

Week07.

비디오 재생

개발환경 구축 절차

2

주 차	수 업 내 용
1	수업 소개
2	개발 환경 구축과 맛보기 프로젝트
3	텍스트 출력과 레이아웃
4	이미지의 출력
5	이벤트 처리와 액티비티 간 이동
6	오디오 재생
7	비디오 재생
8	중간고사
9	애니메이션
10	사물인터넷과 센서 – 터치 센서, 모션 센서
11	사물인터넷과 센서 – 위치 센서, 환경 센서
12	NFC 활용
13	공공 DB 오픈 API 활용
14	구글 맵과 위치 추적
15	기말 고사




강의 자료-<https://github.com/hopypark>


3

hopypark

GitHub, Inc. [US] | <https://github.com/hopypark>

나의 북마크 | 'Spring' 카테고리의 | 스프링+MyBatis+My | VOA 한국어 | Variability vs. Compl | 조승연의 굿모닝팝스 | 고급두뇌를 위한 하 | Time Series Forecast | Calculation of Inform

 Search or jump to... Pull requests Issues Marketplace Explore



hopypark

Add a bio

Edit profile

Overview Repositories 5 Stars 0 Followers 0 Following 0

Pinned repositories

Customize your pinned repositories

Lecture2018

Intro_ML


Jupyter Notebook

142 contributions in the last year

Contribution settings

	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep
Mon												
Wed												
Fri												

Learn how we count contributions.

Less  More

Contribution activity

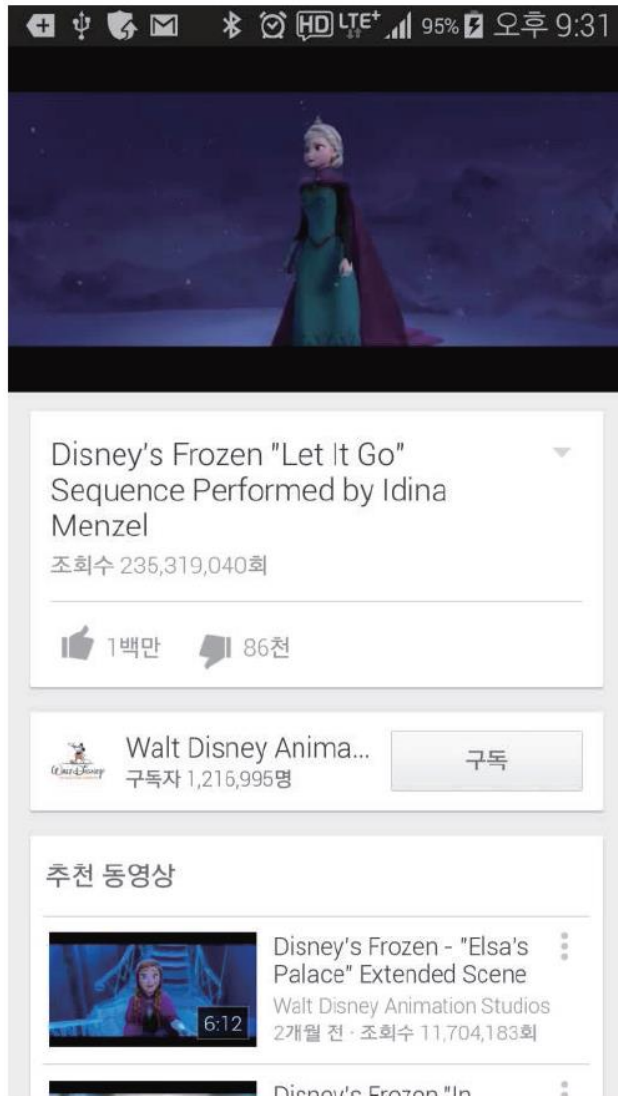
Jump to 2018

October 2018

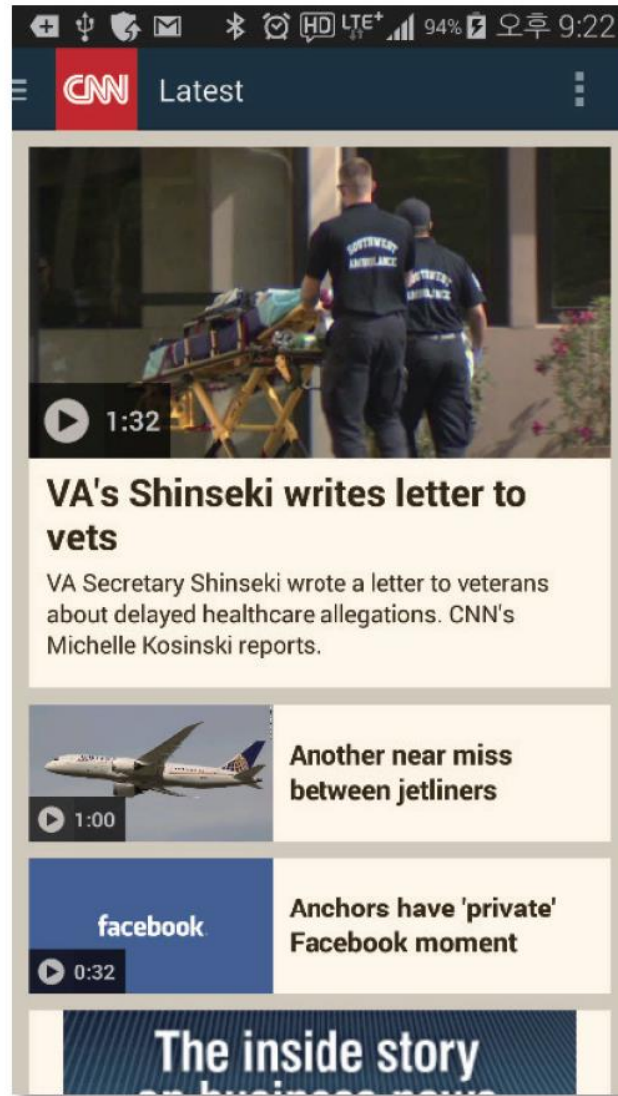
2017

비디오 재생 앱의 예

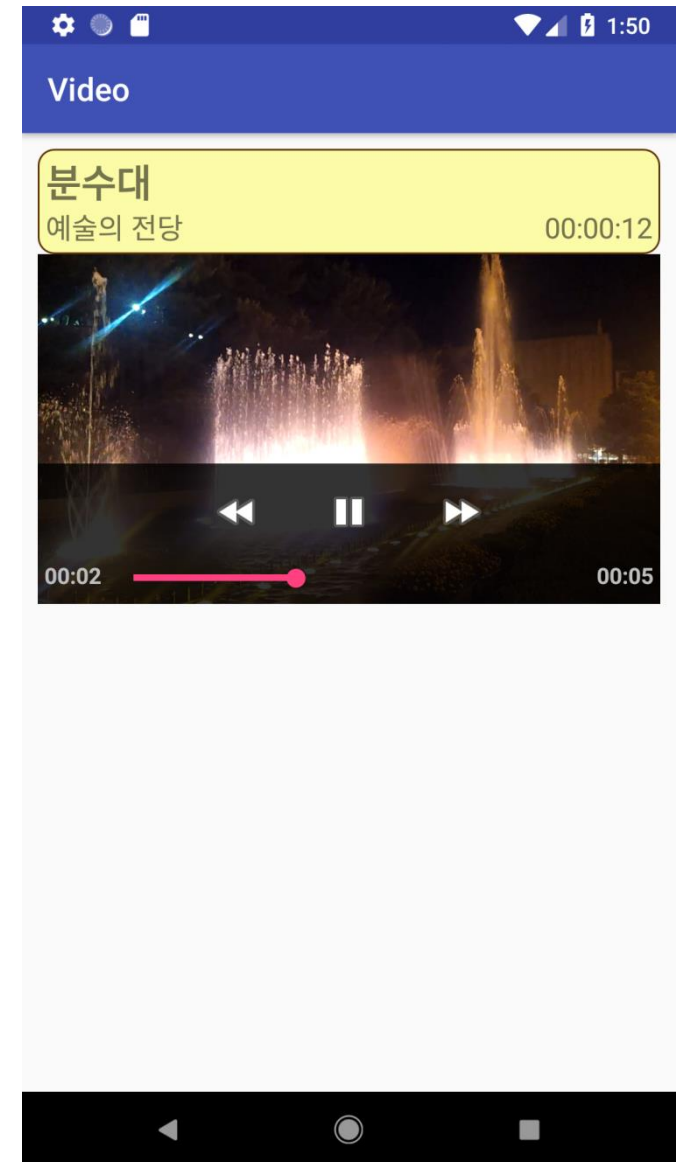
4



(a) YouTube 동영상



(b) CNN Video



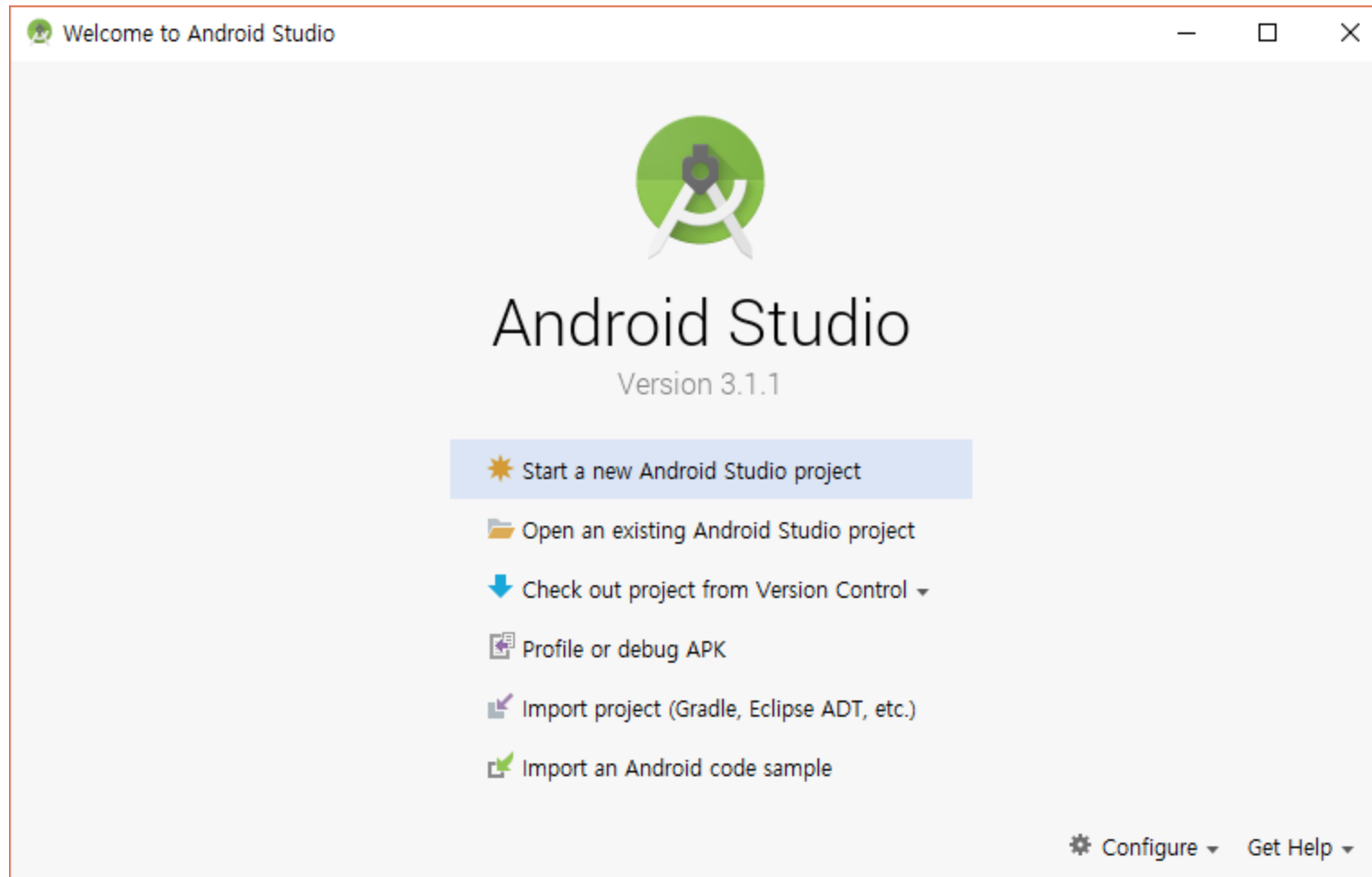
오디오 재생 원리

5



Start a new Android Studio project-type1

6



Create Android Project

8

Create New Project

Create Android Project

Application name
Video

Company domain
kyungtae.example.com

Project location
C:\Users\Kyungtae\AndroidStudioProjects\ktpark\Video

☐ Include C++ support
☐ Include Kotlin support

Previous Next Cancel Finish

Application name: Video

Project location: C:\Users\user00\AndroidStudioProjects\ktpark\Video

Create Android Project

9

Project location

C:\Users\Kyungtae\AndroidStudioProjects\ktpark\Video

Project location: C:\Users\User00\AndroidStudioProjects\ktpark\Video

☐ Include C++ support


☐ Include Kotlin support

Previous Next Cancel Finish

Target Android Devices

11

Create New Project



Target Android Devices

Select the form factors and minimum SDK

Some devices require additional SDKs. Low API levels target more devices, but offer fewer API features.

☒ Phone and Tablet

API 27: Android 8.1 (Oreo)

By targeting **API 27 and later**, your app will run on < 1% of devices. [Help me choose](#)

☐ Include Android Instant App support

☐ Wear

API 21: Android 5.0 (Lollipop)

☐ TV

API 21: Android 5.0 (Lollipop)

☐ Android Auto

☐ Android Things

API 24: Android 7.0 (Nougat)

Previous

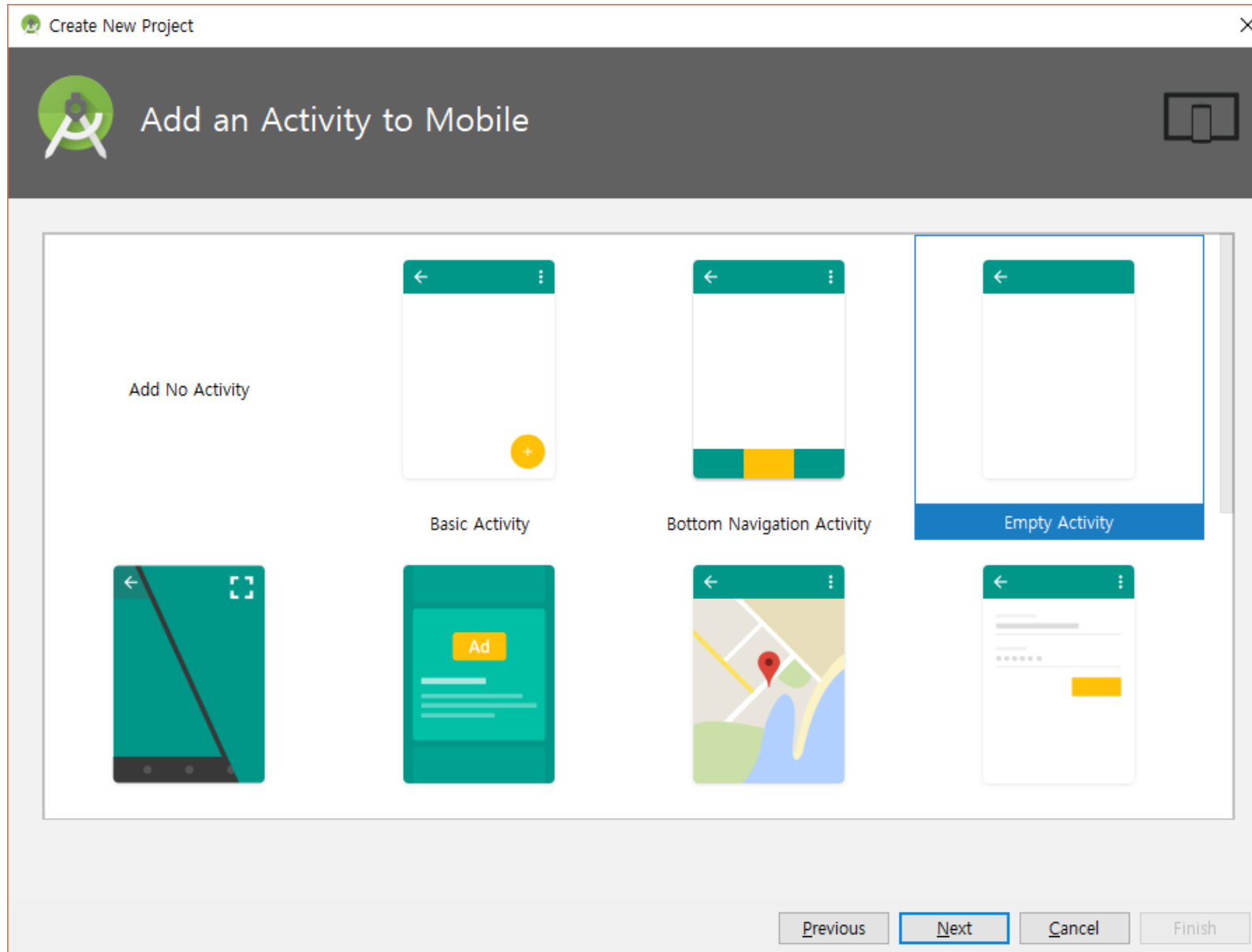
Next

Cancel

Finish

Add an Activity to Mobile



12



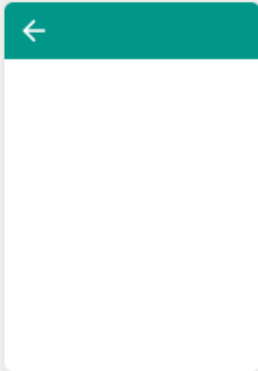
Configure Activity

13

Create New Project

Configure Activity

Creates a new empty activity



Activity Name:

☒ Generate Layout File

Layout Name:

☒ Backwards Compatibility (AppCompat)

The name of the activity class to create

Previous

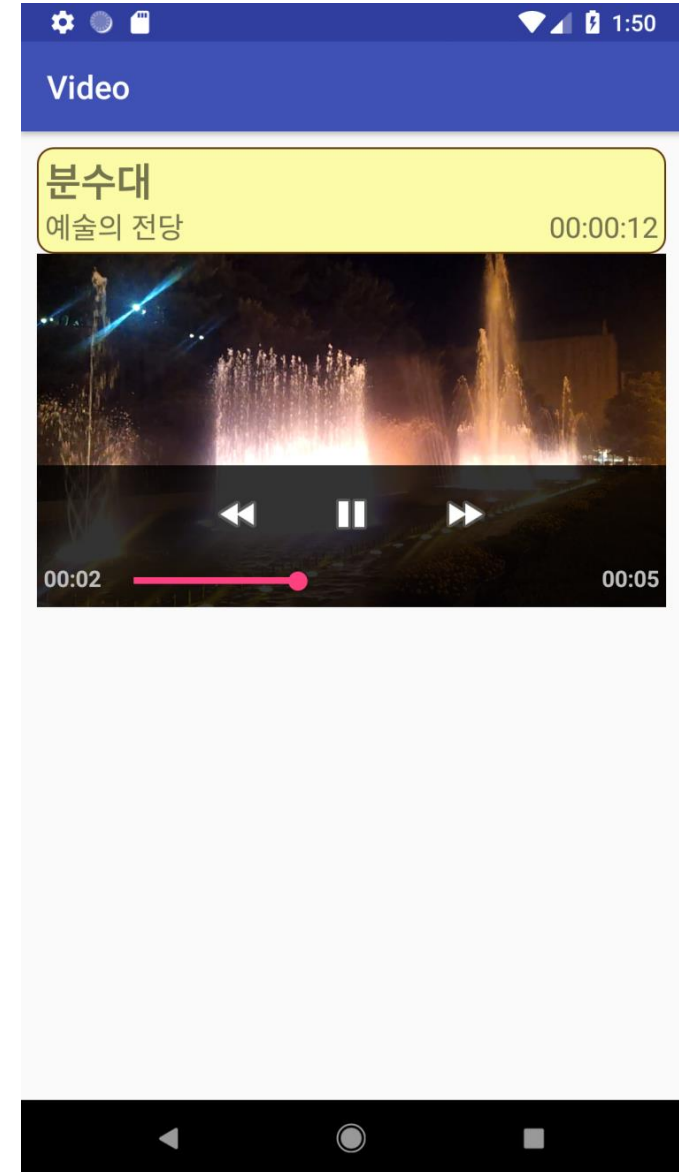
Next

Cancel

Finish

Step 0.프로젝트 개요

14



Step 1. 프로젝트 생성

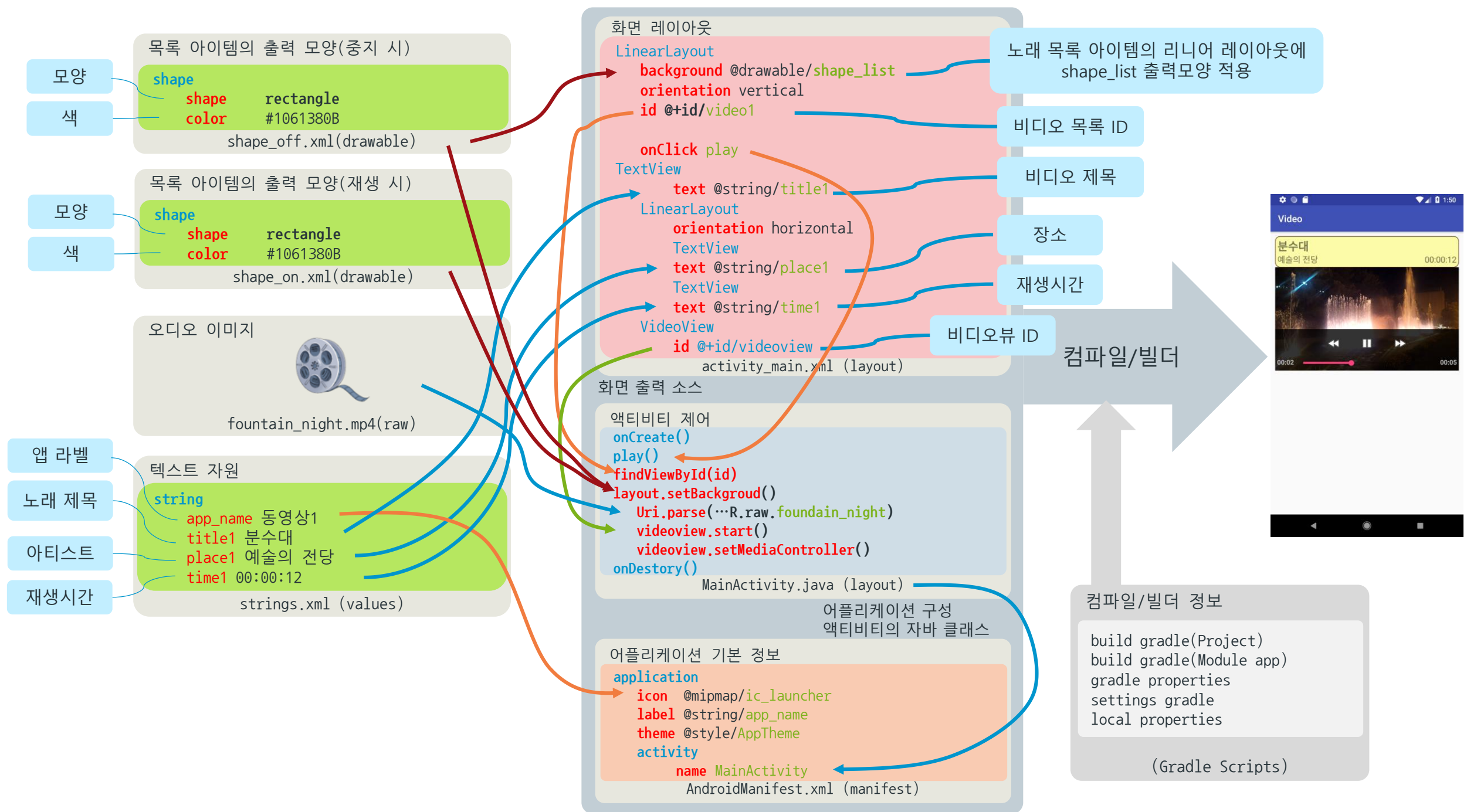
15

절차	내 용
①프로젝트 시작	메뉴에서 ‘ File → New Project ’ 클릭
②프로젝트 구성	Application Name: Video
	Company Domain: 사용자계정.example.com (디폴트 사용)
	Project location: ~\user00\AndroidStudioProject\ktpark\Video
③제품형태	Phone and Tablet (사용할 안드로이드 버전 지정: Android 8.1 Oreo)
④액티비티 유형	Empty Activity
⑤파일 옵션	Activity Name: MainActivity (디폴트 사용)
	Layout Name: activity_main (디폴트 사용)

Step 2. 파일 편집

16

모듈	폴더	소스 파일	편집 내용
manifests		AndroidManifest.xml	
java	com.example.사용자계정.video	MainActivity.java	<ul style="list-style-type: none"> 비디오 목록 출력과 비디오 자동 재생
res	drawable	shape_off.xml	<ul style="list-style-type: none"> 목록 아이템에 대한 출력 스타일 설계 (테두리, 패딩, 모서리) - 중지 시
		shape_on.xml	<ul style="list-style-type: none"> 목록 아이템에 대한 출력 스타일 설계 (테두리, 패딩, 모서리) - 재생시
	layout	activity_main.xml	<ul style="list-style-type: none"> 비디오 목록의 화면 배치 목록 아이템에 출력 모양 적용 (shape_list.xml)
	mipmap	ic_launcher.png	
	raw	fountain_night.mp4	<ul style="list-style-type: none"> 비디오 파일
	values	colors.xml	
		dimens.xml	<ul style="list-style-type: none"> 화면의 구성자원 크기(여백, 글자크기 등)
		strings.xml	<ul style="list-style-type: none"> 어플리케이션 라벨 비디오에 대한 제목, 제작자, 재생시간에 대한 텍스트 리소스 정의
		styles.xml	



Step 2.1 이미지 파일 복사

18

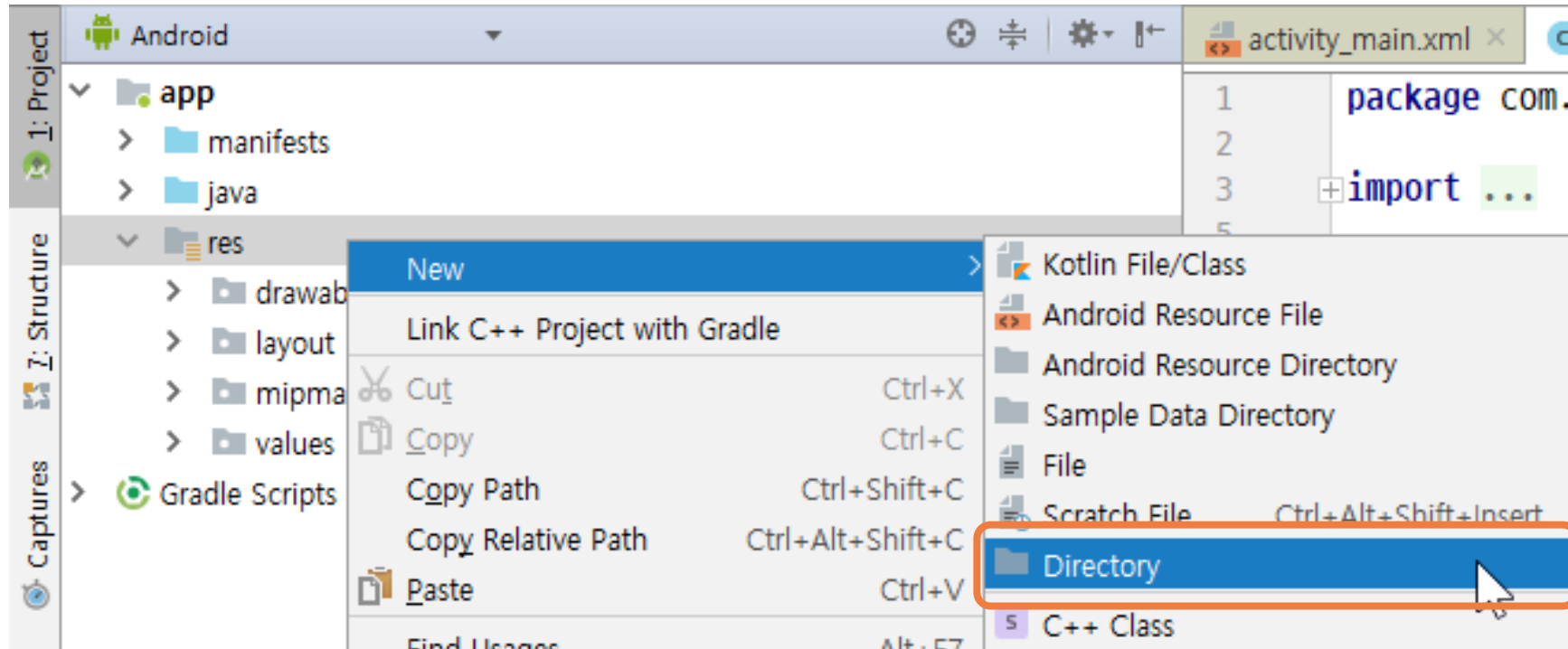
- res 폴더에 있는 **raw** 폴더에 fountain_night.mp4 파일 저장

모듈	폴더	소스 파일	내용
res	raw	fountain_night.mp4	동영상 파일

res/raw 폴더에
비디오 파일 올리기

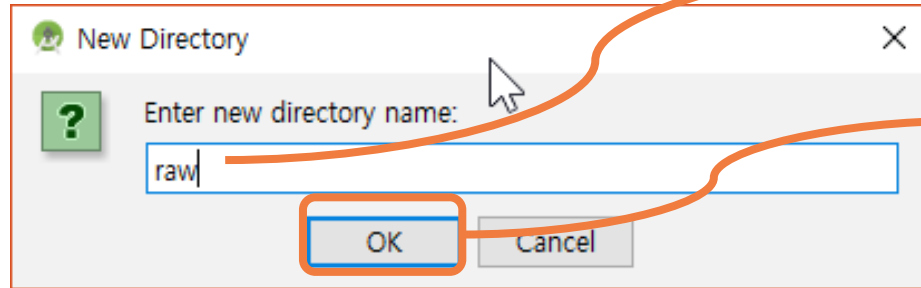
drawable/raw 폴더에 오디오 파일 추가하기

- app→res→New→Directory 클릭



res/New/Directory 클릭

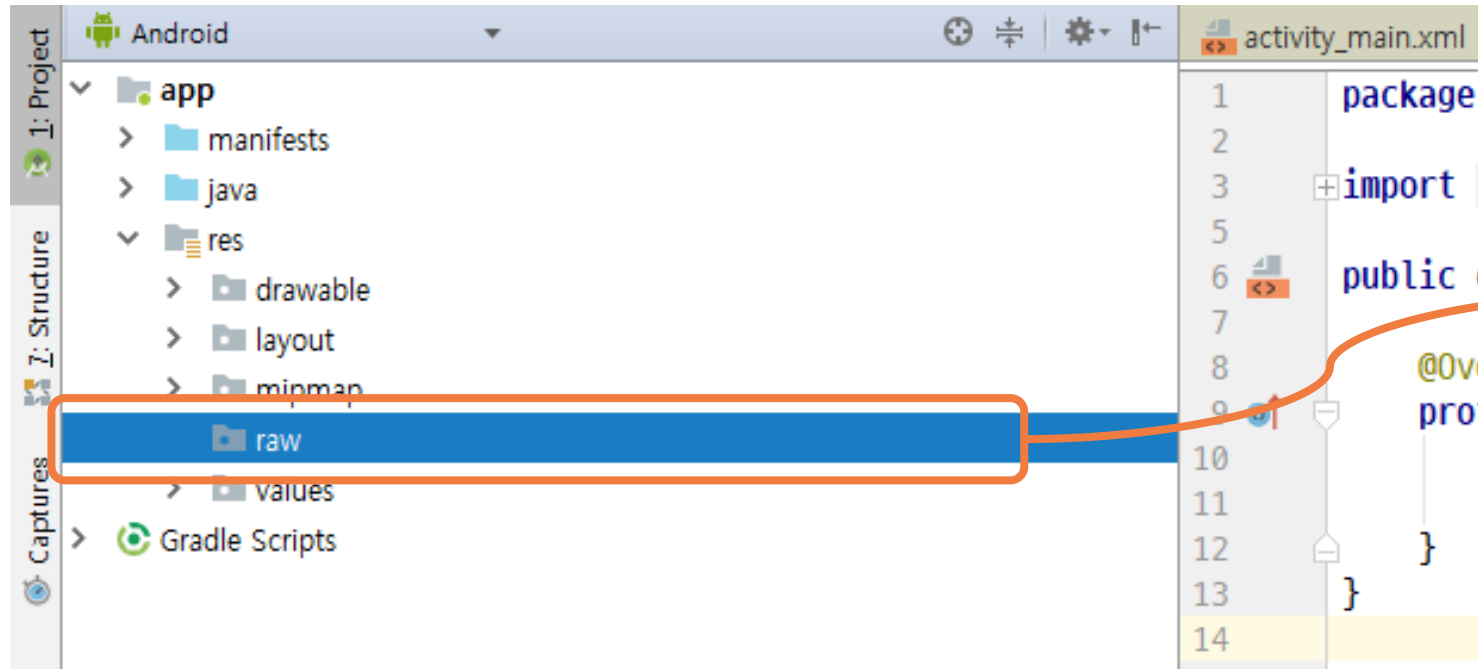
- 폴더 이름 작성



폴더 이름:raw

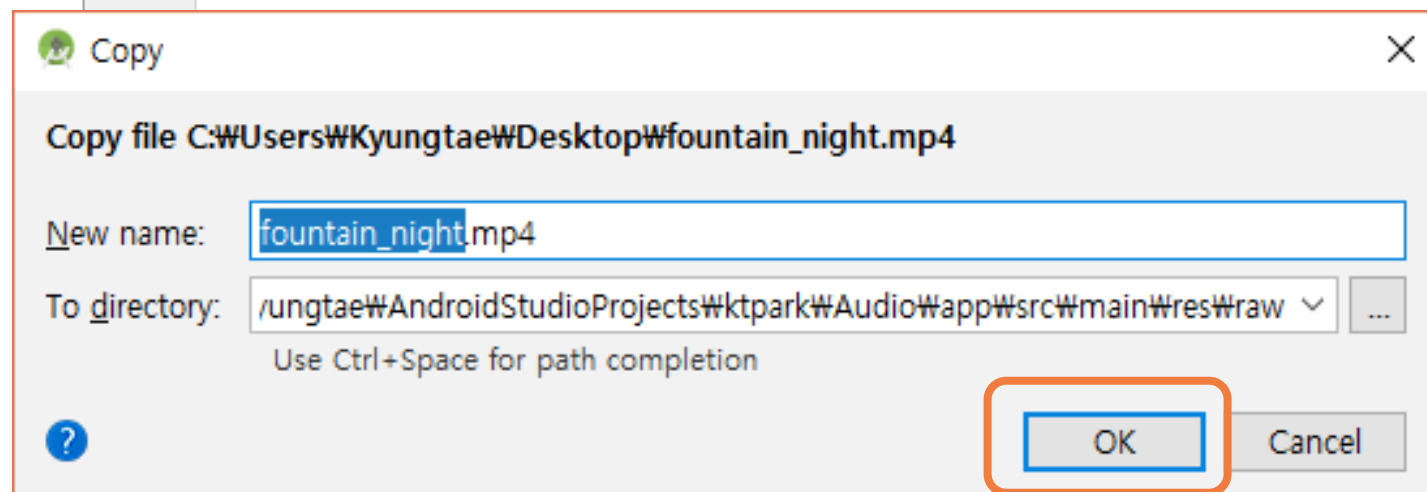
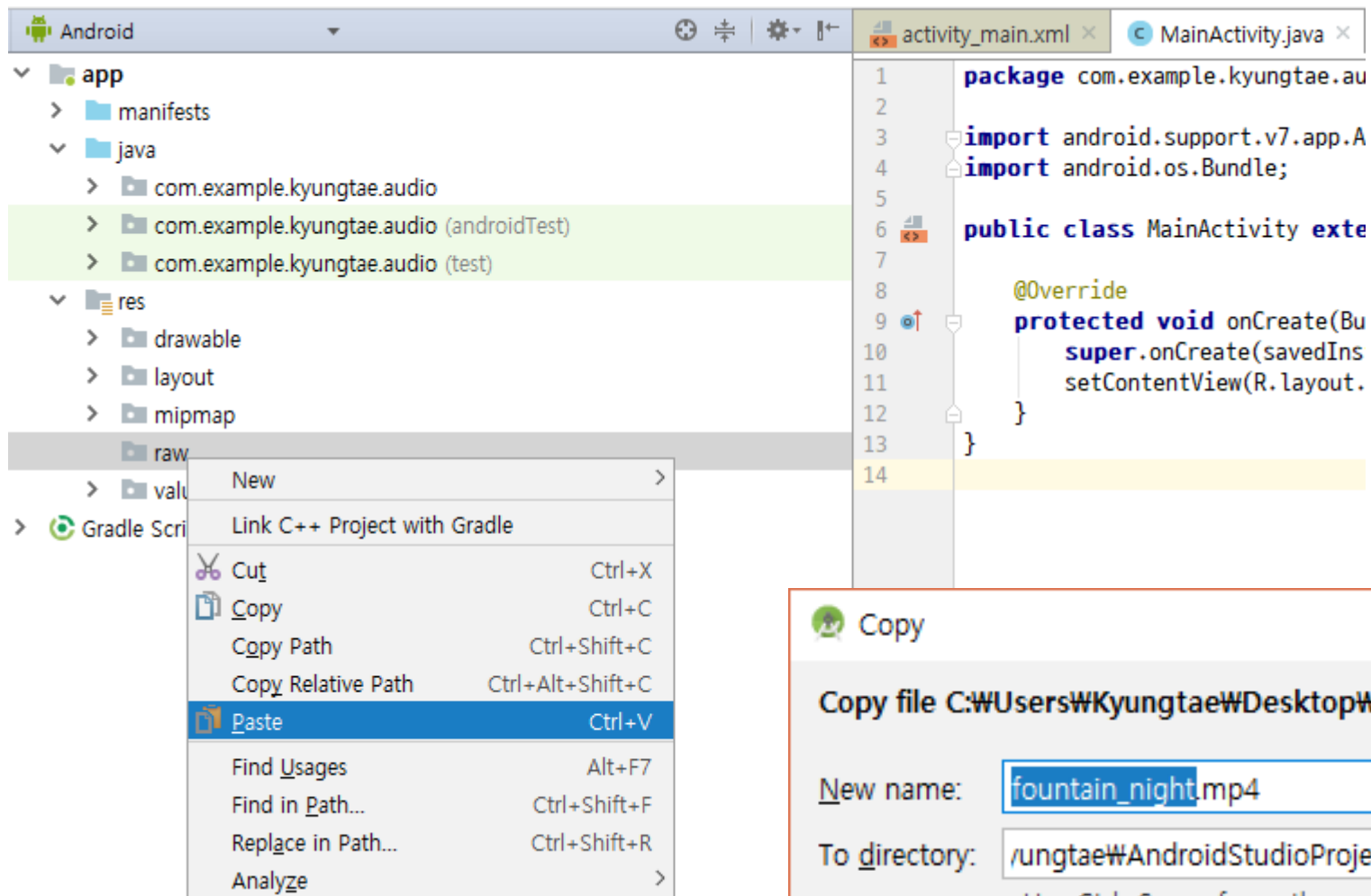
OK 클릭

- 실행 결과

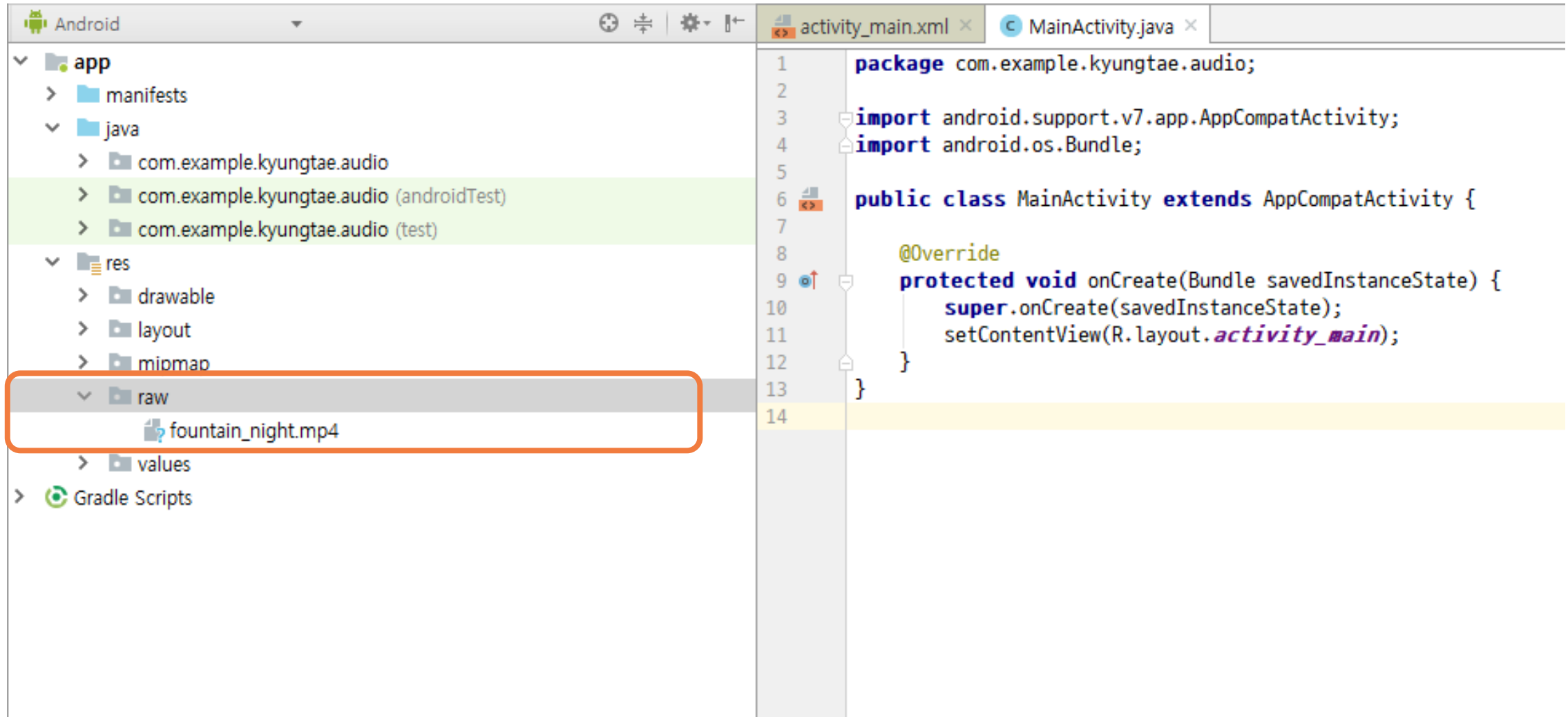


raw 폴더 생성됨

fountain_night.mp4 추가하기



fountain_night.mp4



The screenshot shows the Android Studio IDE interface. On the left, the 'Project' view displays the file structure of an Android application. The 'app' directory is expanded, showing 'manifests', 'java', 'res', and 'Gradle Scripts'. The 'res' directory is further expanded, showing 'drawable', 'layout', 'miomap', 'raw', and 'values'. The 'raw' directory is highlighted with an orange border, and the file 'fountain_night.mp4' is visible within it. On the right, the 'MainActivity.java' file is open in the editor. The code defines a package, imports necessary classes, and implements the MainActivity class, which extends AppCompatActivity and overrides the onCreate method to call setContentView(R.layout.activity_main).

```
1 package com.example.kyungtae.audio;
2
3 import android.support.v7.app.AppCompatActivity;
4 import android.os.Bundle;
5
6 public class MainActivity extends AppCompatActivity {
7
8     @Override
9     protected void onCreate(Bundle savedInstanceState) {
10         super.onCreate(savedInstanceState);
11         setContentView(R.layout.activity_main);
12     }
13 }
14
```

Step 2.2 텍스트 자원의 편집

- strings.xml



Step 2.3 Drawable Resource 추가 및 편집

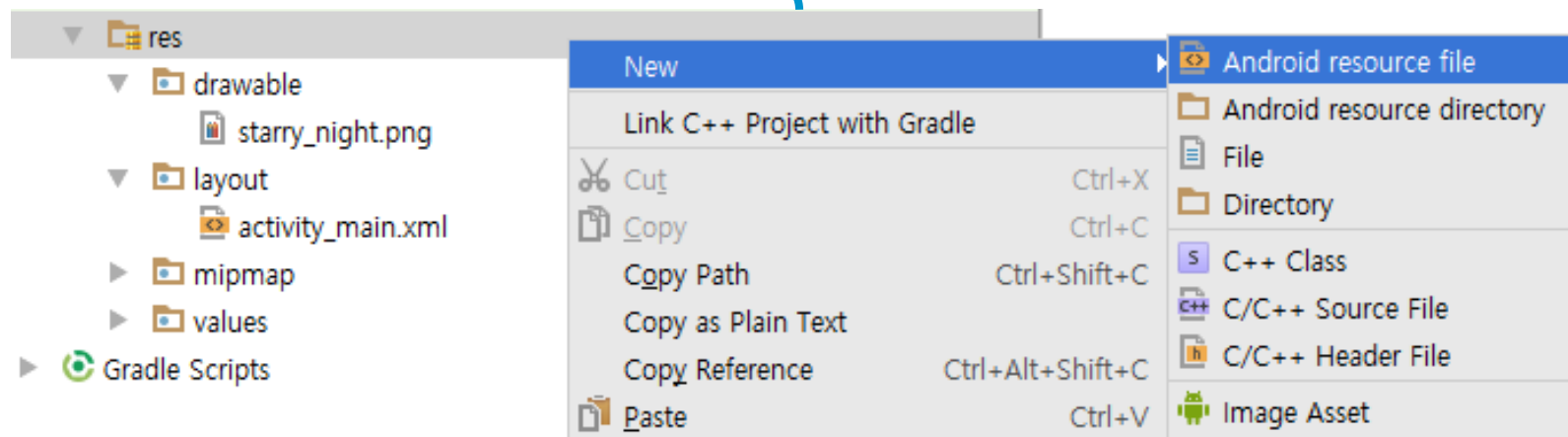
27

- **shape_list.xml** 생성(res/drawable 폴더)
 - drawable resource를 이용한 그림 출력

XML 파일 생성

분수대
예술의 전당

00:00:12



• Set New Resource File

File name: shape_on

Resource type: Drawable

Root element: shape

Source set: main

Directory name: drawable

New Resource File

File name: shape_on

Resource type: Drawable

Root element: shape

Source set: main

Directory name: drawable

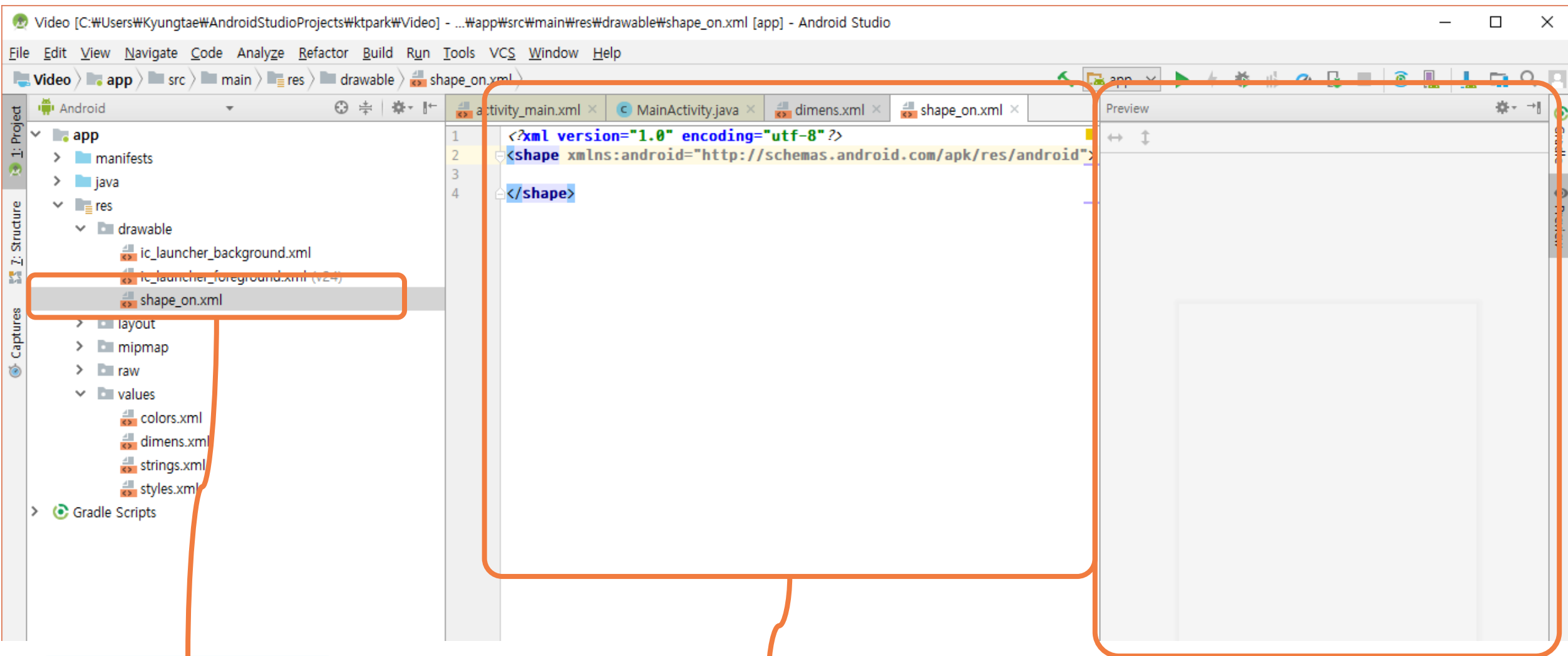
Available qualifiers:

- Country Code
- Network Code
- Locale
- Layout Direction
- Smallest Screen Width
- Screen Width
- Screen Height
- Size
- Ratio
- Orientation
- UI Mode

Chosen qualifiers:

Nothing to show

OK Cancel



shape_on.xml 파일

shape_on.xml 파일의
텍스트 코딩 영역

shape_on 파일에 의한
shape 미리보기 영역

• shape_on.xml 소스(동영상 재생 시)

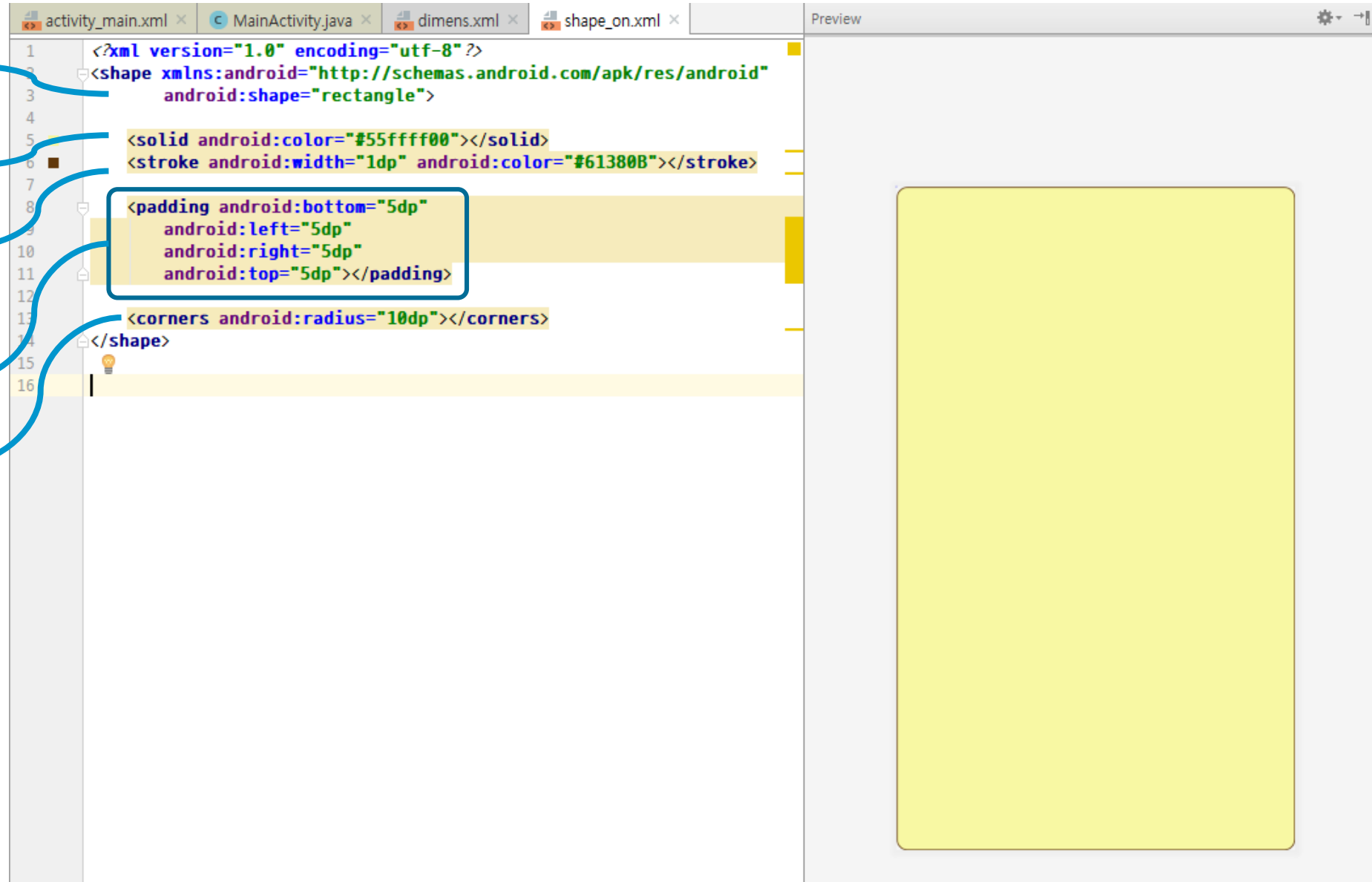
출력모양을 사각형으로 지정

출력모양을 내부의 색

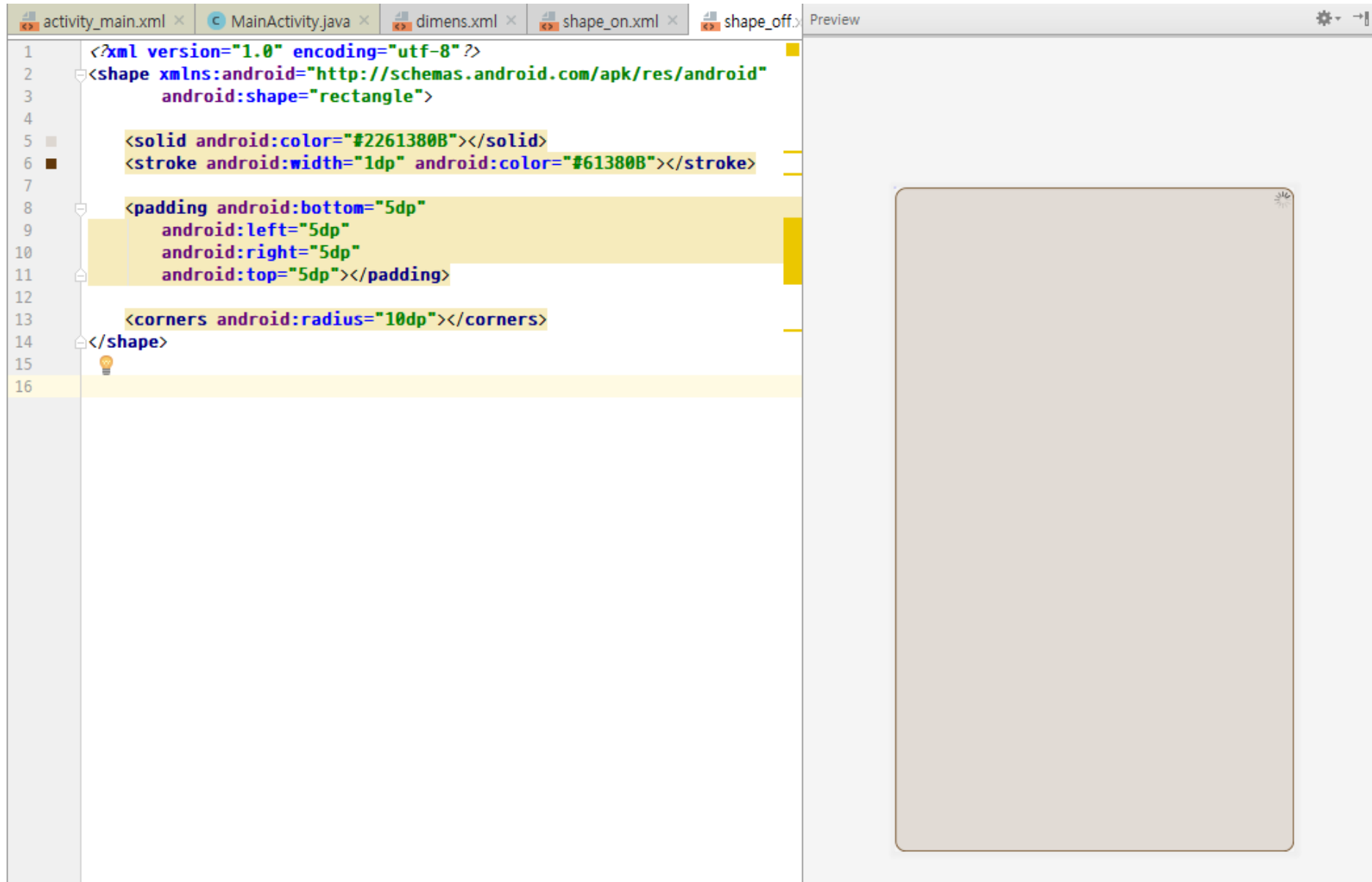
출력모양을 테두리의 색

내부 패딩 정보

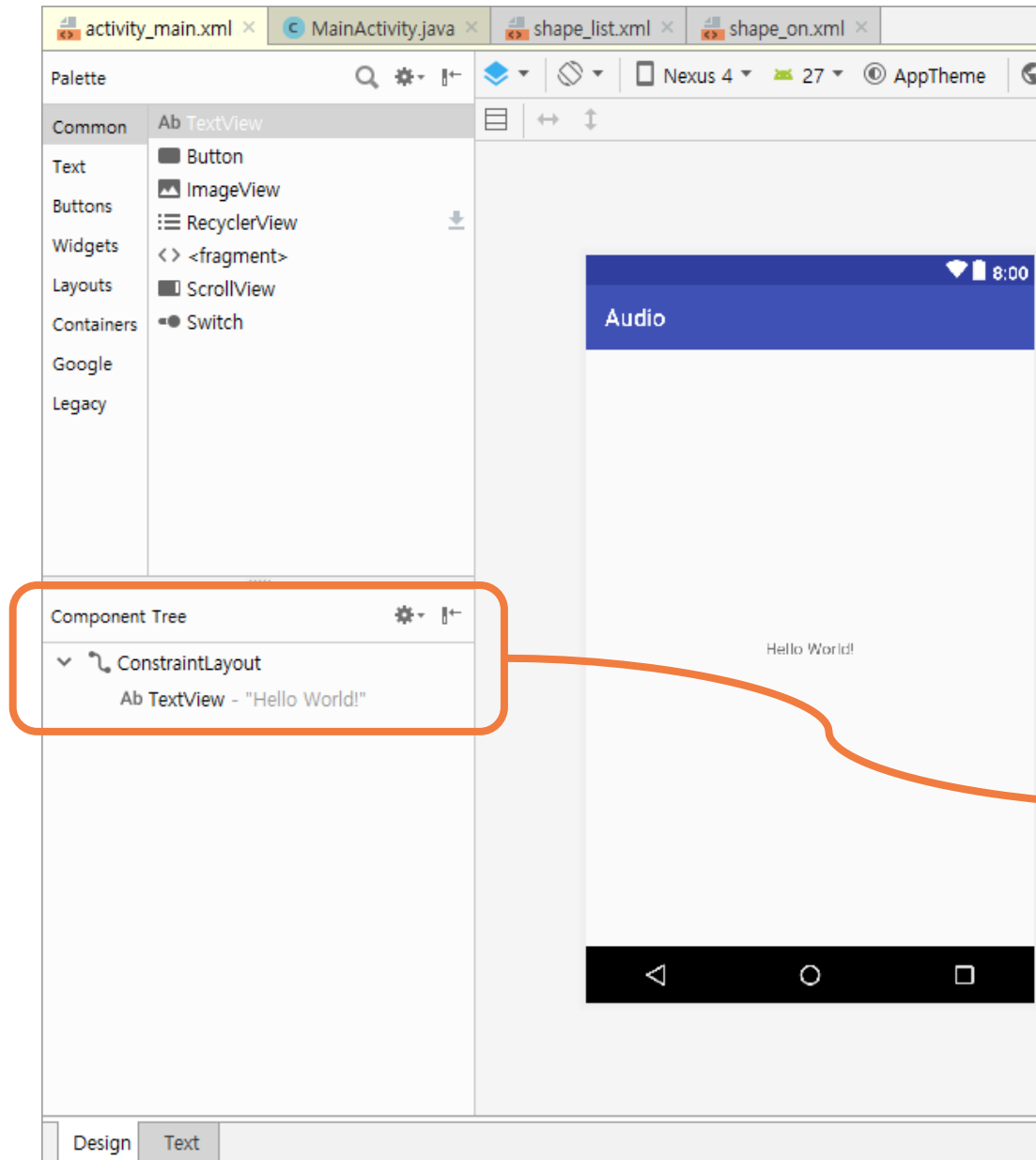
출력모양 모서리를 둥근 모양
으로 지정(반지름은 5dp)



- shape_off.xml 소스(동영상 중지 시)



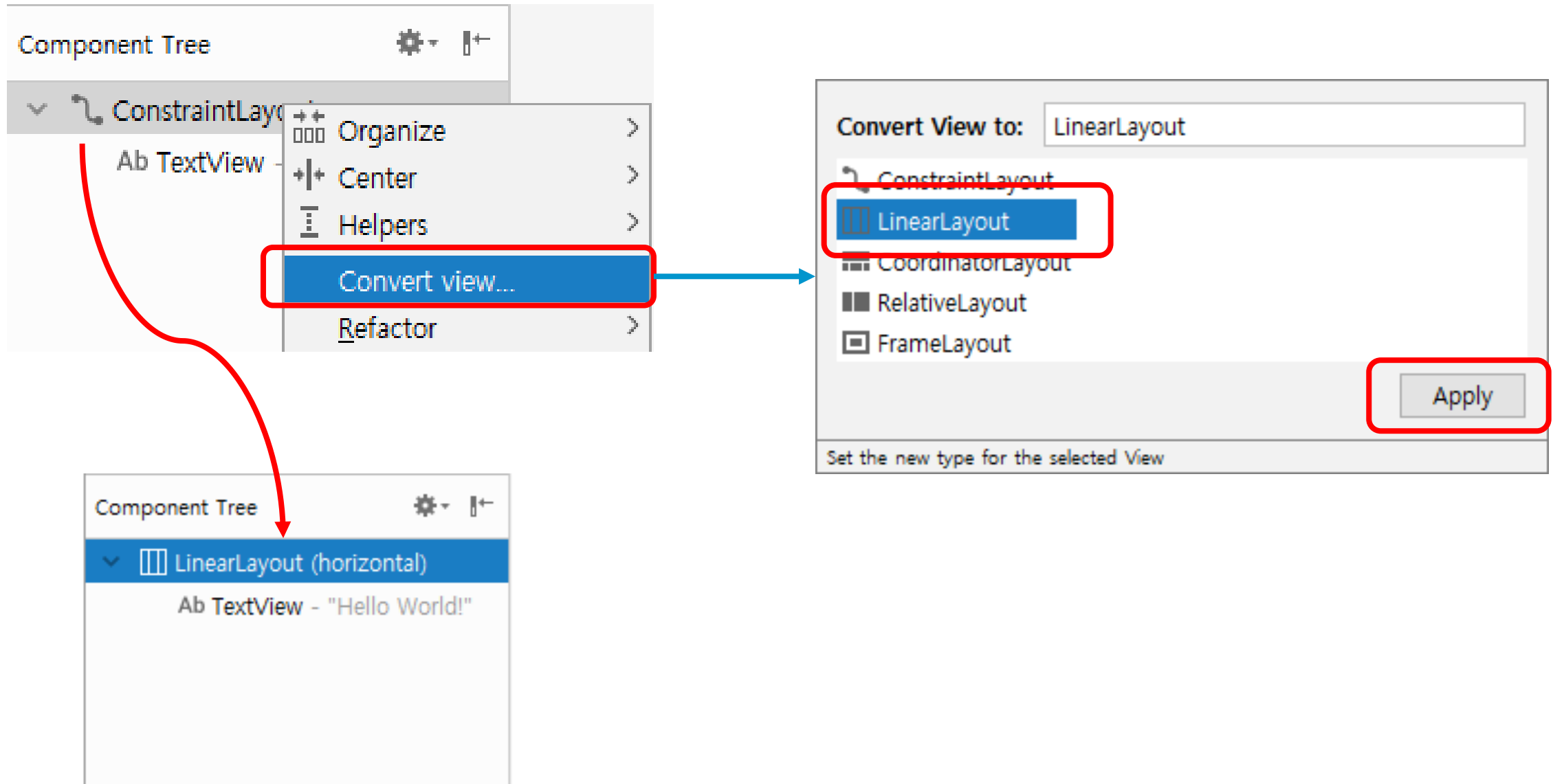
2.4 화면 설계



ConstraintLayout →
LinearLayout으로 변경

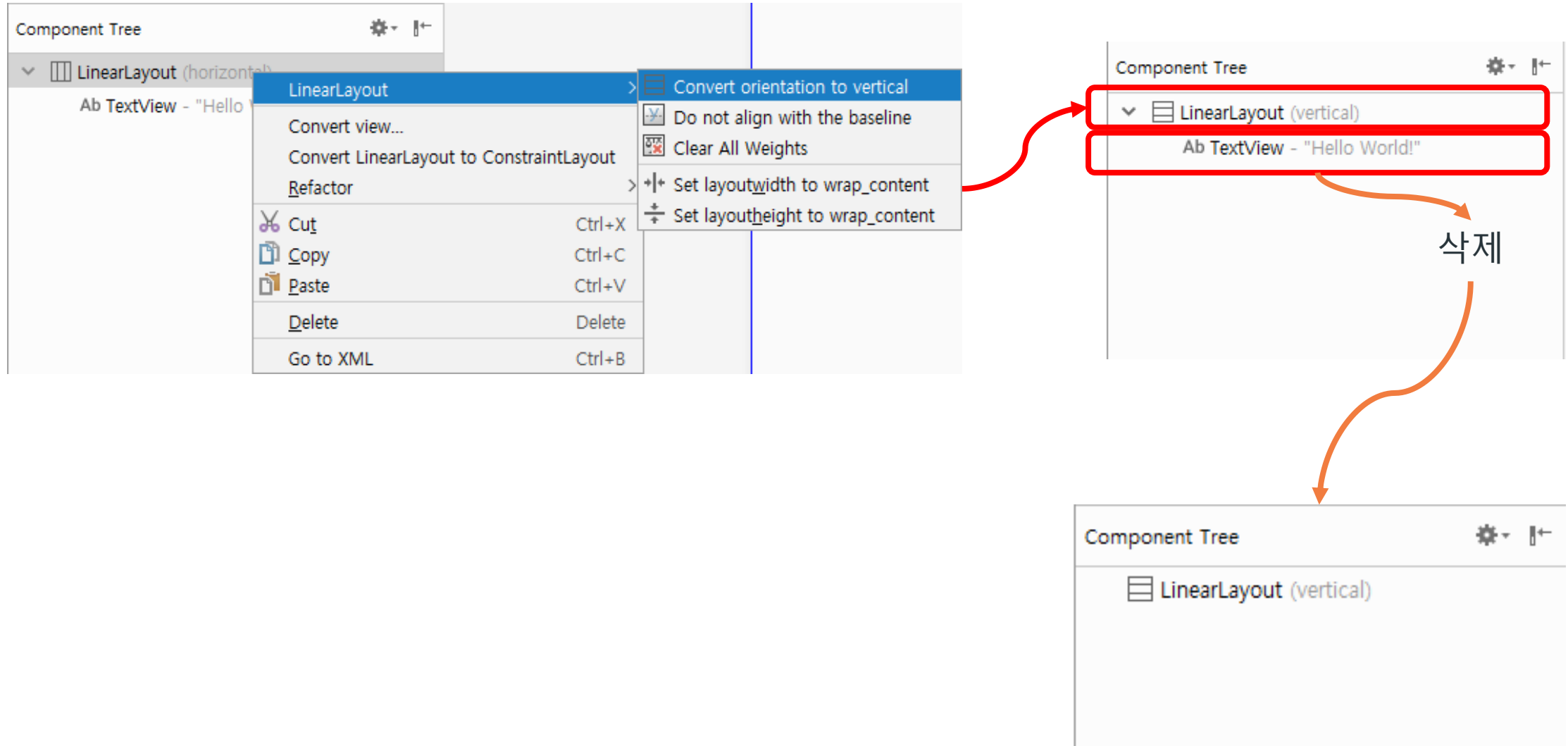
ConstraintLayout을 LinearLayout로 바꾸기

33

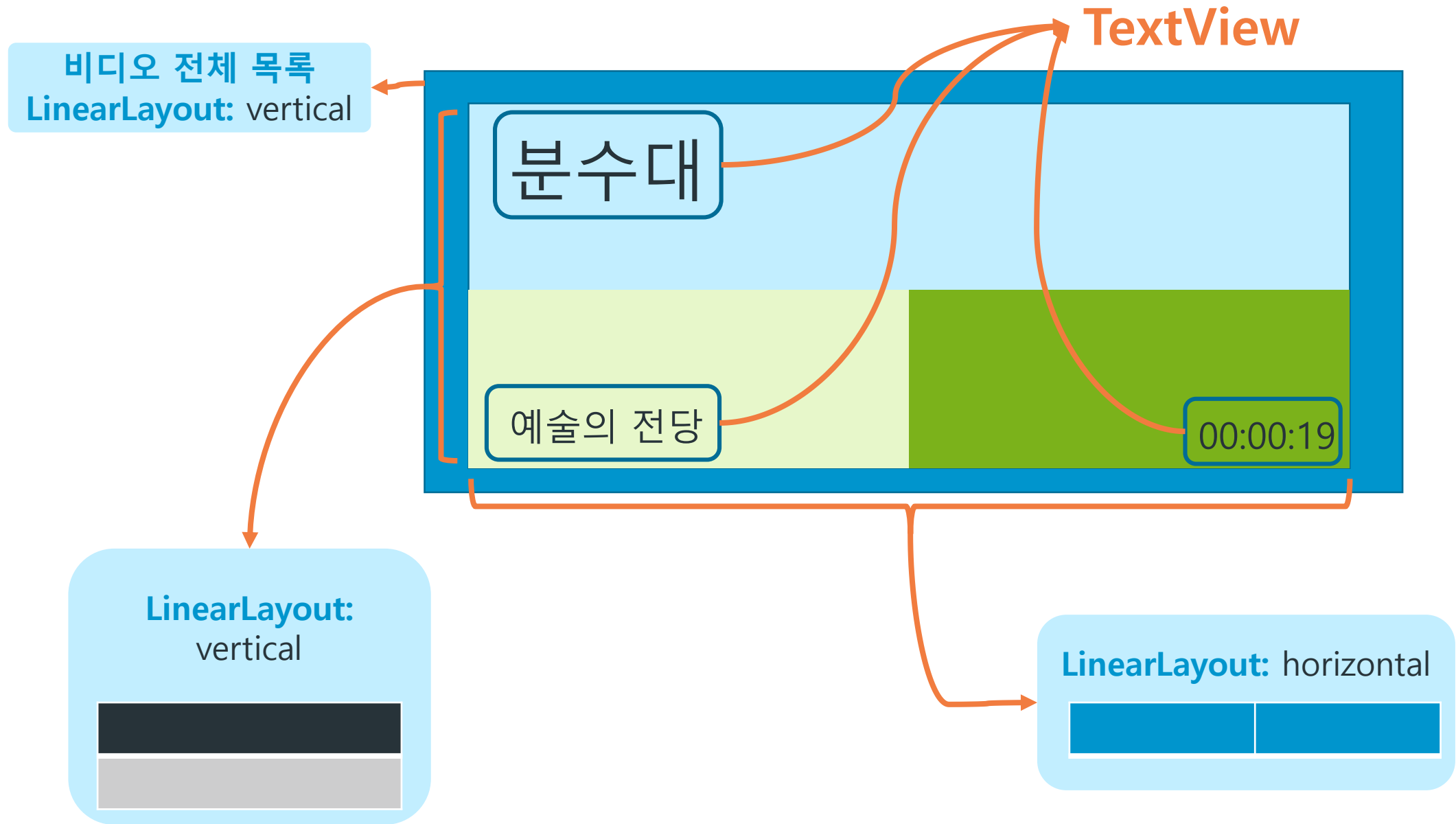


LinearLayout의 방향을 Horizontal → Vertical로 변경하기

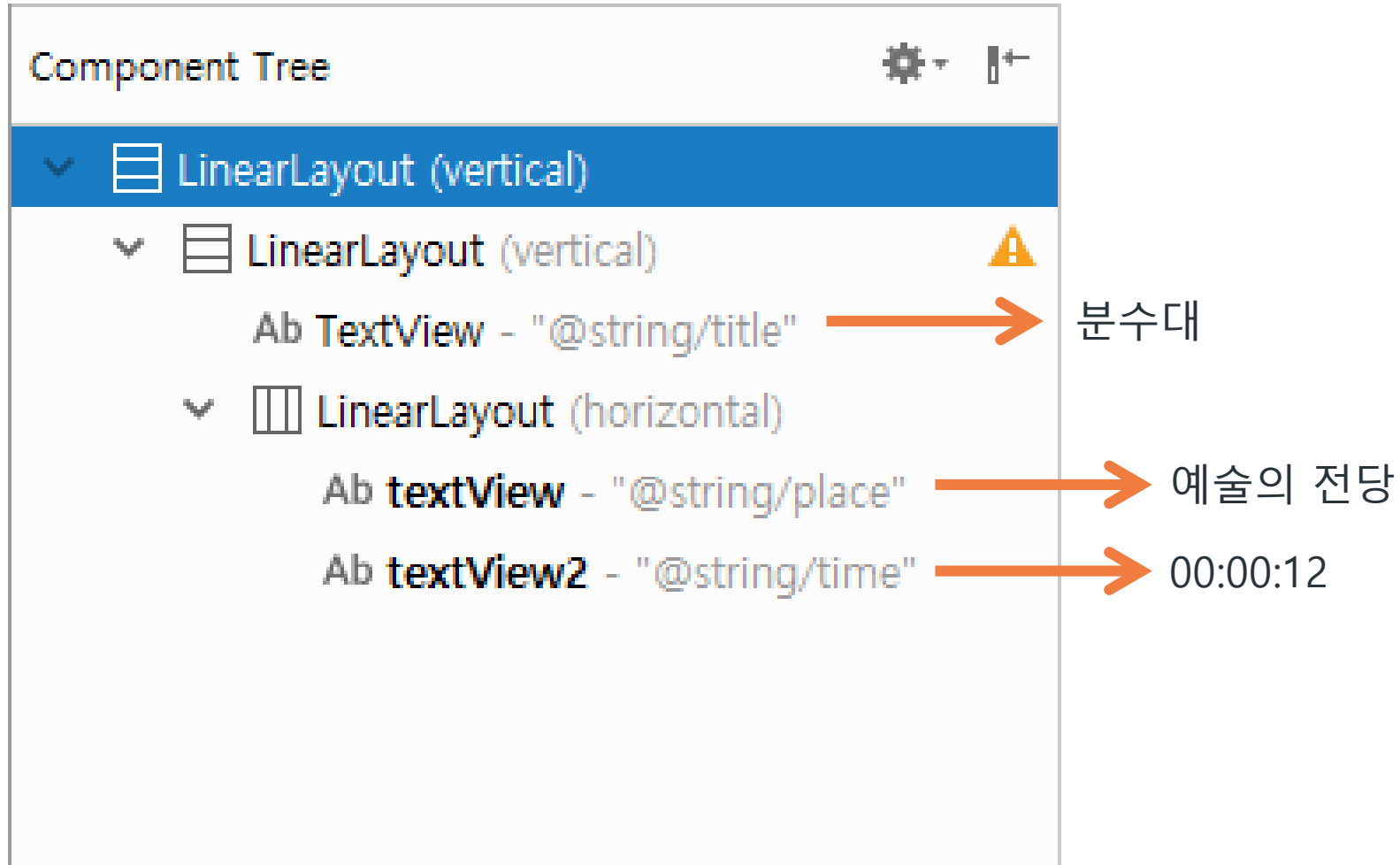
34



- 비디오 목록 표시를 위한 Layout 구조



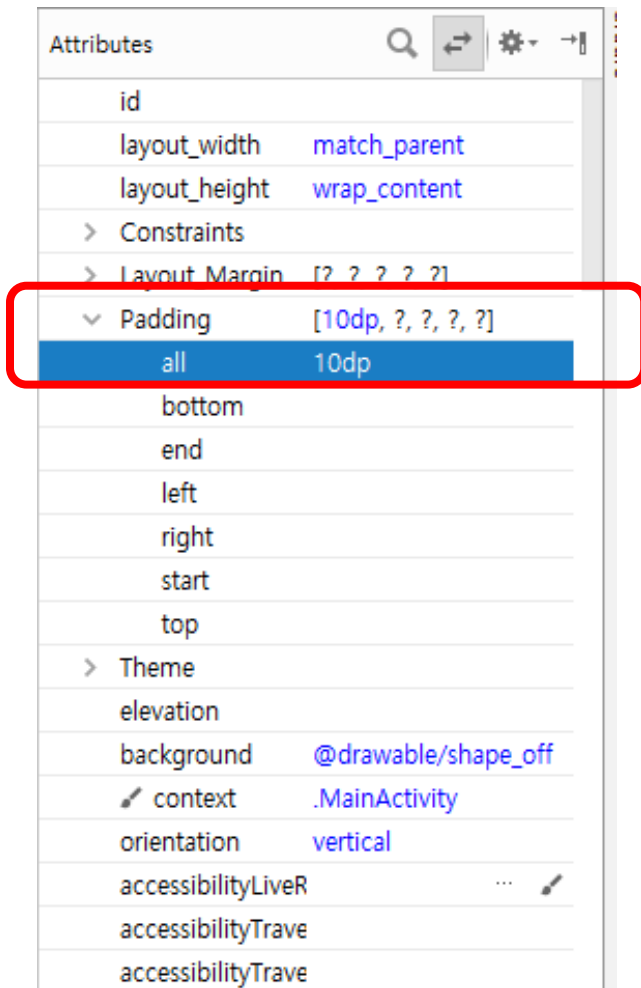
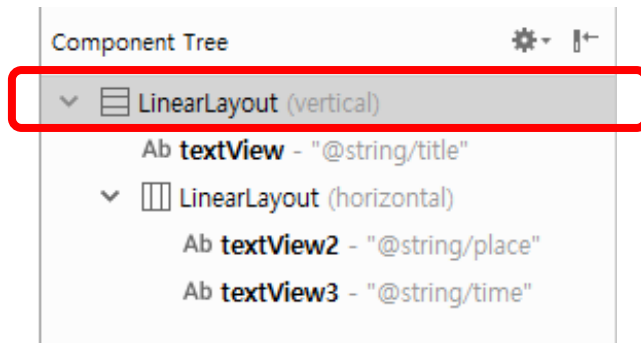
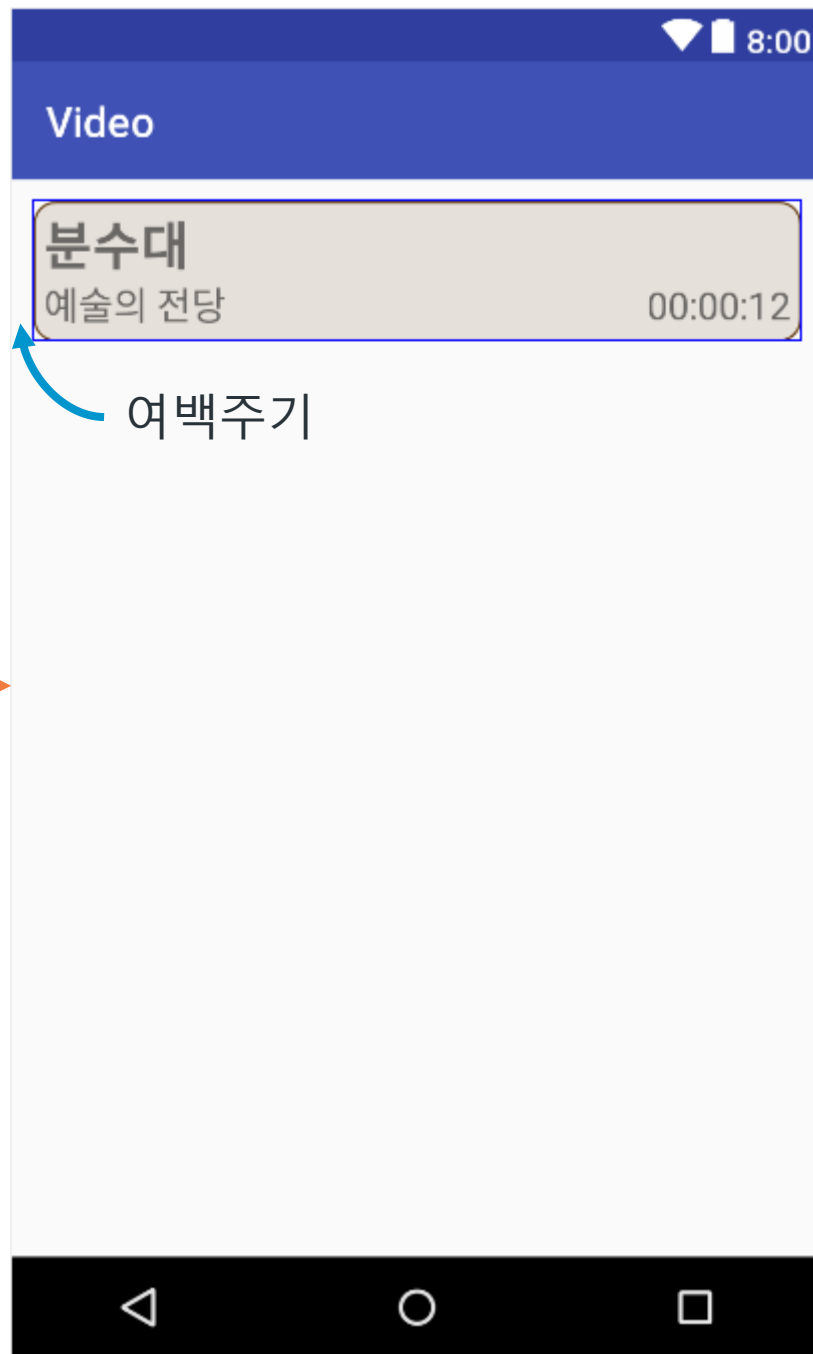
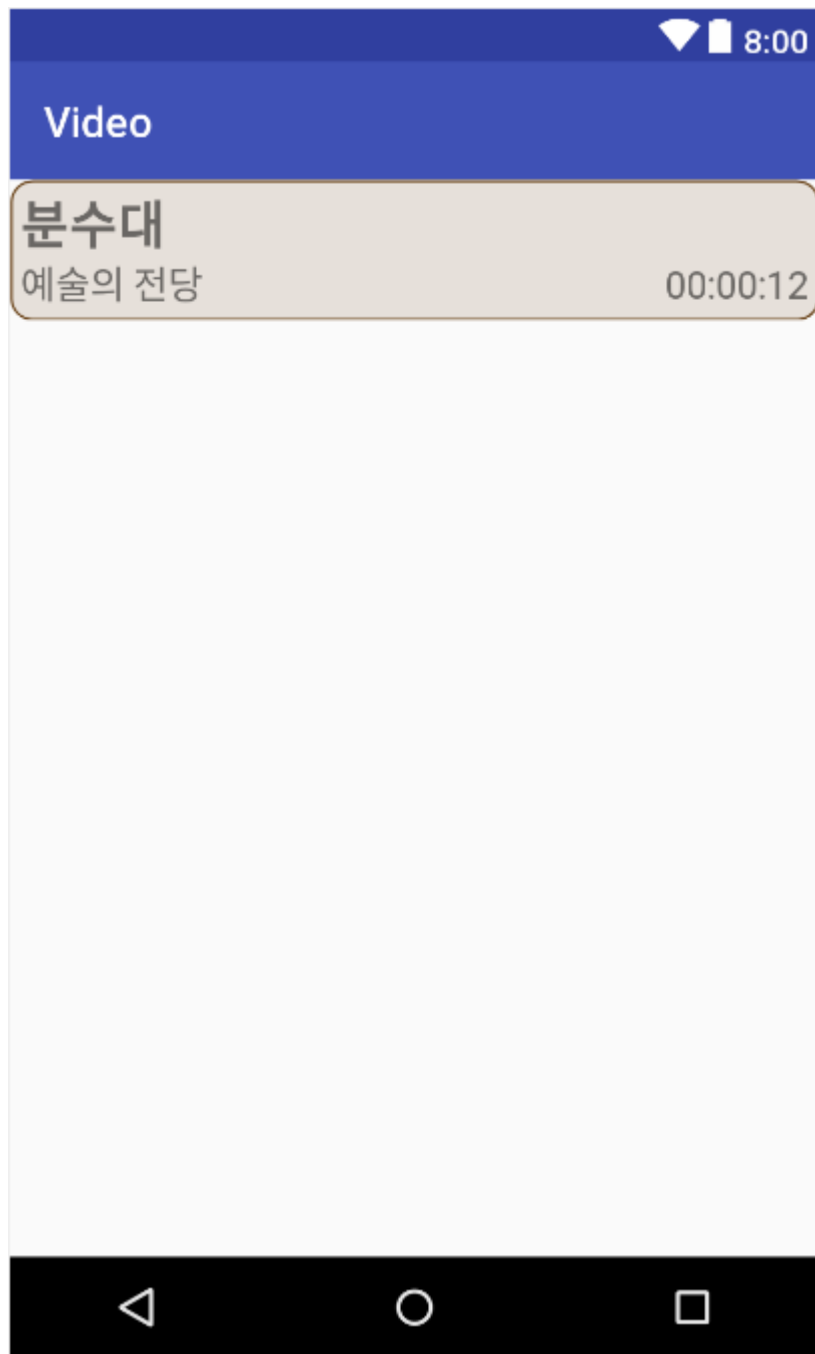
- 동영상 내용 표시를 위한 Layout 구조-Component Tree



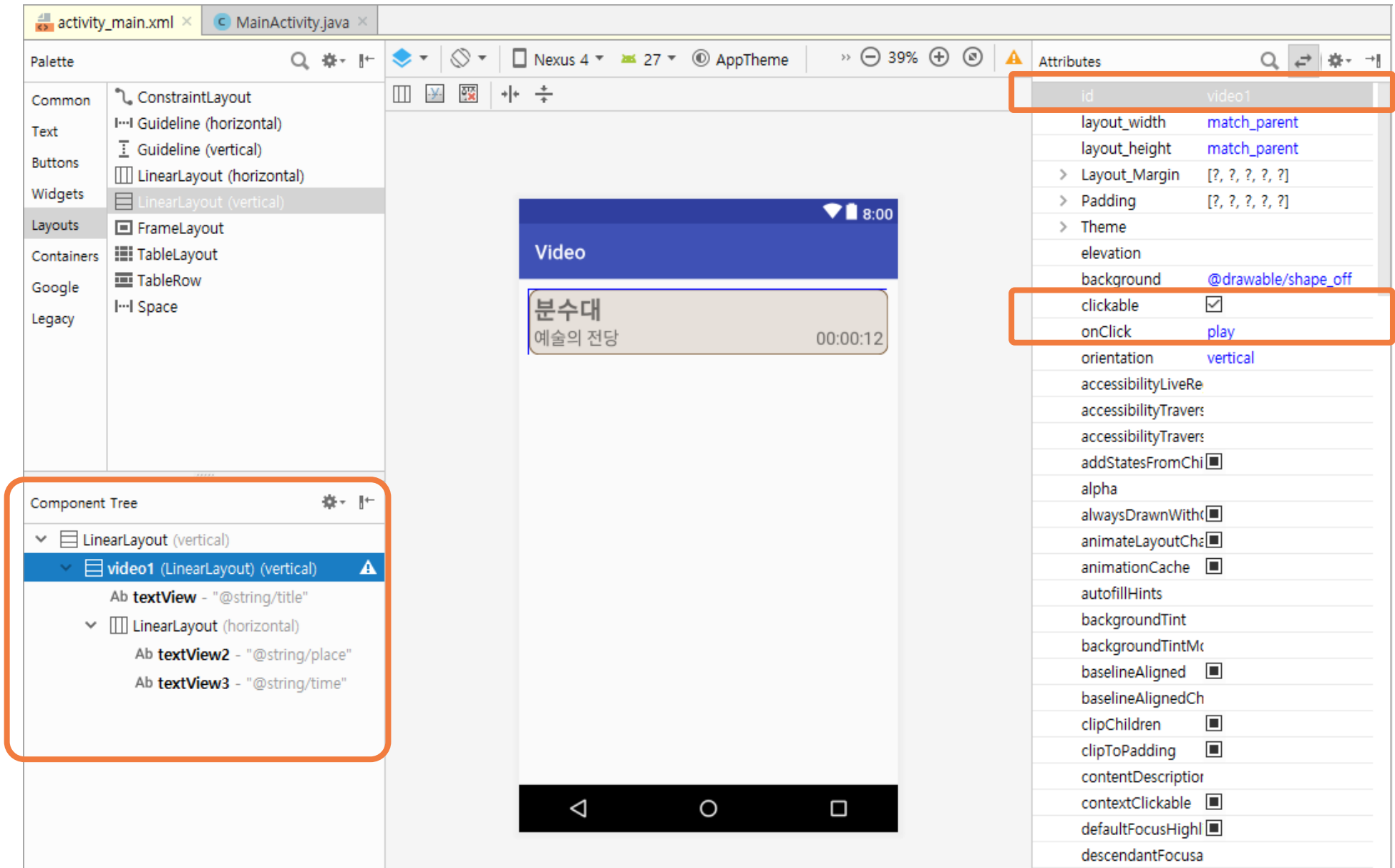
• 동영상 제목 표시 Layout

The screenshot displays the Android Studio IDE with the following components:

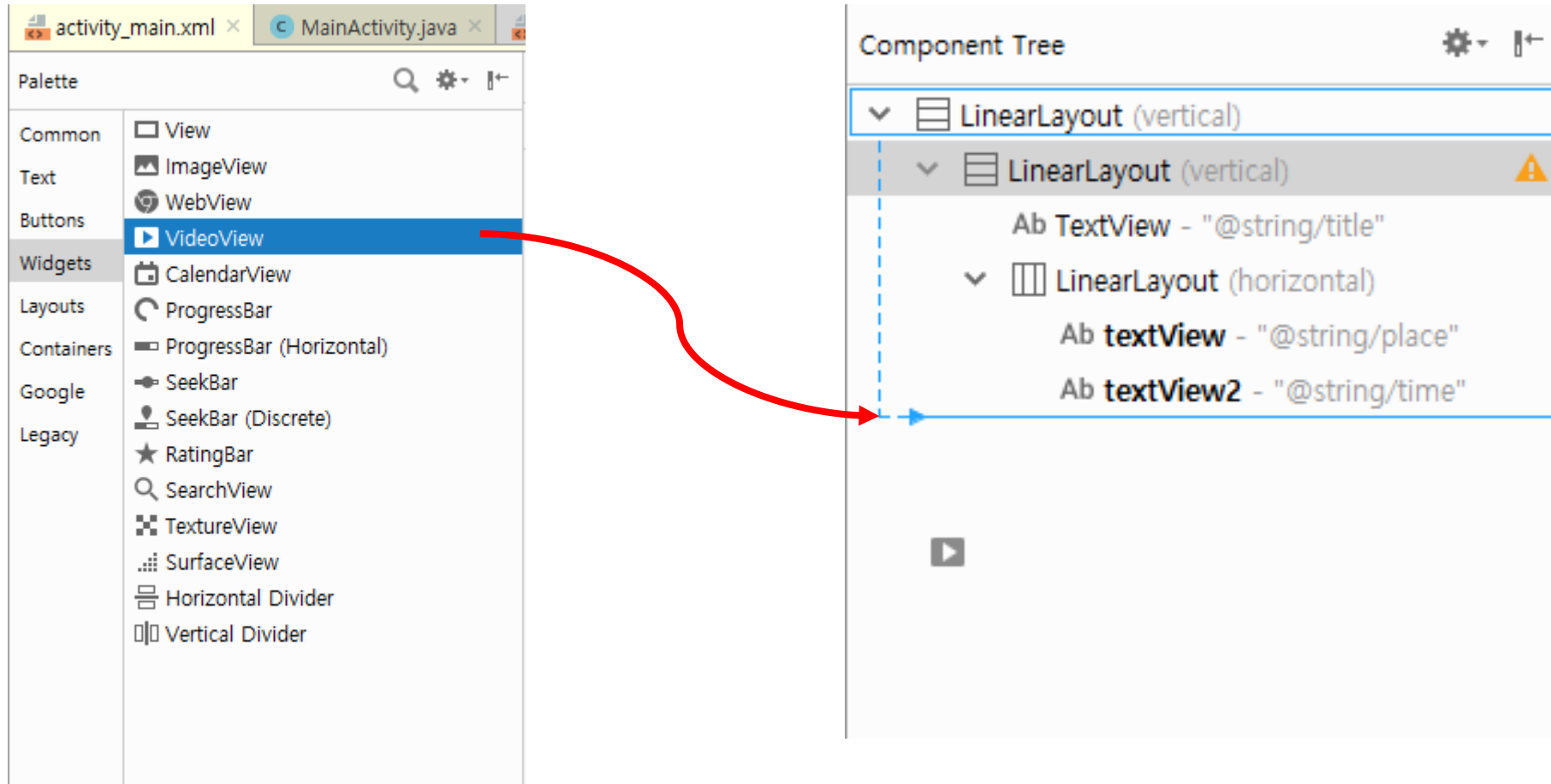
- Palette:** A list of UI widgets on the left side, including Common, Text, Buttons, Widgets, Layouts, Containers, Google, and Legacy. The 'VideoView' widget is highlighted under the 'Widgets' category.
- Design View:** The central area showing the visual representation of the video player. It features a blue header with the title 'Video', a light blue video frame containing the text '분수대 예술의 전당' and a duration of '00:00:12', and a dark blue background.
- Attributes Panel:** Located on the right, it shows the properties for the selected 'video1' component. The 'id' attribute is set to 'video1', and the 'background' attribute is set to '@drawable/shape_off'. Other attributes like 'layout_width', 'layout_height', and 'orientation' are also visible.
- Component Tree:** Located at the bottom left, it shows the hierarchy of the layout. The 'video1 (LinearLayout) (vertical)' component is selected, and its children are listed: 'Ab TextView - "@string/title"', 'Ab textView - "@string/place"', and 'Ab textView2 - "@string/time"'.

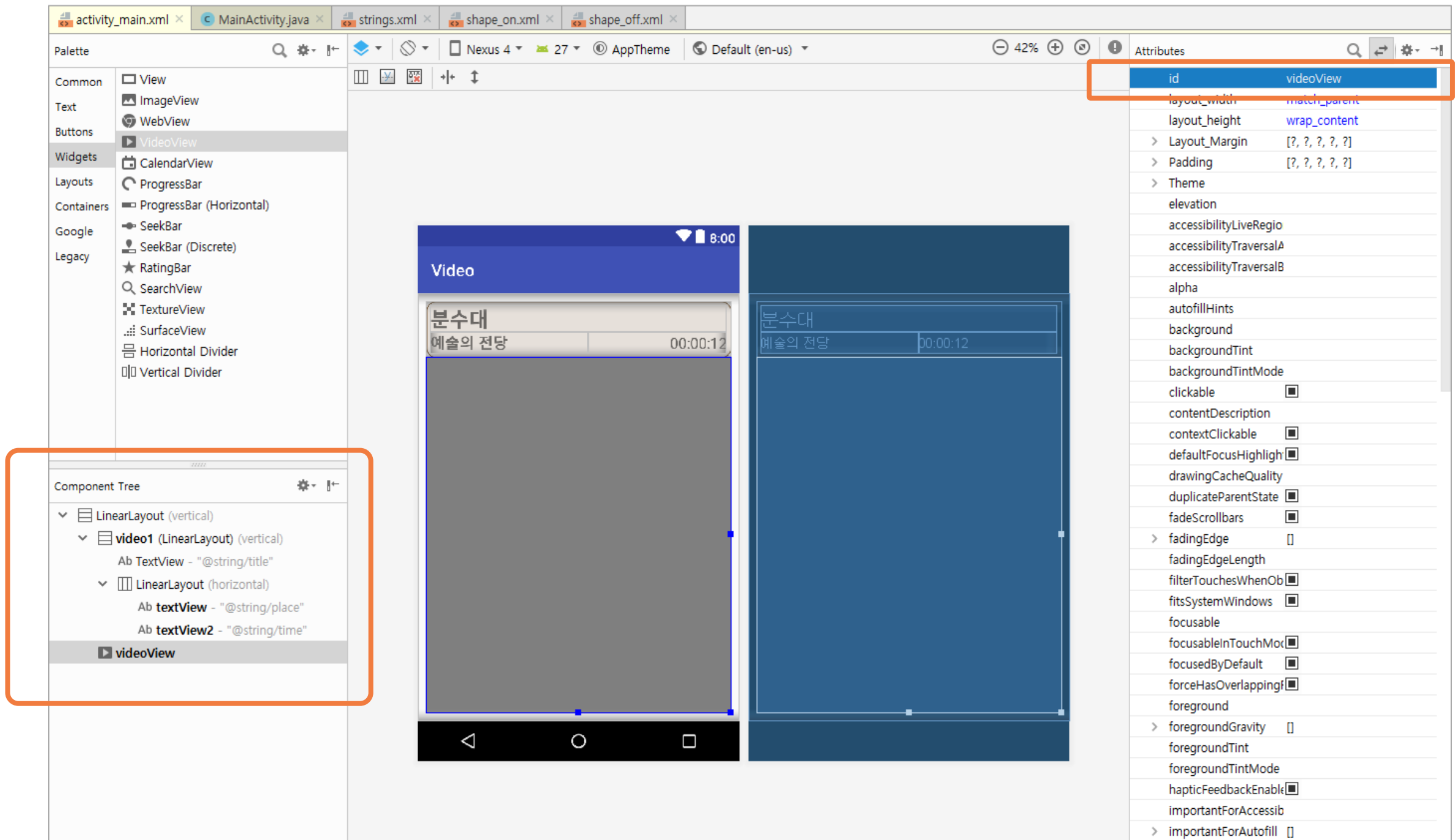


동영상 제목에 onClick속성에 play() 콜백함수 설정



동영상 VideoView 추가





activity_main.xml x MainActivity.java x strings.xml x shape_on.xml x shape_off.xml x

Palette

Common

Text

Buttons

Widgets

Layouts

Containers

Google

Legacy

Component Tree

- LinearLayout (vertical)
 - video1 (LinearLayout) (vertical)
 - Ab TextView - "@string/title"
 - LinearLayout (horizontal)
 - Ab textView - "@string/place"
 - Ab textView2 - "@string/time"
 - videoView

Attributes

id	videoView
layout_width	match_parent
layout_height	wrap_content
Layout_Margin	[?, ?, ?, ?]
Padding	[?, ?, ?, ?]
Theme	
elevation	
accessibilityLiveRegion	
accessibilityTraversalA	
accessibilityTraversalB	
alpha	
autofillHints	
background	
backgroundTint	
backgroundTintMode	
clickable	<input checked="" type="checkbox"/>
contentDescription	
contextClickable	<input checked="" type="checkbox"/>
defaultFocusHighlight	<input checked="" type="checkbox"/>
drawingCacheQuality	
duplicateParentState	<input checked="" type="checkbox"/>
fadeScrollbars	<input checked="" type="checkbox"/>
fadingEdge	<input type="checkbox"/>
fadingEdgeLength	
filterTouchesWhenObscured	<input checked="" type="checkbox"/>
fitsSystemWindows	<input checked="" type="checkbox"/>
focusable	
focusableInTouchMode	<input checked="" type="checkbox"/>
focusedByDefault	<input checked="" type="checkbox"/>
forceHasOverlappingRendering	<input checked="" type="checkbox"/>
foreground	
foregroundGravity	<input type="checkbox"/>
foregroundTint	
foregroundTintMode	
hapticFeedbackEnabled	<input checked="" type="checkbox"/>
importantForAccessibility	
importantForAutofill	<input type="checkbox"/>

2.5 Activity 제어(MainActivity.java)

53

- 비디오 리소스에 대한 VideoView를 생성

The image shows a side-by-side comparison of the MainActivity.java file in an IDE. The left pane shows the initial state where the VideoView class is not imported. The right pane shows the state after adding the import and creating a VideoView object. A lightbulb icon and the Korean word '선택' (Selection) are present next to the VideoView class in the right pane, indicating an IDE suggestion or selection action.

```
1 package com.example.kyungtae.video;
2
3 import android.support.v7.app.AppCompatActivity;
4 import android.os.Bundle;
5
6 public class MainActivity extends AppCompatActivity {
7
8     VideoV
9     c VideoView (android.widget) 선택
10     @Override
11     protected void onCreate(Bundle savedInstanceState) {
12         super.onCreate(savedInstanceState);
13         setContentView(R.layout.activity_
14     }
15 }
16
```

```
1 package com.example.kyungtae.video;
2
3 import android.support.v7.app.AppCompatActivity;
4 import android.os.Bundle;
5 import android.widget.VideoView;
6
7 public class MainActivity extends AppCompatActivity {
8
9     VideoView
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_main);
15     }
16 }
17
```

• VideoView 객체생성과 VideoView 컴포넌트 연결

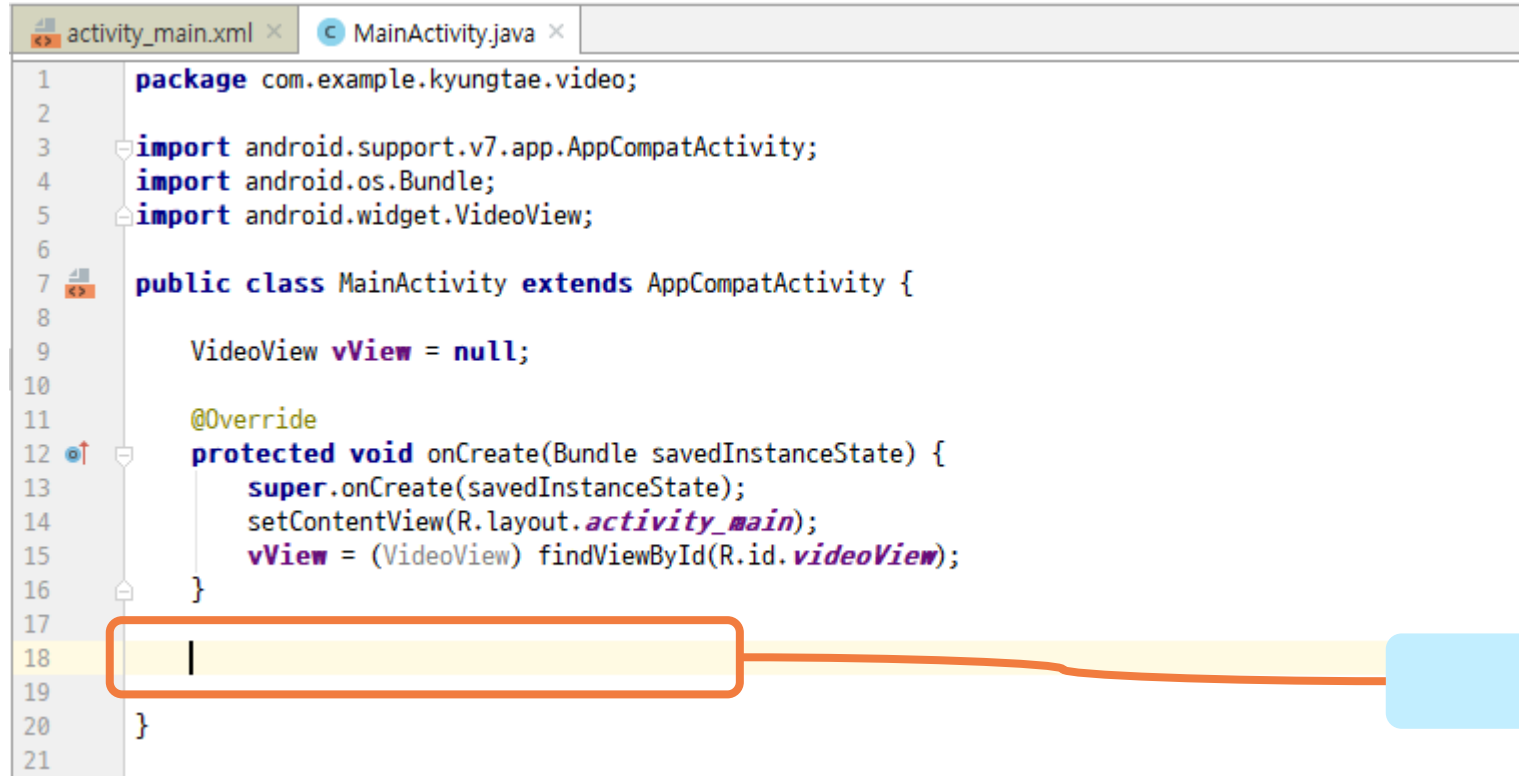
54

```
activity_main.xml x MainActivity.java x
1 package com.example.kyungtae.video;
2
3 import android.support.v7.app.AppCompatActivity;
4 import android.os.Bundle;
5 import android.widget.VideoView;
6
7 public class MainActivity extends AppCompatActivity {
8
9     VideoView vView = null;
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_main);
15         vView = (VideoView) findViewById(R.id.videoView);
16     }
17 }
18
```

비디오뷰 객체 생성(null)

동영상 제목을 클릭했을 때 Video 재생을 위한 함수(play()) 생성

- MainActivity.java 파일

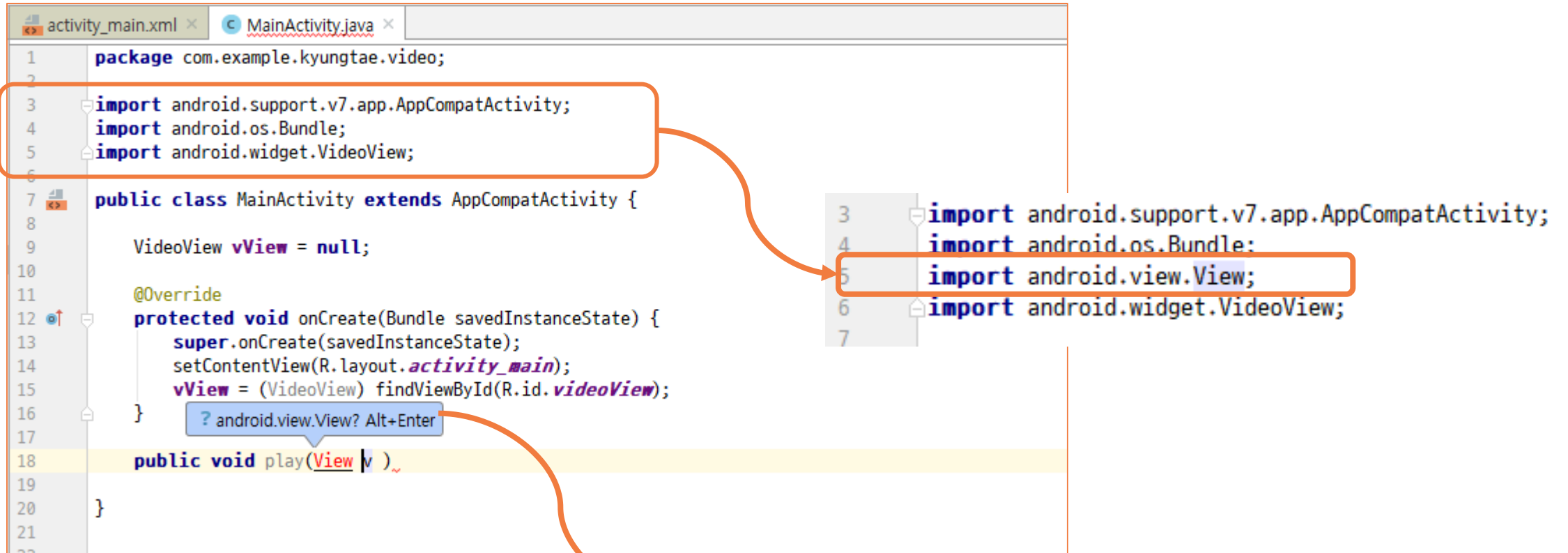


```
1 package com.example.kyungtae.video;
2
3 import android.support.v7.app.AppCompatActivity;
4 import android.os.Bundle;
5 import android.widget.VideoView;
6
7 public class MainActivity extends AppCompatActivity {
8
9     VideoView vView = null;
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_main);
15         vView = (VideoView) findViewById(R.id.videoView);
16     }
17
18
19
20 }
21
```

The image shows an IDE window with two tabs: 'activity_main.xml' and 'MainActivity.java'. The 'MainActivity.java' tab is active, displaying the code above. A yellow rectangular box highlights line 18, which contains a vertical cursor. An orange line extends from the right side of this box to a light blue callout box on the right.

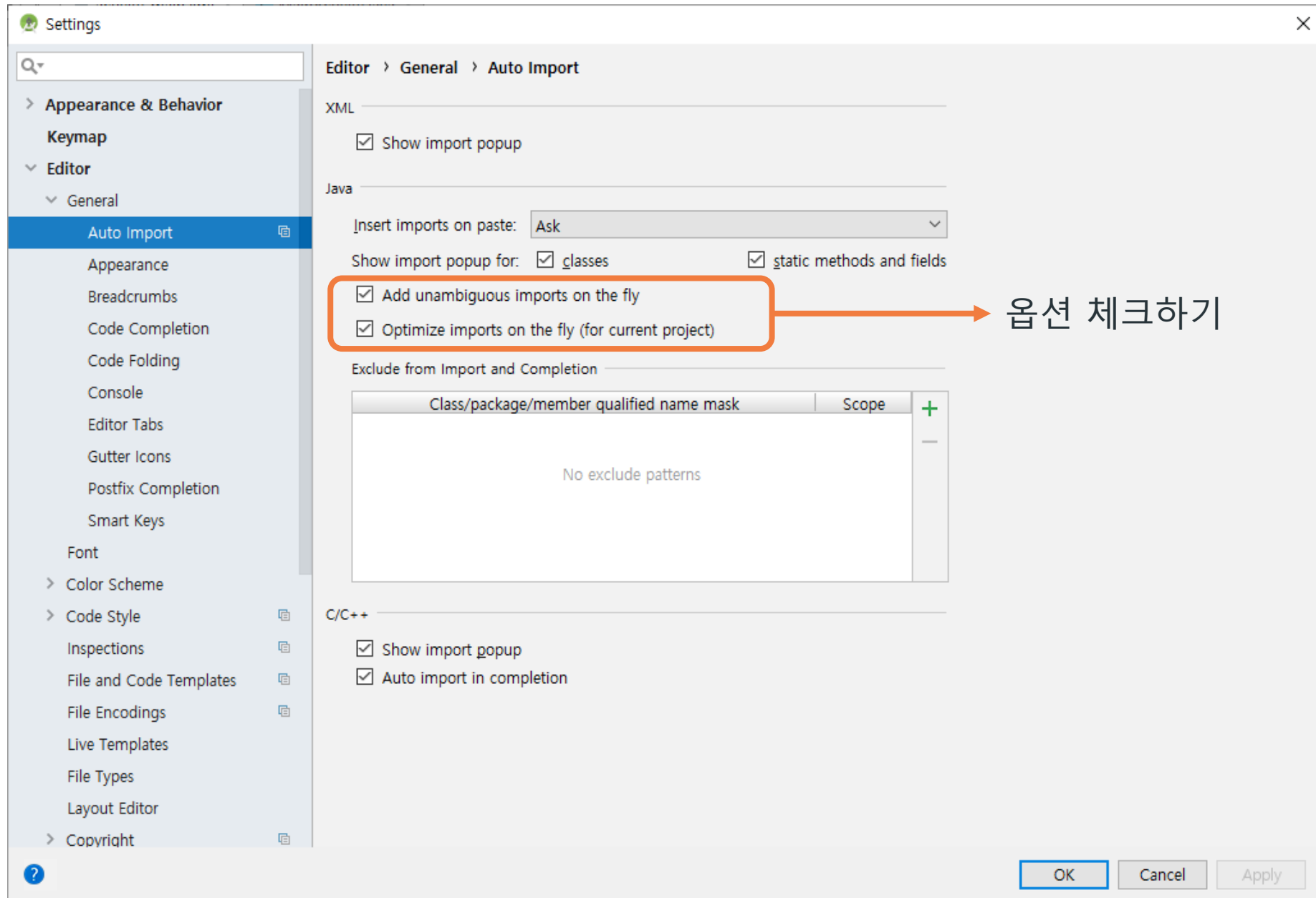
커서 위치 확인

- 코드 입력 중 블루 팝업



코드 구현에 필요한 클래스 임포트를 위해
"Alt+Enter" 키를 누른다.

- 클래스 자동 import 처리 옵션 설정



• 비디오 제목을 클릭했을 때 호출되는 메소드(play()) 추가

59

```
30 public void play(View v){
31     int id = v.getId();
32     LinearLayout layout = (LinearLayout) findViewById(id);
33
34     Resources res = getResources();
35
36     if (vView.isPlaying()){
37         vView.pause();
38         Drawable drawable = res.getDrawable(R.drawable.shape_off, theme: null);
39         layout.setBackground(drawable);
40     }else{
41         Uri uri = Uri.parse("android.resource://com.example.kyungtae.video/" + R.raw.fountain_night);
42         vView.setVideoURI(uri);
43         vView.start();
44         vView.setVisibility(View.VISIBLE);
45
46         Drawable drawable = res.getDrawable(R.drawable.shape_on, theme: null);
47         layout.setBackground(drawable);
48
49         MediaController mc = new MediaController(context: this);
50         vView.setMediaController(mc);
51     }
52 }
53
```

재생 중 일 때

정지 일 때

미디어 플레이어 중지

새로운 Drawable 객체 인식해서
동영상 제목 Layout 배경을 재 설정

uri위치의 동영상 재생 및
화면 표시

동영상 파일의 uri 인식

새로운 Drawable 객체 인식해서
동영상 제목 Layout 배경을 재 설정

미디어 제어기 생성

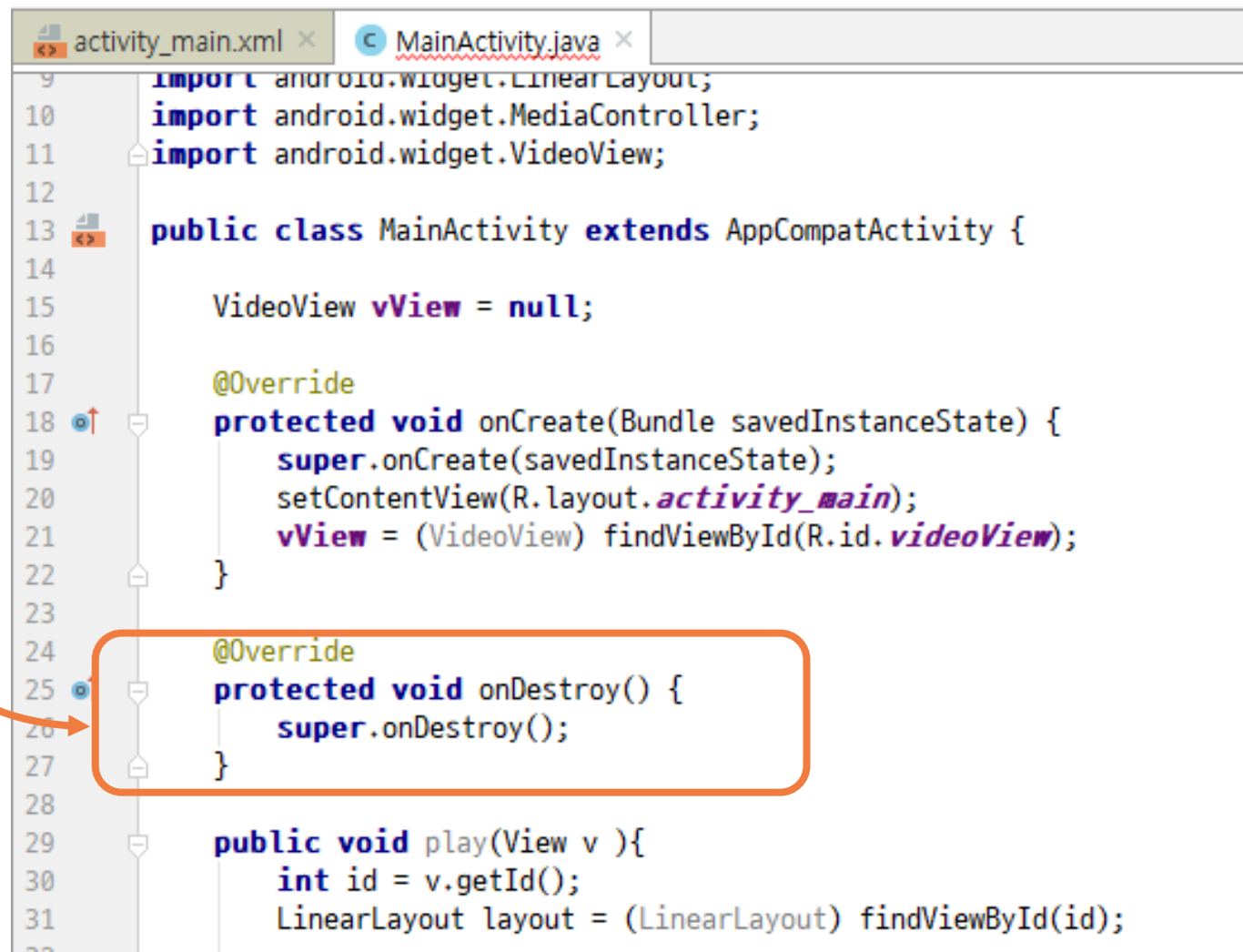
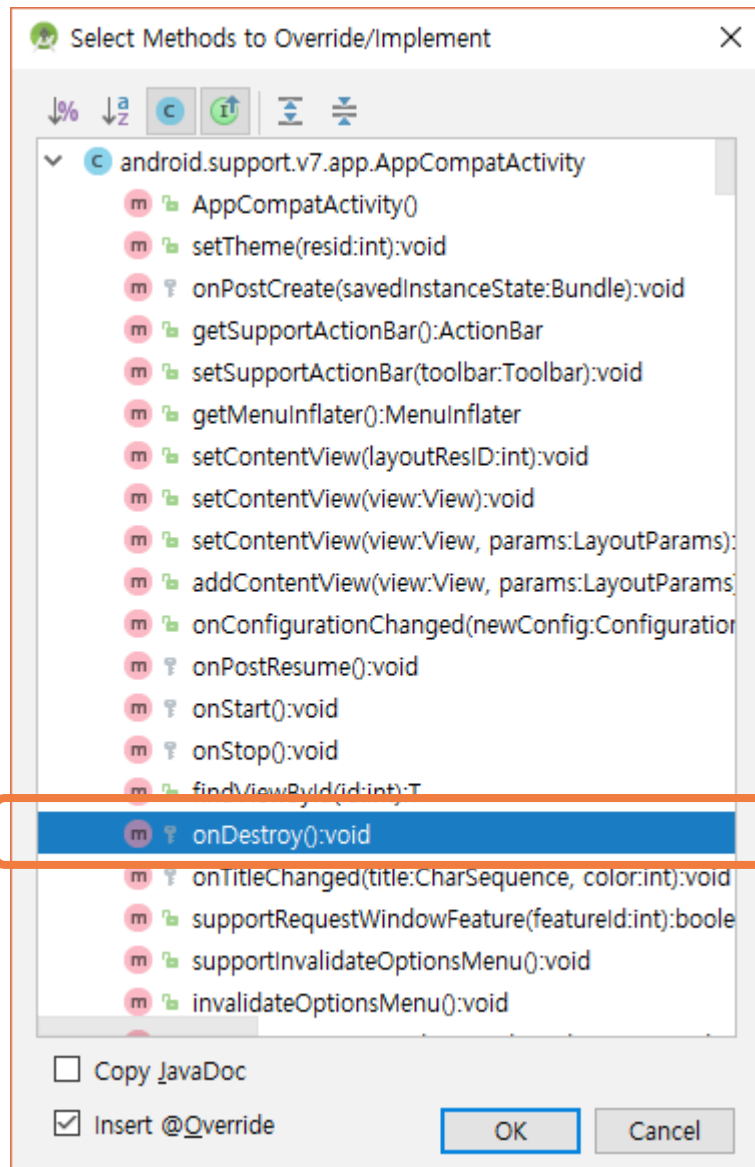
비디오뷰의 미디어 제어기로 설정

• Video 재생을 종료했을 때 - onDestroy() 함수 추가

The screenshot shows an IDE with the 'Override Methods...' menu open. The menu includes options like 'Implement Methods...', 'Delegate Methods...', 'Generate...', 'Surround With...', 'Unwrap/Remove...', 'Completion', 'Folding', 'Insert Live Template...', 'Surround with Live Template...', 'Comment with Line Comment', 'Comment with Block Comment', 'Reformat Code', 'Show Reformat File Dialog', 'Auto-Indent Lines', 'Optimize Imports', 'Rearrange Code', 'Move Statement Down', 'Move Statement Up', 'Move Element Left', 'Move Element Right', 'Move Line Down', 'Move Line Up', 'Update Copyright...', and 'Convert Java File to Kotlin File'. The main editor displays the MainActivity.java file. The code includes an import statement for VideoView, a class declaration for MainActivity extending AppCompatActivity, and an onCreate method. A new @Override method, protected void play(View v), is being added at line 24. The cursor is positioned at the end of the line. A blue callout bubble points to the cursor with the text '커서를 play()함수 위에 위치 시킴'.

```
11 import android.widget.VideoView;
12
13 public class MainActivity extends AppCompatActivity {
14
15     VideoView vView = null;
16
17     @Override
18     protected void onCreate(Bundle savedInstanceState) {
19         super.onCreate(savedInstanceState);
20         setContentView(R.layout.activity_main);
21         vView = (VideoView) findViewById(R.id.videoView);
22     }
23
24     @Override
25     protected void play(View v) {
26         int id = v.getId();
27         LinearLayout layout = (LinearLayout) findViewById(id);
28
29         Resources res = getResources();
30
31         if (vView.isPlaying()) {
32             vView.pause();
33             Drawable drawable = res.getDrawable(R.drawable.shape_off, theme);
34             layout.setBackground(drawable);
35         } else {
36             Uri uri = Uri.parse("android.resource://com.example.kyungtae.vide
37             vView.setVideoURI(uri);
38             vView.start();
39             vView.setVisibility(View.VISIBLE);
40         }
41     }
42 }
```

- onDestroy()는 수퍼 클래스에 정의 되어 있으므로 Override 함



- 프로젝트 리소스를 얻기 위한 함수 추가

23
24
25
26
27
28
29

```
@Override  
protected void onDestroy() {  
    vView.pause();  
    super.onDestroy();  
}
```

액티비티 종료

비디오 뷰 중지

- raw 폴더의 동영상 파일인식이 안 될 경우
 - Uri 리소스 획득 방법 변경

```
int id_video = res.getIdentifier( name: "fountain_nigh", defType: "raw", getPackageName());  
Uri uri = Uri.parse("android.resource://com.example.kyungtae.video" + id_video);
```

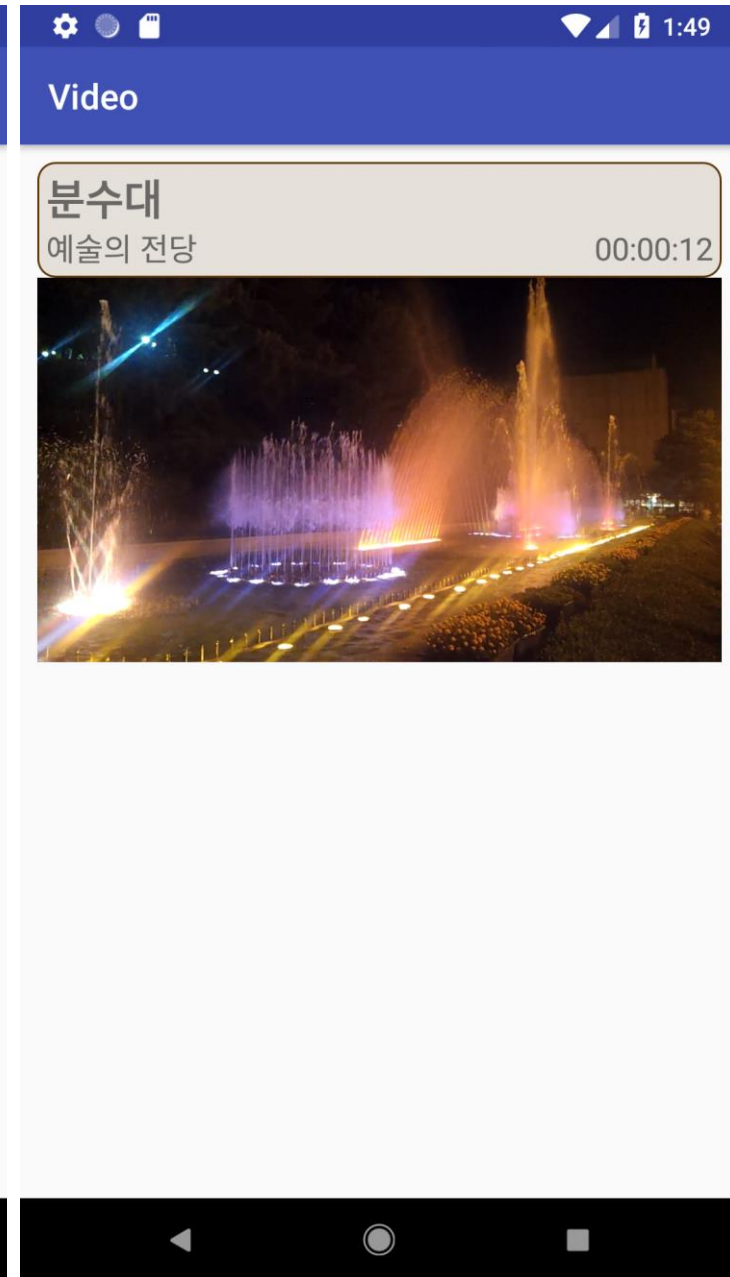
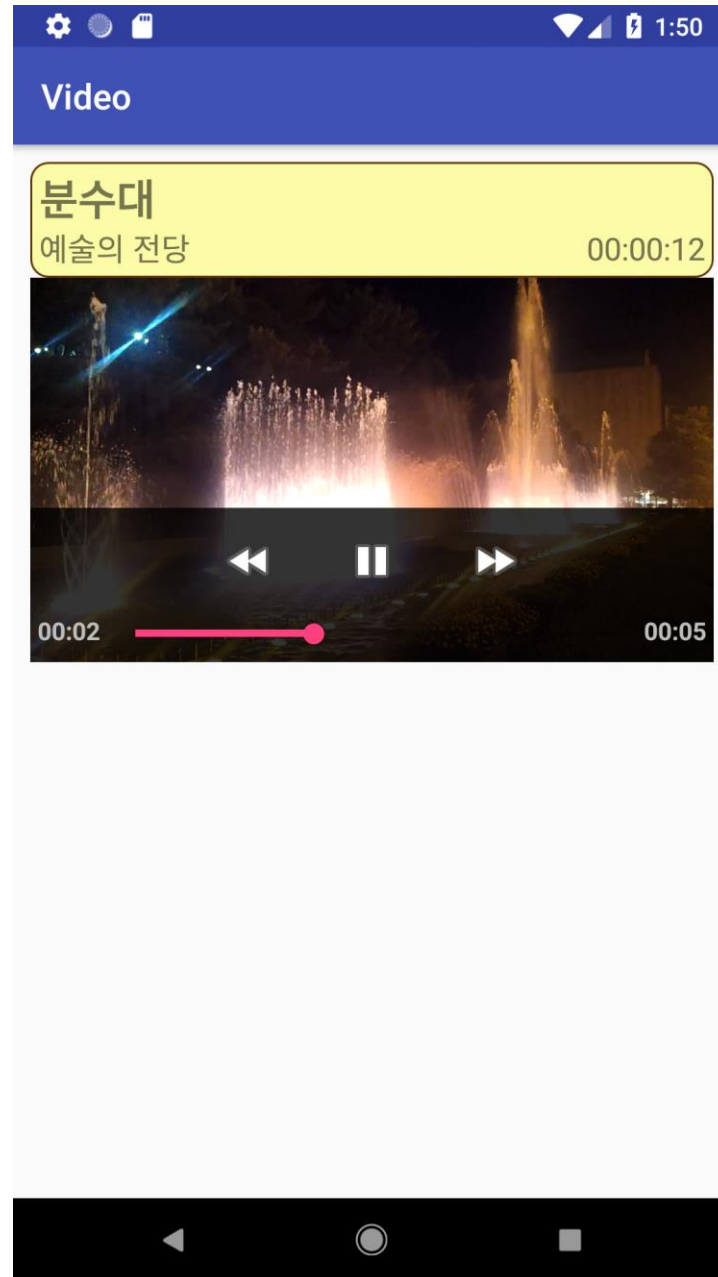
클래스와 속성/메소드

- 클래스

클래스	설명
MediaController	미디어 실행 제어를 포함하는 뷰

- 메소드

클래스	메소드	설명
VideoView	Boolean <code>isPlaying()</code>	비디오뷰의 실행 여부
	void <code>pause()</code>	비디오뷰의 중지
	void <code>setMediaController(MediaController controller)</code>	미디어 제어기를 설정함



Q & A uestion nswer

69

