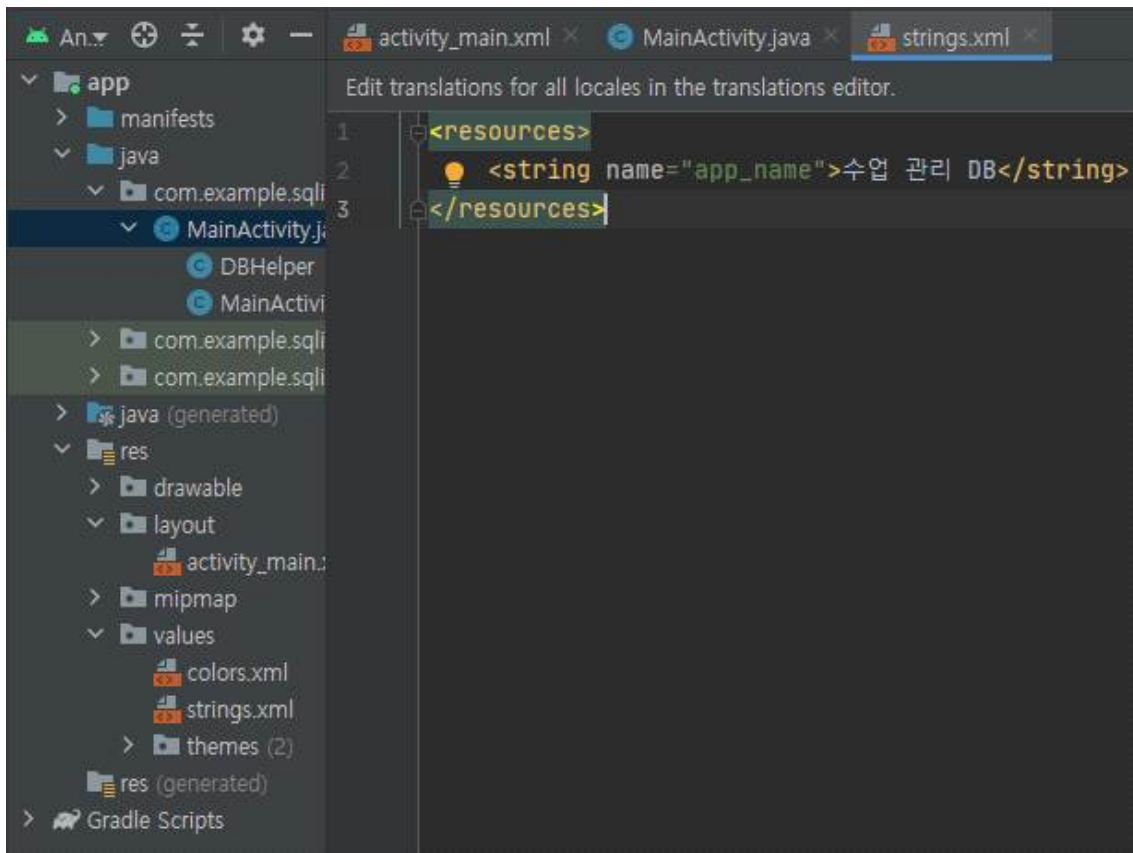
```

```

        Toast.makeText(getApplicationContext(), "데이터가 존재하지 않습
니다.", Toast.LENGTH_SHORT).show();
        break;
    }
    while(cursor.moveToNext()) {
        String subject_temp = cursor.getString(0);
        String people_temp = cursor.getString(1);
        SubjectResult += "\n" + subject_temp;
        PeopleResult += "\n" + people_temp;
    }
    Toast.makeText(getApplicationContext(), subject + " 정보를 조회합니
다.", Toast.LENGTH_SHORT).show();
    }
    editSubjectResult.setText(SubjectResult);
    editPeopleResult.setText(PeopleResult);
    break;
    }
    }
}

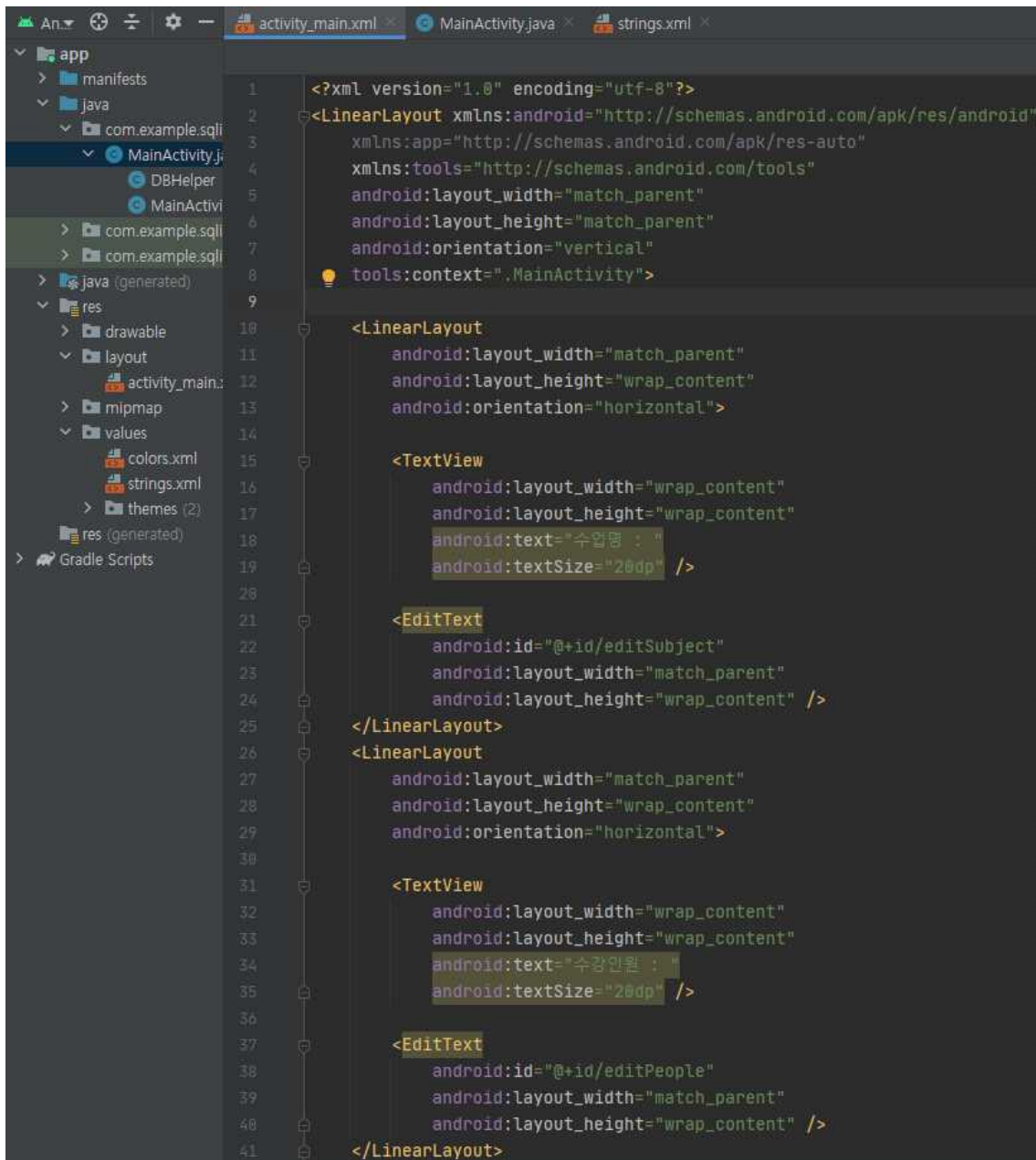
```

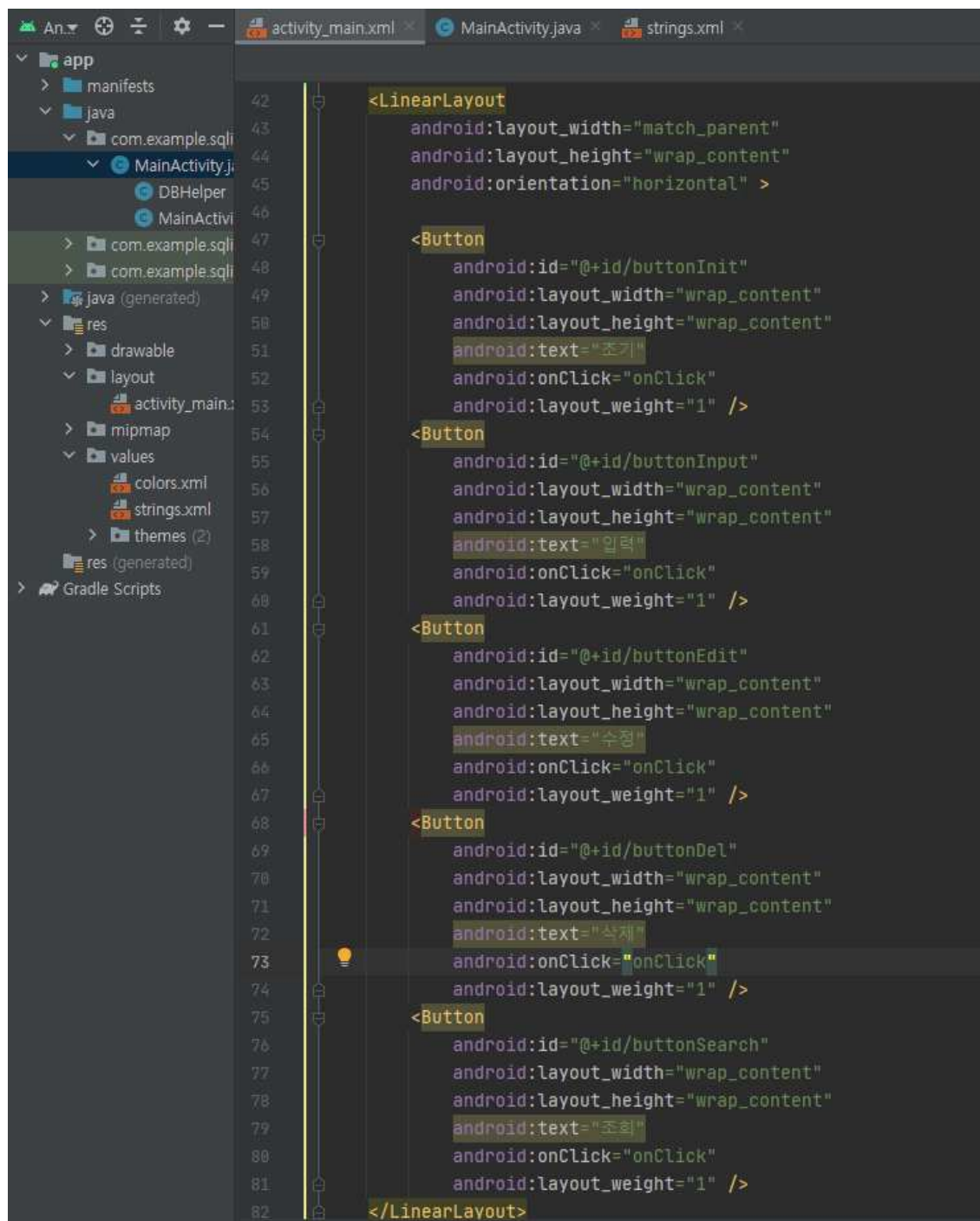
<string.xml>



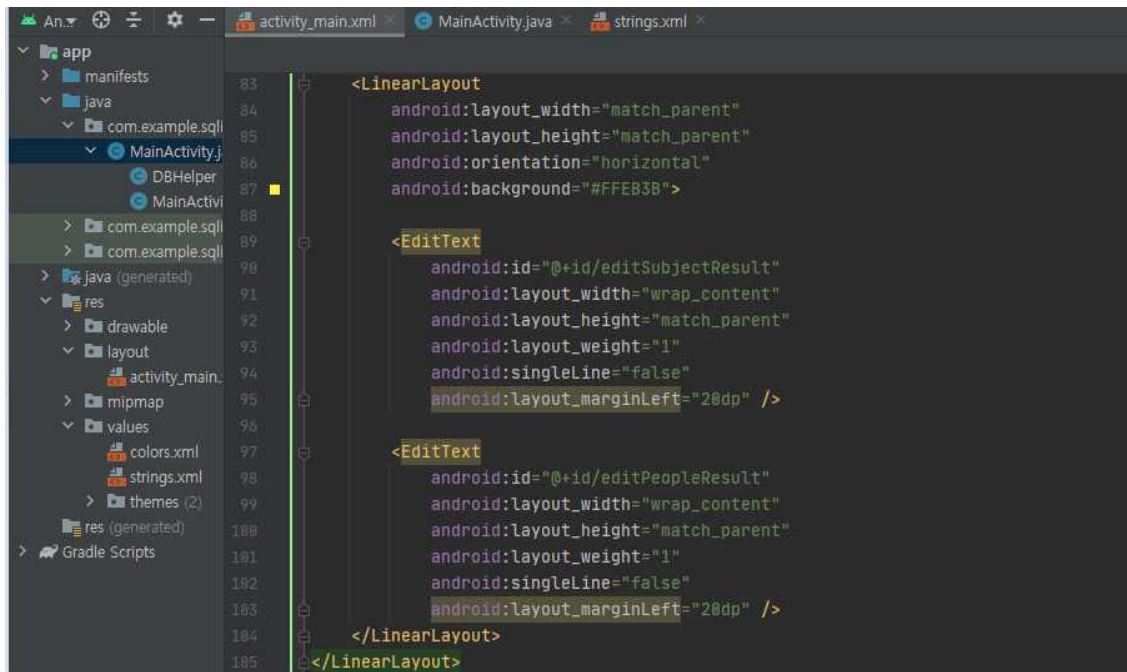
```
<resources >
    <string name="app_name">수업 관리 DB </string >
</resources >
```

## <activity\_main.xml>





```
42 <LinearLayout
43     android:layout_width="match_parent"
44     android:layout_height="wrap_content"
45     android:orientation="horizontal" >
46
47     <Button
48         android:id="@+id/buttonInit"
49         android:layout_width="wrap_content"
50         android:layout_height="wrap_content"
51         android:text="초기"
52         android:onClick="onClick"
53         android:layout_weight="1" />
54
55     <Button
56         android:id="@+id/buttonInput"
57         android:layout_width="wrap_content"
58         android:layout_height="wrap_content"
59         android:text="입력"
60         android:onClick="onClick"
61         android:layout_weight="1" />
62
63     <Button
64         android:id="@+id/buttonEdit"
65         android:layout_width="wrap_content"
66         android:layout_height="wrap_content"
67         android:text="수정"
68         android:onClick="onClick"
69         android:layout_weight="1" />
70
71     <Button
72         android:id="@+id/buttonDel"
73         android:layout_width="wrap_content"
74         android:layout_height="wrap_content"
75         android:text="삭제"
76         android:onClick="onClick"
77         android:layout_weight="1" />
78
79     <Button
80         android:id="@+id/buttonSearch"
81         android:layout_width="wrap_content"
82         android:layout_height="wrap_content"
83         android:text="조회"
84         android:onClick="onClick"
85         android:layout_weight="1" />
86 </LinearLayout>
```



```
<?xml version="1.0" encoding="utf-8"?>
```

```
<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:layout_width="match_parent"
```

```
android:layout_height="match_parent"
```

```
android:orientation="vertical"
```

```
tools:context=".MainActivity">
```

```
    <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="수업명 : "
```

```
                android:textSize="20dp"/>
```

```
            <EditText
```

```
                android:id="@+id/editSubject"
```

```
                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="수강인원 : "
android:textSize="20dp"/>
    <EditText
android:id="@+id/editPeople"
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">
    <Button
android:id="@+id/buttonInit"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="초기"
android:onClick="onClick"
android:layout_weight="1"/>
    <Button
android:id="@+id/buttonInput"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="입력"
android:onClick="onClick"
android:layout_weight="1"/>
    <Button
android:id="@+id/buttonEdit"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="수정"
android:onClick="onClick"
android:layout_weight="1"/>
    <Button
android:id="@+id/buttonDel"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
```

```
android:text="삭제"
android:onClick="onClick"
android:layout_weight="1"/>
    <Button
android:id="@+id/buttonSearch"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="조회"
android:onClick="onClick"
android:layout_weight="1"/>
    </LinearLayout >
    <LinearLayout
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="horizontal"
android:background="#FFEB3B">
        <EditText
android:id="@+id/editSubjectResult"
android:layout_width="wrap_content"
android:layout_height="match_parent"
android:layout_weight="1"
android:singleLine="false"
android:layout_marginLeft="20dp"/>
        <EditText
android:id="@+id/editPeopleResult"
android:layout_width="wrap_content"
android:layout_height="match_parent"
android:layout_weight="1"
android:singleLine="false"
android:layout_marginLeft="20dp"/>
    </LinearLayout >
</LinearLayout >
```