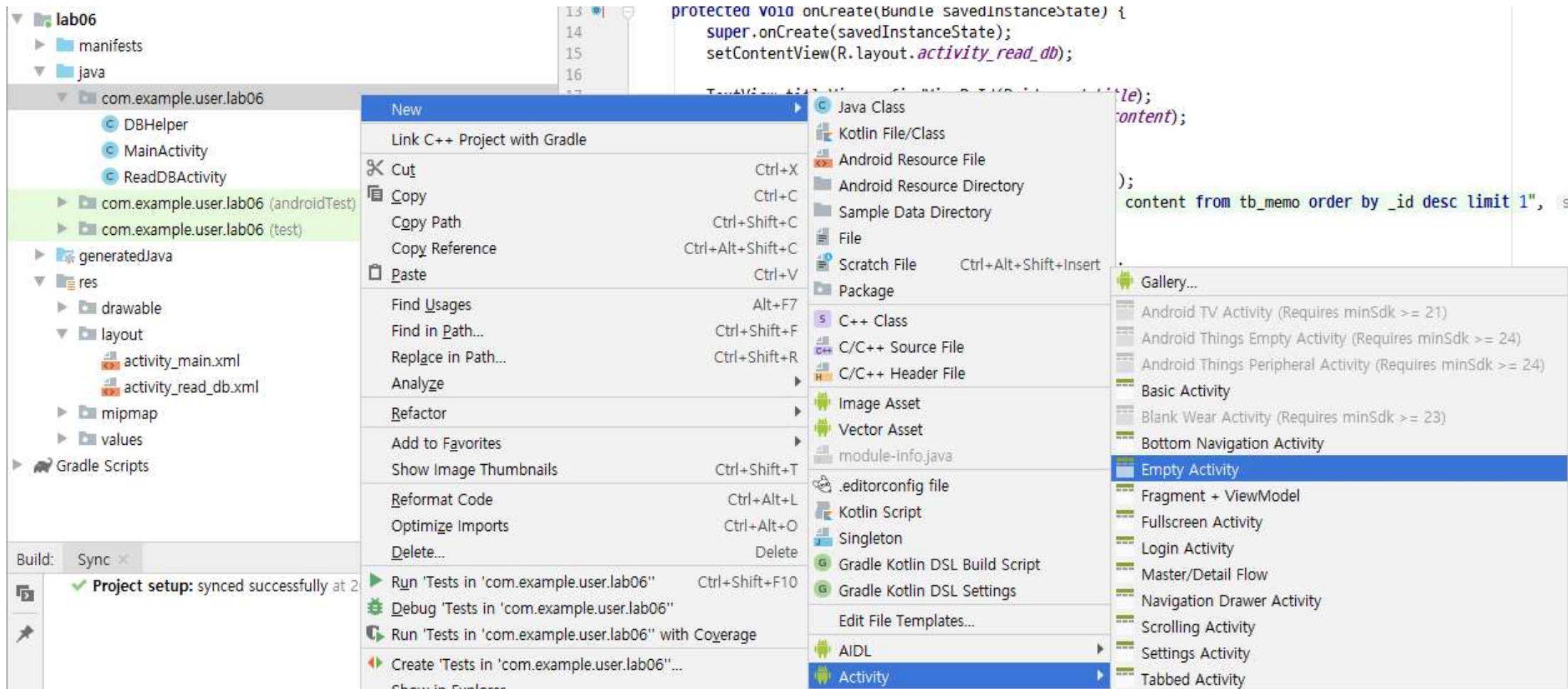




Realm 실습

Step 1 _ 액티비티 생성




New Android Activity

 **Configure Activity**
Android Studio



Creates a new empty activity



Activity Name:

Lab06_2Activity

☒ Generate Layout File

Layout Name:

activity_lab06_2

☒ Launcher Activity

Package name:

com.example.user.lab06

Source Language:

Java

If true, this activity will have a CATEGORY_LAUNCHER intent filter, making it visible in the launcher

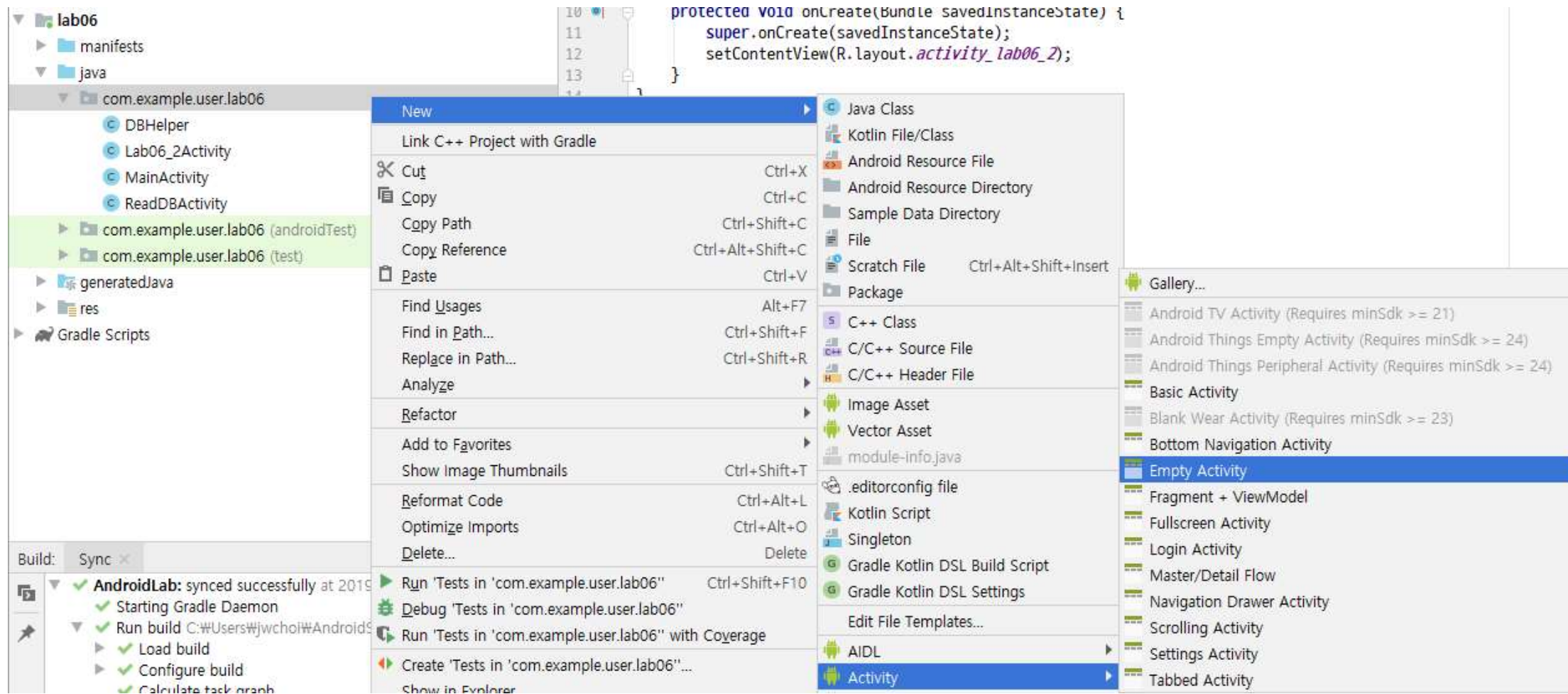
Previous

Next


Cancel


Finish

Step 2 _ RealmReadActivity 생성

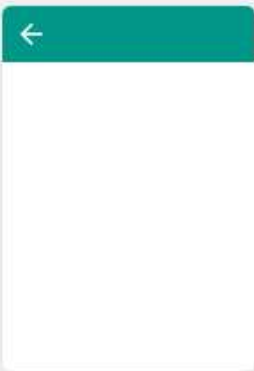


New Android Activity

 **Configure Activity**
Android Studio



Creates a new empty activity



Activity Name:

☒ Generate Layout File

Layout Name:

☐ Launcher Activity

Package name:

Source Language:

The name of the activity class to create

Previous

Next

Cancel

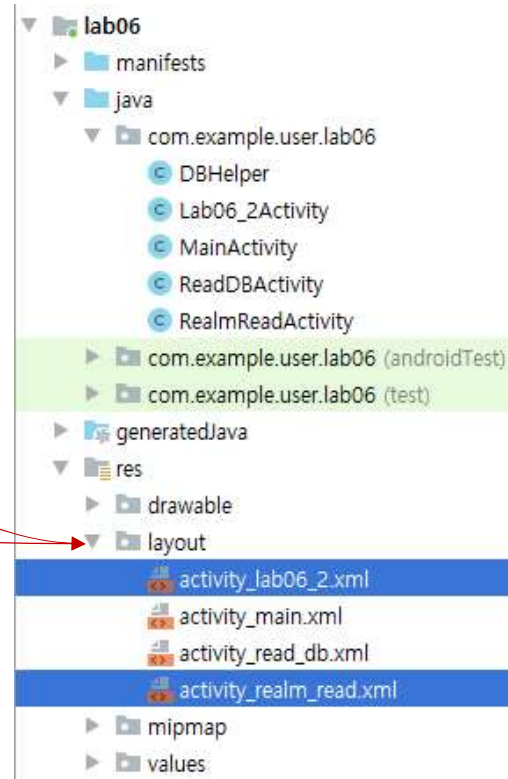
Finish

Step 3 _ 레이아웃 XML 복사

activity_lab06_2.xml
activity_realn_read.xml

Overwrite

Overwrite



Step 4 _ Realm 설정

Gradle files have changed since last project sync. A project sync may be necessary for the IDE to work properly. (3) Sync Now

```
1 // Top-level build file where you can add configuration options common to all sub-projects/modules.
2
3 buildscript {
4     repositories {
5         google()
6         jcenter()
7     }
8 }
9 dependencies {
10     classpath 'com.android.tools.build:gradle:3.4.2'
11     classpath 'io.realm:realm-gradle-plugin:3.5.0' (2)
12     // NOTE: Do not place your application dependencies here; they belong
13     // in the individual module build.gradle files
14 }
15
16 allprojects {
17     repositories {
18         google()
19         jcenter()
20     }
21 }
22
23 task clean(type: Delete) {
24     delete rootProject.buildDir
25 }
26
27
28
```

플러그인 설치

buildscript() > dependencies()

Android

- layout
 - activity_lab06_2.xml
 - activity_main.xml
 - activity_read_db.xml
 - activity_realm_read.xml
- mipmap
- values
- Gradle Scripts
 - build.gradle (Project: AndroidLab)
 - build.gradle (Module: app)
 - build.gradle (Module: lab01)
 - build.gradle (Module: lab02)
 - build.gradle (Module: lab03)
 - build.gradle (Module: lab04)
 - build.gradle (Module: lab05)
 - build.gradle (Module: lab06)** ①
 - gradle-wrapper.properties (Gradle Version)
 - proguard-rules.pro (ProGuard Rules for app)
 - proguard-rules.pro (ProGuard Rules for lab01)
 - proguard-rules.pro (ProGuard Rules for lab02)
 - proguard-rules.pro (ProGuard Rules for lab03)
 - proguard-rules.pro (ProGuard Rules for lab04)
 - proguard-rules.pro (ProGuard Rules for lab05)
 - proguard-rules.pro (ProGuard Rules for lab06)
 - gradle.properties (Project Properties)
 - settings.gradle (Project Settings)

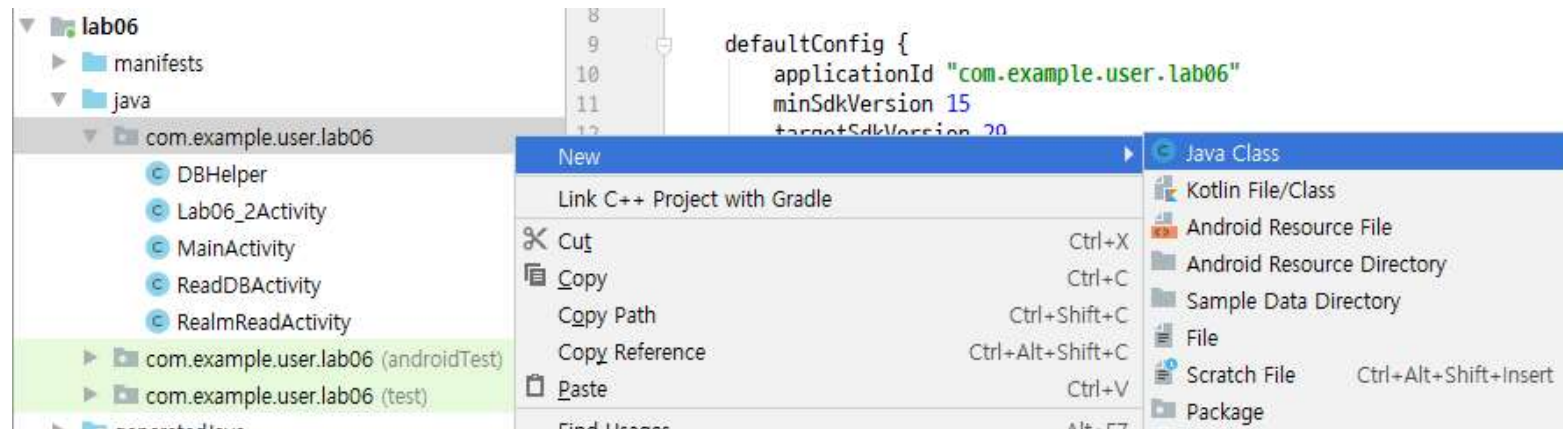
lab06


Gradle files have changed since last project sync. A project sync may be necessary for the IDE to work properly. ③ Sync Now

```
1  apply plugin: 'com.android.application'
2  apply plugin: 'realm-android' ②
3
4  android {
5      compileSdkVersion 29
6      buildToolsVersion "29.0.0"
7
8
9      defaultConfig {
10         applicationId "com.example.user.lab06"
11         minSdkVersion 15
12         targetSdkVersion 29
13         versionCode 1
14         versionName "1.0"
15
16         testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
17     }
18 }
19
20 buildTypes {
21     release {
22         minifyEnabled false
23         proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
24     }
25 }
26
27 }
28
29 dependencies {
30     implementation fileTree(dir: 'libs', include: ['*.jar'])
31 }
```

플러그인 사용 준비

Step 5 _ MemoVO.java 작성



 Create New Class ✕

Name:

MemoVO

Kind:

Class

Superclass:

io.realm.RealObject

Interface(s):

Package:

com.example.user.lab06

Visibility:

☒ Public

☐ Package Private

Modifiers:

☒ None

☐ Abstract

☐ Final

☐ Show Select Overrides Dialog

OK

Cancel

Help

```
public class MemoV0 extends RealmObject {  
    public String title;  
    public String content;  
}
```

Step 6 _ Lab06_2Activity.java 작성

```
public class Lab06_2Activity extends AppCompatActivity implements View.OnClickListener {
```

```
    EditText titleView;  
    EditText contentView;  
    Button addBtn;
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_lab06_2);  
  
        titleView = findViewById(R.id.realm_add_title);  
        contentView = findViewById(R.id.realm_add_content);  
        addBtn = findViewById(R.id.realm_add_btn);  
  
        addBtn.setOnClickListener(this);  
    }
```

```
    @Override
```

```
    public void onClick(View v) {  
        final String title = titleView.getText().toString();  
        final String content = contentView.getText().toString();
```

```
        Realm.init(this);  
        Realm mRealm = Realm.getDefaultInstance();  
        mRealm.executeTransaction(new Realm.Transaction() {
```

```
            @Override
```

```
            public void execute(Realm realm) {  
                MemoVO vo = realm.createObject(MemoVO.class);  
                vo.title = title;  
                vo.content = content;
```

```
            }  
        });
```

```
        Intent intent = new Intent(this, RealmReadActivity.class);  
        intent.putExtra("title", title);  
        startActivity(intent);
```

```
    }  
}
```

Step 7 _ RealmReadActivity.java 작성

```
public class RealmReadActivity extends AppCompatActivity {

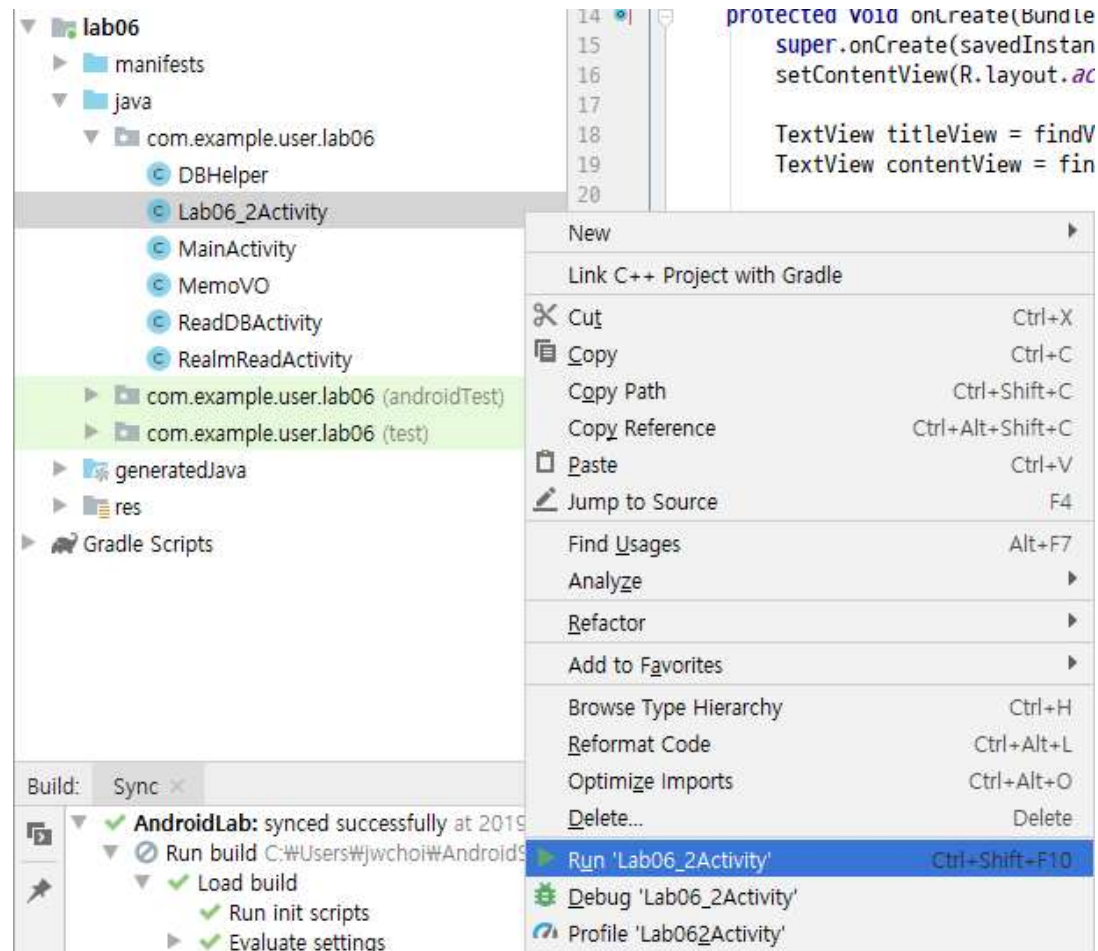
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_realm_read);

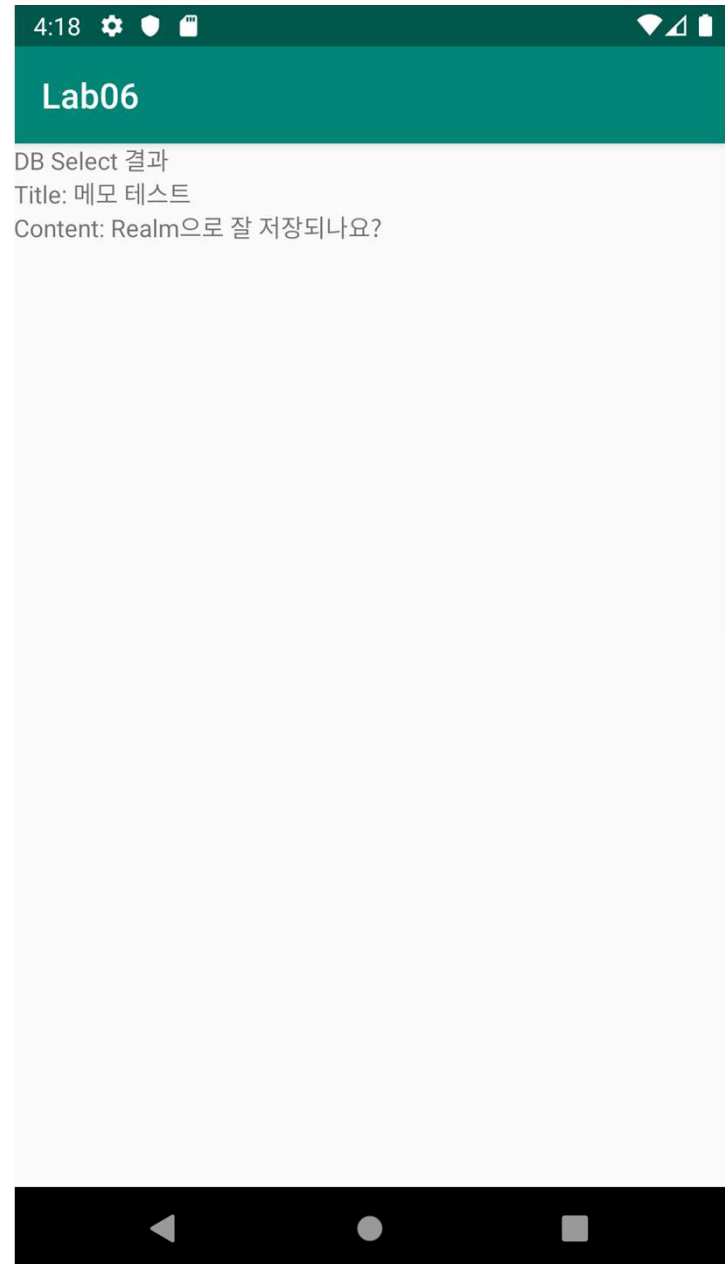
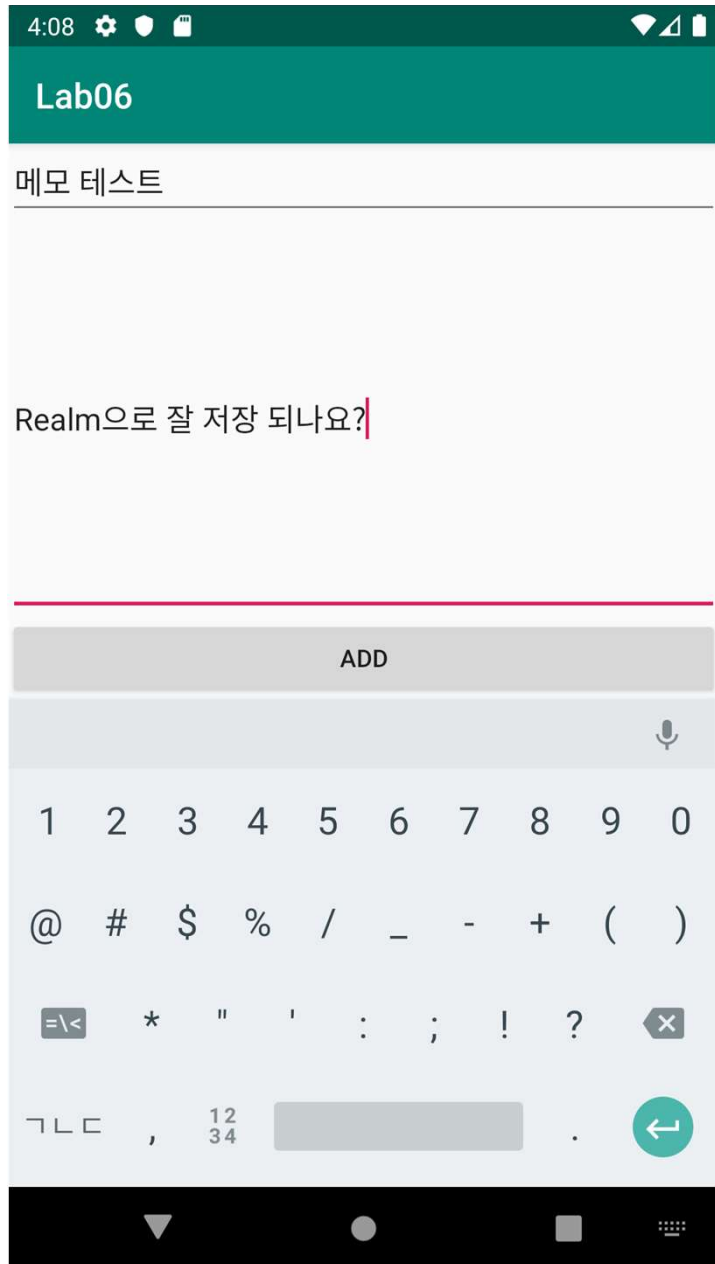
        TextView titleView = findViewById(R.id.realm_read_title);
        TextView contentView = findViewById(R.id.realm_read_content);

        Intent intent = getIntent();
        String title = intent.getStringExtra("title");

        Realm mRealm = Realm.getDefaultInstance();
        MemoVO vo = mRealm.where(MemoVO.class).equalTo("title", title).findFirst();
        titleView.setText(vo.title);
        contentView.setText(vo.content);
    }
}
```

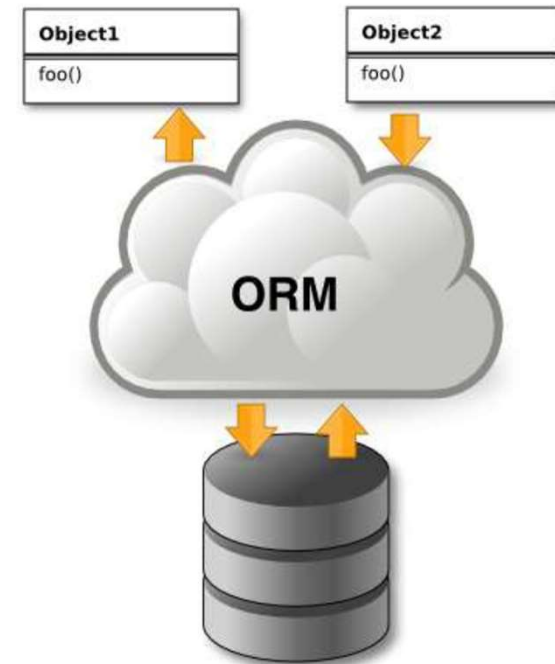

Step 8 _ Lab06_2Activity.java 실행





Realm 소개

- <https://realm.io>에서 오픈소스로 만들어지고 있는 데이터베이스
 - 안드로이드 표준에서 제공하는 것은 아님
 - 앱에서 이용하려면 플러그인 방식으로 앱에 추가해야 사용할 수 있음.
 - SQLite와 비교했을 때 가장 큰 차이
 - ORM(Object-Relational Mappings)을 제공한다는 것
 - 데이터 영속화를 위해 SQL 문을 사용하지 않아도 됨



Realm이 관리할 VO(Value-Object) 클래스

```
public class MemoVO extends RealmObject {  
    public String title;  
    public String content;  
}
```

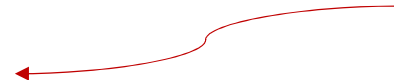
```
public class MemoVO extends RealmObject {  
    private String title;  
    private String content;  
  
    public String getTitle() {  
        return title;  
    }  
  
    public void setTitle(String title) {  
        this.title = title;  
    }  
  
    public String getContent() {  
        return content;  
    }  
  
    public void setContent(String content) {  
        this.content = content;  
    }  
}
```

Realm 객체 획득

static void

`init`(android.content.Context context)

Initializes the Realm library and creates a default configuration that is ready to use.



```
Realm.init(this);
```

```
Realm mRealm = Realm.getDefaultInstance();
```


VO 객체의 데이터 저장

```
mRealm.executeTransaction(new Realm.Transaction() {  
    @Override  
    public void execute(Realm realm) {  
        MemoVO vo = realm.createObject(MemoVO.class);  
        vo.setTitle(title);  
        vo.setContent(content);  
    }  
});
```

저장된 데이터의 획득

```
MemoVO vo = mRealm.where(MemoVO.class).equalTo("title", title).findFirst();
```

```
<E extends RealmModel>  
RealmQuery<E>
```

```
where(Class<E> clazz)
```

```
Returns a typed RealmQuery, which can be used to query for specific objects of this type
```

```
RealmResults<MemoVO> results = mRealm.where(MemoVO.class).equalTo("title", "Tiger").findAll();
```

```
java.lang.Object  
java.util.AbstractCollection<E>  
java.util.AbstractList<E>  
io.realm.RealmResults<E>
```

VO 객체의 데이터 수정

```
mRealm.executeTransaction(new Realm.Transaction() {  
    @Override  
    public void execute(Realm realm) {  
        MemoVO vo = realm.where(MemoVO.class).equalTo("title", title).findFirst();  
        vo.setTitle(title);  
        vo.setContent(content);  
    }  
});
```

획득한 VO 객체의 데이터 삭제

```
MemoVO vo = mRealm.where(MemoVO.class).equalTo("title", title).findFirst();  
vo.deleteFromRealm();
```

void	<code>deleteFromRealm()</code> Deletes the object from the Realm it is currently associated to.
------	--

특정 VO 타입의 모든 데이터 삭제

```
mRealm.delete(MemoVO.class);
```