

고급 위젯

Instructor: Park, JoonSeok (pjs50@pusan.ac.kr)

날짜/시간

▶ 아날로그 시계와 디지털 시계

Android APIs

API level:

15

[android.text.format](#)

[android.text.method](#)

[android.text.style](#)

[android.text.util](#)

[android.transition](#)

[android.util](#)

[android.view](#)

[android.view.accessibility](#)

[android.view.animation](#)

[android.view.inputmethod](#)

[android.view.textservice](#)

[android.webkit](#)

[android.widget](#)

public class

DigitalClock

extends [TextView](#)

Summary: [Inherited XML Attrs](#) | [Inherited Constants](#) | [Inherited Fields](#) | [Ctors](#) | [Methods](#) | [Protected Methods](#) | [Inherited Methods](#) | [\[Expand All\]](#)

Added in API level 1

Deprecated since API level 17

[java.lang.Object](#)

↳ [android.view.View](#)

↳ [android.widget.TextView](#)

↳ [android.widget.DigitalClock](#)

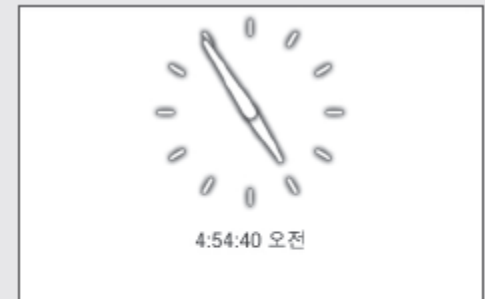
This class was deprecated in API level 17.

It is recommended you use [TextClock](#) instead.

Class Overview

날짜/시간

```
1 <LinearLayout>
2   <AnalogClock
3     android:layout_width="match_parent"
4     android:layout_height="wrap_content" />
5   <DigitalClock
6     android:layout_width="match_parent"
7     android:layout_height="wrap_content"
8     android:gravity="center" />
9 </LinearLayout>
```



날짜/시간

▶ 크로노미터

The screenshot shows the Android Studio interface. On the left, the 'Android APIs' pane is open, displaying a list of API packages. The 'API level' is set to 15. The 'android.widget' package is selected. The main pane displays the documentation for the **Chronometer** class. It shows that **Chronometer** is a public class that extends **TextView**. The class hierarchy is shown as **java.lang.Object** → **android.view.View** → **android.widget.TextView** → **android.widget.Chronometer**. The 'Class Overview' section states: 'Class that implements a simple timer.'

```
1 <LinearLayout>
2   <Chronometer
3       android:id="@+id/chronometer1"
4       android:layout_width="match_parent"
5       android:layout_height="wrap_content"
6       android:format="시간 측정 : %S"
7       android:gravity="center"
8       android:textSize="30dp" />
9 </LinearLayout>
```

시간 측정 : 00:00

날짜/시간

▶ 타임피커, 데이터피커, 캘린더 뷰

```
1 <LinearLayout>
2   <TimePicker
3     android:layout_width="match_parent"
4     android:layout_height="120dp" />
5   <DatePicker
6     android:layout_width="match_parent"
7     android:layout_height="120dp" />
8   <CalendarView
9     android:layout_width="match_parent"
10    android:layout_height="match_parent"
11    android:showWeekNumber="false" />
12 </LinearLayout>
```



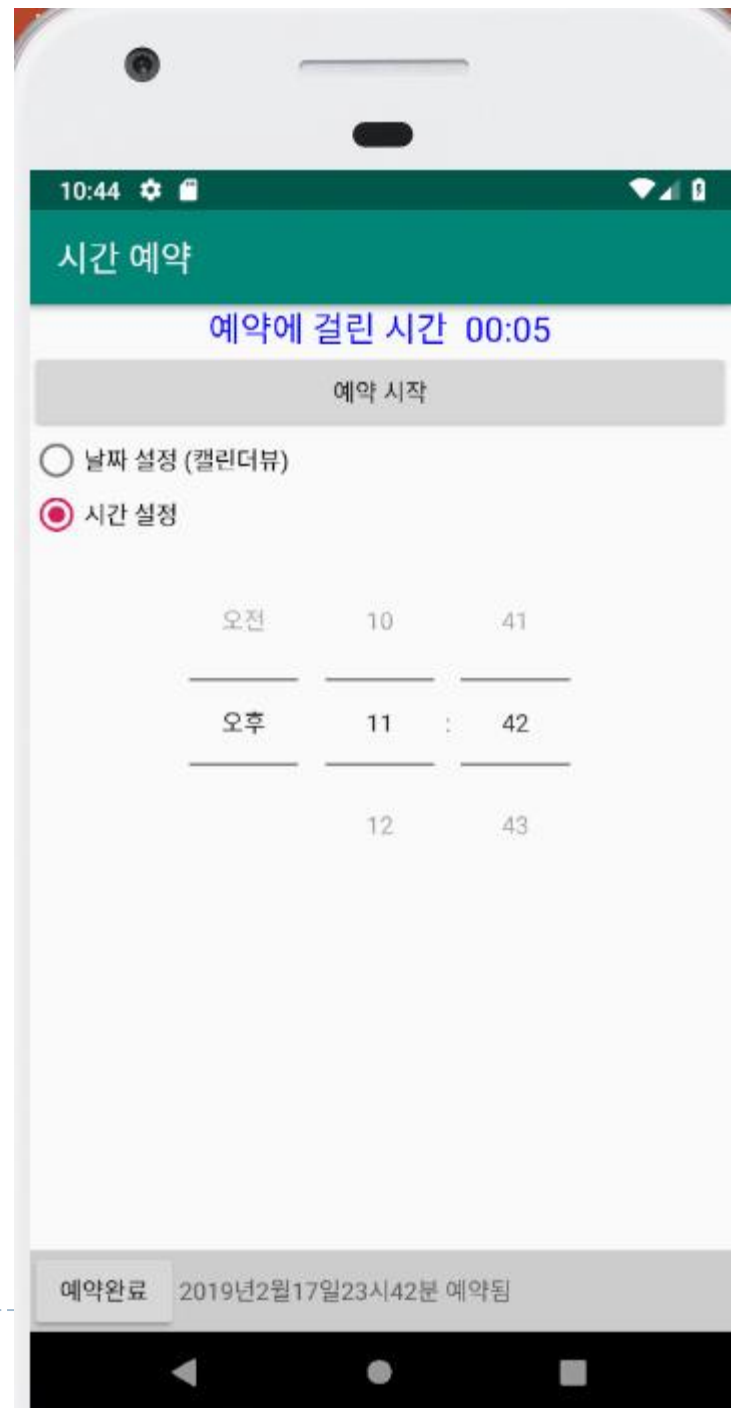
캘린더뷰는
허니콤(3.0)부터
지원됨

날짜/시간 예약 앱 만들기

▶ 기능

- ▶ 타이머 기능: <예약 시작>과 <예약 완료>를 클릭하면 크로노미터가 타이머로 동작
- ▶ 날짜/시간 설정: <날짜 설정>과 <시간 설정>을 클릭하면 예약할 날짜와 시작을 변경
- ▶ 날짜/시간 결정: <예약 완료>를 클릭하면 설정한 날짜와 시간이 결정







Create Android Project

Application name**Company domain****Project location**

...

Package name☐ Include C++ support☐ Include Kotlin support

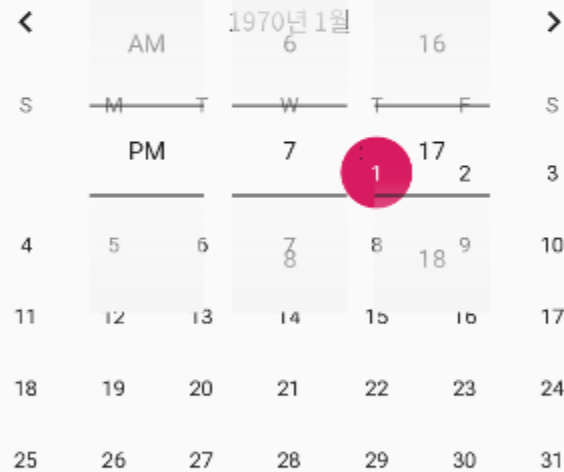
Reservation_Test

예약에 걸린 시간 (1):(23)

예약 시작

☐ 날짜 설정 (캘린더뷰)

☐ 시간 설정



예약완료 0000년00월00일00시00분 예약됨

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      android:layout_width="fill_parent"
5      android:layout_height="fill_parent"
6      android:baselineAligned="false"
7      android:orientation="vertical" >
8
9      <LinearLayout
10         android:layout_width="match_parent"
11         android:layout_height="wrap_content"
12         android:orientation="vertical" >
13
14         <Chronometer
15             android:id="@+id/chronometer1"
16             android:layout_width="match_parent"
17             android:layout_height="wrap_content"
18             android:format="예약에 걸린 시간 %s"
19             android:gravity="center"
20             android:textSize="20dp" />
21
22         <Button
23             android:id="@+id/btnStart"
24             android:layout_width="match_parent"
25             android:layout_height="wrap_content"
26             android:text="예약 시작" />
27     </LinearLayout>

```

28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44

```
<RadioGroup
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" >

    <RadioButton
        android:id="@+id/rdoCal"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="날짜 설정 (캘린더뷰)" />

    <RadioButton
        android:id="@+id/rdoTime"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="시간 설정" />

</RadioGroup>
```



```
45
46 <LinearLayout
47     android:layout_width="match_parent"
48     android:layout_height="0dip"
49     android:layout_weight="1" >
50
51     <FrameLayout
52         android:layout_width="match_parent"
53         android:layout_height="match_parent"
54         android:gravity="center" >
55
56         <CalendarView
57             android:id="@+id/calendarView1"
58             android:layout_width="match_parent"
59             android:layout_height="match_parent"
60             android:showWeekNumber="false" />
61
62         <TimePicker
63             android:timePickerMode="spinner"
64             android:id="@+id/timePicker1"
65             android:layout_width="match_parent"
66             android:layout_height="match_parent" />
67     </FrameLayout>
68 </LinearLayout>
```

<LinearLayout

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#CCCCCC" >
```

<Button

```
    android:id="@+id/btnEnd"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="예약완료" />
```

<TextView

```
    android:id="@+id/tvYear"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="0000" />
```

<TextView

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="년" />
```

<TextView

```
    android:id="@+id/tvMonth"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="00" />
```

98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="월" />

<TextView
    android:id="@+id/tvDay"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="00" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="일" />

<TextView
    android:id="@+id/tvHour"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="00" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="시" />
```

124

125

126

127

128

129

130

131

132

133

134

135

136

137

```
<TextView
```

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

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:text="00" />
```

```
<TextView
```

```
    android:layout_width="wrap_content"
```


```
    android:layout_height="wrap_content"
```

```
    android:text="분 예약됨" />
```

```
</LinearLayout>
```

```
</LinearLayout>
```



```
11
12  public class MainActivity extends AppCompatActivity {
13
14     Chronometer chrono;
15     Button btnStart, btnEnd;
16     RadioButton rdoCal, rdoTime;
17     CalendarView calView;
18     TimePicker tPicker;
19     TextView tvYear, tvMonth, tvDay, tvHour, tvMinute;
20     int selectYear, selectMonth, selectDay;
21
22     onCreate() {
```




```

24 protected void onCreate(Bundle savedInstanceState) {
25     super.onCreate(savedInstanceState);
26     setContentView(R.layout.activity_main);
27
28     setTitle("시간 예약");
29
30     // 버튼
31     btnStart = (Button) findViewById(R.id.btnStart);
32     btnEnd = (Button) findViewById(R.id.btnEnd);
33
34     // 크로노미터
35     chrono = (Chronometer) findViewById(R.id.chronometer1);
36
37     // 라디오버튼 2개
38     rdoCal = (RadioButton) findViewById(R.id.rdoCal);
39     rdoTime = (RadioButton) findViewById(R.id.rdoTime);
40
41     // FrameLayout의 2개 위젯
42     tPicker = (TimePicker) findViewById(R.id.timePicker1);
43     calView = (CalendarView) findViewById(R.id.calendarView1);
44
45     // 텍스트뷰 중에서 연, 월, 일, 시, 분 숫자
46     tvYear = (TextView) findViewById(R.id.tvYear);
47     tvMonth = (TextView) findViewById(R.id.tvMonth);
48     tvDay = (TextView) findViewById(R.id.tvDay);
49     tvHour = (TextView) findViewById(R.id.tvHour);
50     tvMinute = (TextView) findViewById(R.id.tvMinute);
51
52     // 처음에는 2개를 안보이게 설정
53     tPicker.setVisibility(View.INVISIBLE);
54     calView.setVisibility(View.INVISIBLE);

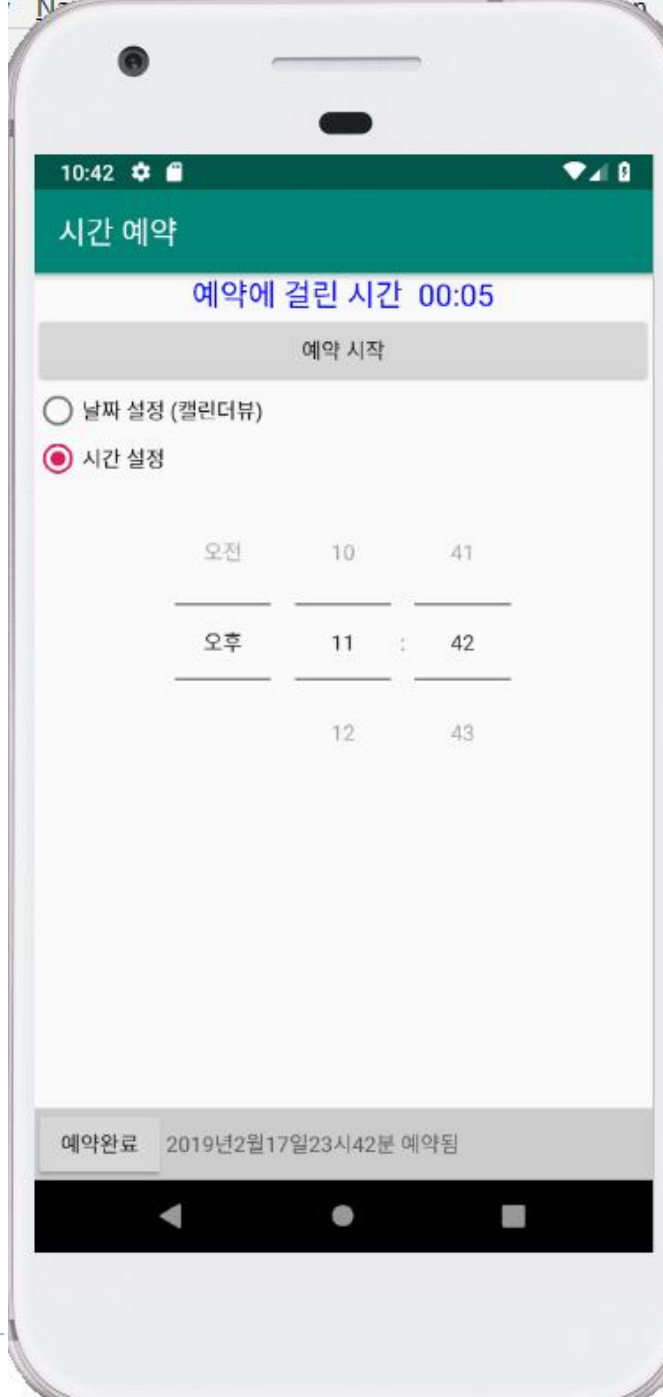
```

55
56
57
58
59
60
61
62
63
64
65
66
67
68
69

```
rdoCal.setOnClickListener(new View.OnClickListener() {  
    public void onClick(View v) {  
        tPicker.setVisibility(View.INVISIBLE);  
        calView.setVisibility(View.VISIBLE);  
    }  
});  
  
rdoTime.setOnClickListener(new View.OnClickListener() {  
    public void onClick(View v) {  
        tPicker.setVisibility(View.VISIBLE);  
        calView.setVisibility(View.INVISIBLE);  
    }  
});
```

// 버튼을 클릭하면 날짜, 시간을 가져온다.

```
btnEnd.setOnClickListener(new View.OnClickListener() {  
    public void onClick(View v) {  
        chrono.stop();  
        chrono.setTextColor(Color.BLUE);  
  
        tvYear.setText(Integer.toString(selectYear));  
        tvMonth.setText(Integer.toString(selectMonth));  
        tvDay.setText(Integer.toString(selectDay));  
        tvHour.setText(Integer.toString(tPicker.getCurrentHour()));  
        tvMinute.setText(Integer.toString(tPicker.getCurrentMinute()));  
    }  
});  
  
calView.setOnDateChangeListener(new CalendarView.OnDateChangeListener() {  
    @Override  
    public void onSelectedDayChange(@NonNull CalendarView view, int year, int month, int dayOfMonth) {  
        selectYear = year;  
        selectMonth = month + 1;  
        selectDay = dayOfMonth;  
    }  
});
```



텍스트뷰- 자동완성, 멀티자동완성

- ▶ 자동완성 텍스트뷰
 - ▶ 사용자가 단어의 일부만 입력해도 자동완성되는 에디트텍스트 (1개 단어만)
- ▶ 멀티자동완성 텍스트뷰
 - ▶ 쉼표로 구분하여 여러 개 단어가 자동 완성 됨





Create Android Project

Application name

Company domain

Project location



Package name

☐ Include C++ support☐ Include Kotlin support

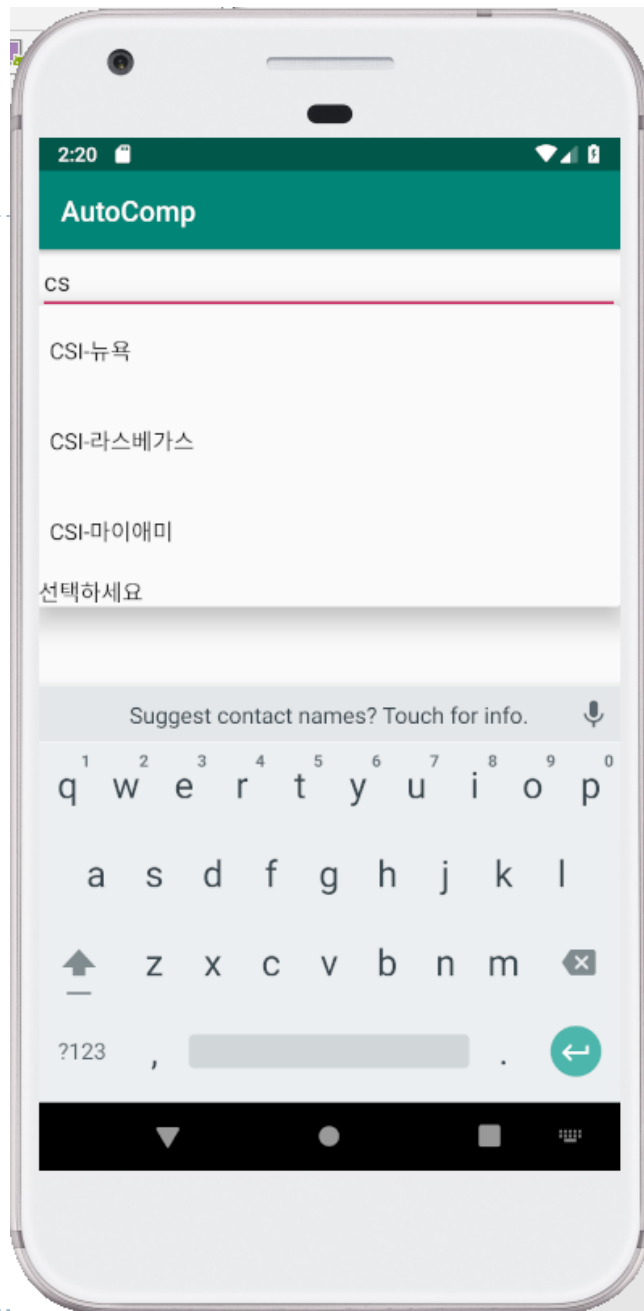
```

1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:tools="http://schemas.android.com/tools"
4     android:layout_width="match_parent"
5     android:layout_height="match_parent"
6     android:orientation="vertical" >
7
8     <AutoCompleteTextView
9         android:id="@+id/autoCompleteTextView1"
10        android:layout_width="match_parent"
11        android:layout_height="wrap_content"
12        android:completionHint="선택하세요"
13        android:completionThreshold="2"
14        android:hint="자동완성텍스트뷰" >
15    </AutoCompleteTextView>
16
17    <MultiAutoCompleteTextView
18        android:id="@+id/multiAutoCompleteTextView1"
19        android:layout_width="match_parent"
20        android:layout_height="wrap_content"
21        android:completionHint="선택하세요"
22        android:completionThreshold="2"
23        android:hint="멀티자동완성텍스트뷰" />
24
25 </LinearLayout>

```



```
5 import android.widget.ArrayAdapter;
6 import android.widget.AutoCompleteTextView;
7 import android.widget.MultiAutoCompleteTextView;
8 import android.widget.MultiAutoCompleteTextView.CommaTokenizer;
9
10 public class MainActivity extends AppCompatActivity {
11
12     @Override
13     protected void onCreate(Bundle savedInstanceState) {
14         super.onCreate(savedInstanceState);
15         setContentView(R.layout.activity_main);
16
17         String[] items = { "CSI-뉴욕", "CSI-라스베가스", "CSI-마이애미", "Friends",
18             "Fringe", "Lost" };
19
20         AutoCompleteTextView auto = (AutoCompleteTextView) findViewById(R.id.autoCompleteTextView1);
21         ArrayAdapter<String> adapter = new ArrayAdapter<String>(context.this,
22             android.R.layout.simple_dropdown_item_1line, items);
23         auto.setAdapter(adapter);
24
25         MultiAutoCompleteTextView multi = (MultiAutoCompleteTextView) findViewById(R.id.multiAutoCompleteTextView1);
26         CommaTokenizer token = new CommaTokenizer();
27         multi.setTokenizer(token);
28         multi.setAdapter(adapter);
29     }
30 }
```

ProgressBar, SeekBar, RatingBar

▶ ProgressBar

- ▶ 작업의 진행 상황을 표시
 - ▶ max: 범위 지정
 - ▶ progress: 시작 지점 지정
 - ▶ secondaryProgress: 두 번 째 프로그레스바 지정

▶ SeekBar

- ▶ 음량 조절이나 동영상 재생시 사용자가 재생위치를 지정

▶ RatingBar

- ▶ 선호도 표시에 주로 사용
 - ▶ numStars: 별의 개수
- ▶ rating: 초기값 지정
- ▶ stepSize: 한번에 채워지는 개수를 정하는 것





Create Android Project

Application name


Company domain

Project location

...

Package name

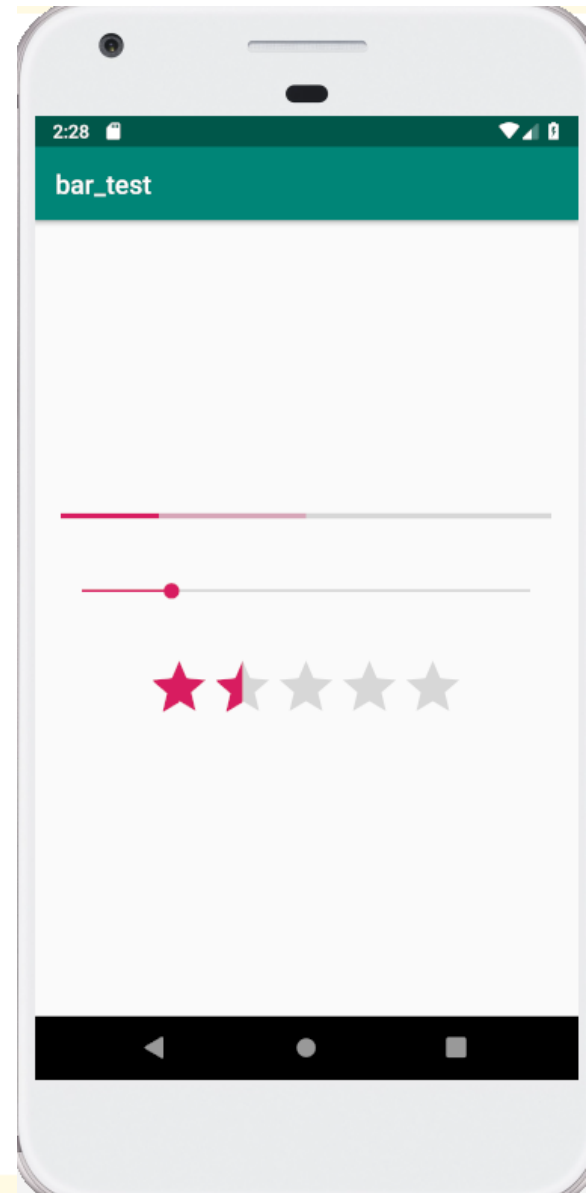
☐ Include C++ support☐ Include Kotlin support

 The application name for most apps begins with an uppercase letter

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:tools="http://schemas.android.com/tools"
4   android:layout_width="fill_parent"
5   android:layout_height="fill_parent"
6   android:gravity="center"
7   android:orientation="vertical" >
8
9   <ProgressBar
10     style="?android:attr/progressBarStyleHorizontal"
11     android:layout_width="match_parent"
12     android:layout_height="wrap_content"
13     android:layout_margin="20dp"
14     android:max="100"
15     android:progress="20"
16     android:secondaryProgress="50" />
17
18   <SeekBar
19     android:layout_width="match_parent"
20     android:layout_height="wrap_content"
21     android:layout_margin="20dp"
22     android:progress="20" />
23
24   <RatingBar
25     android:layout_width="wrap_content"
26     android:layout_height="wrap_content"
27     android:layout_margin="20dp"
28     android:numStars="5"
29     android:rating="1.5"
30     android:stepSize="0.5" />
31
32 </LinearLayout>

```



뷰 컨테이너

- ▶ **ViewGroup** 클래스 하위의 위젯
 - ▶ 다른 위젯을 담는 컨테이너 역할
 - ▶ 예) **ScrollView**
 - ✓ 스크롤뷰는 수직(위아래)으로 스크롤하는 기능
 - ✓ 수평(좌우)으로 스크롤하는 수평스크롤뷰 (**HorizontalScrollView**)는 따로 있음
 - ✓ 주의할 점은 스크롤뷰에는 단 하나의 위젯만 넣을 수 있음

```
java.lang.Object
```

```
    ↳ android.view.View
```

```
        ↳ android.widget.ViewGroup
```

```
            ↳ android.widget.FrameLayout
```

```
                ↳ android.widget.ScrollView
```

스크롤뷰 계층도



```
1
2 <ScrollView xmlns:android="http://www."
3     android:layout_width="fill_parent"
4     android:layout_height="fill_parent"
5     android:orientation="vertical" >
6
7     <LinearLayout
8         android:layout_width="fill_parent"
9         android:layout_height="fill_parent"
10        android:orientation="vertical" >
11
12        <Button
13            android:layout_width="match_parent"
14            android:layout_height="100dp"
15            android:text="버튼 1" />
16        ~~~~ 중간 생략 (버튼 5개) ~~~~
17    </LinearLayout>
18
19 </ScrollView>
```



뷰컨테이너 – 슬라이딩드로어

- ▶ 위젯들을 서랍처럼 열어서 보여주거나 닫아서 감춤

```
<슬라이딩드로어 handle="핸들명" content="콘텐츠명" >  
  <버튼 id="핸들명" />  // 서랍 손잡이 역할  
  <리니어레이아웃 id="콘텐츠명">  
  
    // 이곳이 서랍 내부이며, 필요한 위젯들을 넣으면 됨  
  
  </리니어레이아웃>  
</슬라이딩드로어>
```



Create Android Project

Application name**Company domain****Project location**

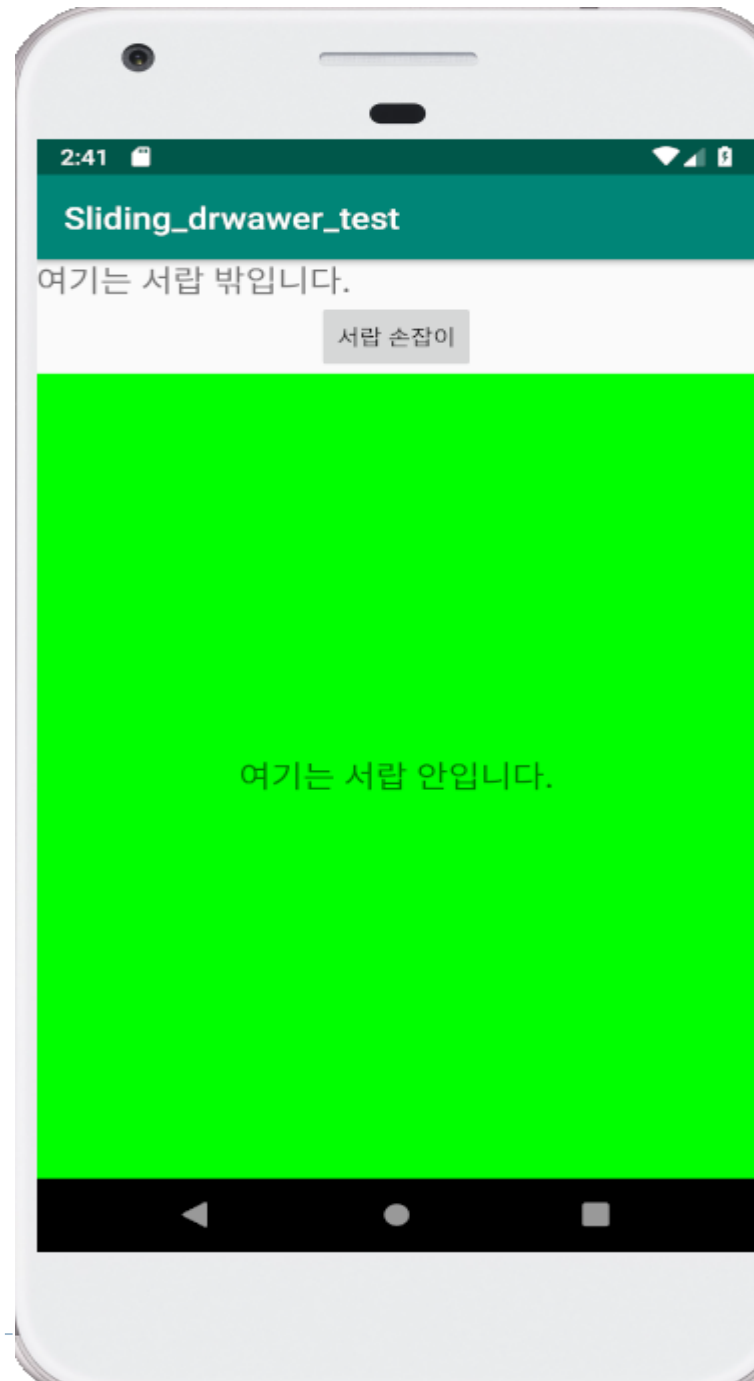
...

Package name☐ Include C++ support☐ Include Kotlin support


```

1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:tools="http://schemas.android.com/tools"
4     android:layout_width="match_parent"
5     android:layout_height="match_parent"
6     android:orientation="vertical" >
7
8     <TextView
9         android:layout_width="match_parent"
10        android:layout_height="wrap_content"
11        android:text="여기는 서랍 밖입니다."
12        android:textSize="20dp" />
13
14    <SlidingDrawer
15        android:id="@+id/slidingDrawer1"
16        android:layout_width="match_parent"
17        android:layout_height="match_parent"
18        android:content="@+id/content"
19        android:handle="@+id/handle" >
20
21        <Button
22            android:id="@+id/handle"
23            android:layout_width="wrap_content"
24            android:layout_height="wrap_content"
25            android:text="서랍 손잡이" />
26
27        <LinearLayout
28            android:id="@+id/content"
29            android:layout_width="match_parent"
30            android:layout_height="match_parent"
31            android:background="#00ff00"
32            android:gravity="center" >
33
34            <TextView
35                android:layout_width="wrap_content"
36                android:layout_height="wrap_content"
37                android:text="여기는 서랍 안입니다."
38                android:textSize="20dp" />
39
40        </LinearLayout>
41    </SlidingDrawer>
42 </LinearLayout>

```



복잡한 기능의 뷰 컨테이너 - 뷰플리퍼

▶ 뷰플리퍼

- ▶ 여러 개의 위젯을 배치한 후, 필요에 따라서 화면을 왼쪽과 오른쪽으로 밀어서 하나의 위젯씩 화면에 보여주는 방식

<리니어레이아웃>

<리니어레이아웃>

// 왼쪽/오른쪽으로 전환할 버튼 또는 이미지뷰

</리니어레이아웃>

<뷰플리퍼>

// 여기에 한번에 하나씩 보여줄 위젯들을 넣음

</뷰플리퍼 >

</리니어레이아웃>

```
1 <LinearLayout>
2   <LinearLayout
3     android:orientation="horizontal" >
4     <Button
5       android:id="@+id/btnPrev"
6       android:text="이전화면" />
7     <Button
8       android:id="@+id/btnNext"
9       android:text="다음화면" />
10  </LinearLayout>
11  <ViewFlipper
12    android:id="@+id/viewFlipper1">
13    <LinearLayout
14      android:background="#ff0000" >
15      ~~~~ 이곳에 필요한 위젯 삽입 ~~~~
16    </LinearLayout>
17    <LinearLayout
18      android:background="#00ff00" >
19      ~~~~ 이곳에 필요한 위젯 삽입 ~~~~
20    </LinearLayout>
21    <LinearLayout
22      android:background="#0000ff" >
23      ~~~~ 이곳에 필요한 위젯 삽입 ~~~~
24    </LinearLayout>
25  </ViewFlipper>
26 </LinearLayout>
```



Create Android Project








Application name**Company domain****Project location**

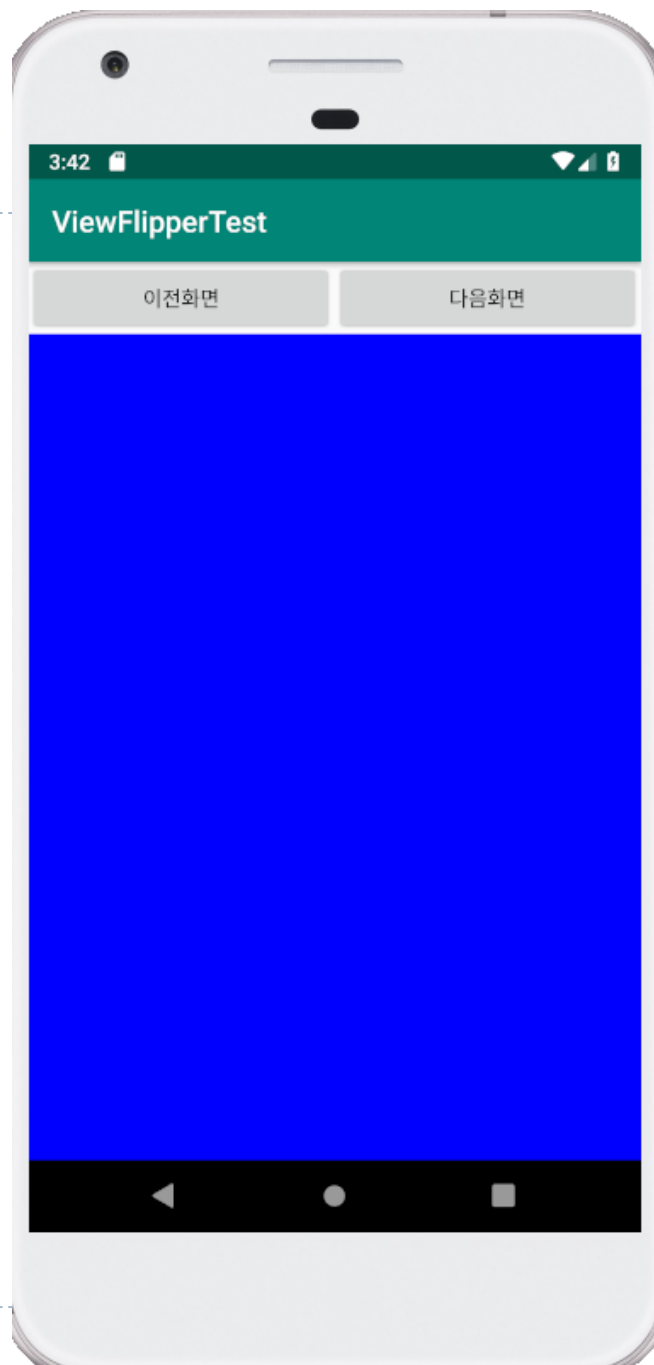
...

Package name☐ Include C++ support☐ Include Kotlin support

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:tools="http://schemas.android.com/tools"
4     android:layout_width="match_parent"
5     android:layout_height="match_parent"
6     android:orientation="vertical" >
7
8     <LinearLayout
9         android:layout_width="match_parent"
10        android:layout_height="wrap_content"
11        android:orientation="horizontal" >
12
13        <Button
14            android:id="@+id/btnPrev"
15            android:layout_width="wrap_content"
16            android:layout_height="wrap_content"
17            android:layout_weight="1"
18            android:text="이전화면" />
19
20        <Button
21            android:id="@+id/btnNext"
22            android:layout_width="wrap_content"
23            android:layout_height="wrap_content"
24            android:layout_weight="1"
25            android:text="다음화면" />
26    </LinearLayout>
```

```
28 <ViewFlipper
29     android:id="@+id/viewFlipper1"
30     android:layout_width="match_parent"
31     android:layout_height="match_parent" >
32
33     <LinearLayout
34         android:layout_width="match_parent"
35         android:layout_height="match_parent"
36         android:background="#ff0000" >
37
38
39     <LinearLayout
40         android:layout_width="match_parent"
41         android:layout_height="match_parent"
42         android:background="#00ff00" >
43
44
45     <LinearLayout
46         android:layout_width="match_parent"
47         android:layout_height="match_parent"
48         android:background="#0000ff" >
49
50 </ViewFlipper>
51
52 </LinearLayout>
```

```
12    protected void onCreate(Bundle savedInstanceState) {  
13     super.onCreate(savedInstanceState);  
14     setContentView(R.layout.activity_main);  
15  
16     Button btnPrev, btnNext;  
17     final ViewFlipper vFlipper;  
18  
19     btnPrev = (Button) findViewById(R.id.btnPrev);  
20     btnNext = (Button) findViewById(R.id.btnNext);  
21  
22     vFlipper = (ViewFlipper) findViewById(R.id.viewFlipper1);  
23  
24     btnPrev.setOnClickListener(new View.OnClickListener() {  
25   public void onClick(View v) {  
26         vFlipper.showPrevious();  
27     }  
28 });  
29  
30     btnNext.setOnClickListener(new View.OnClickListener() {  
31   public void onClick(View v) {  
32         vFlipper.showNext();  
33     }  
34 });  
35
```



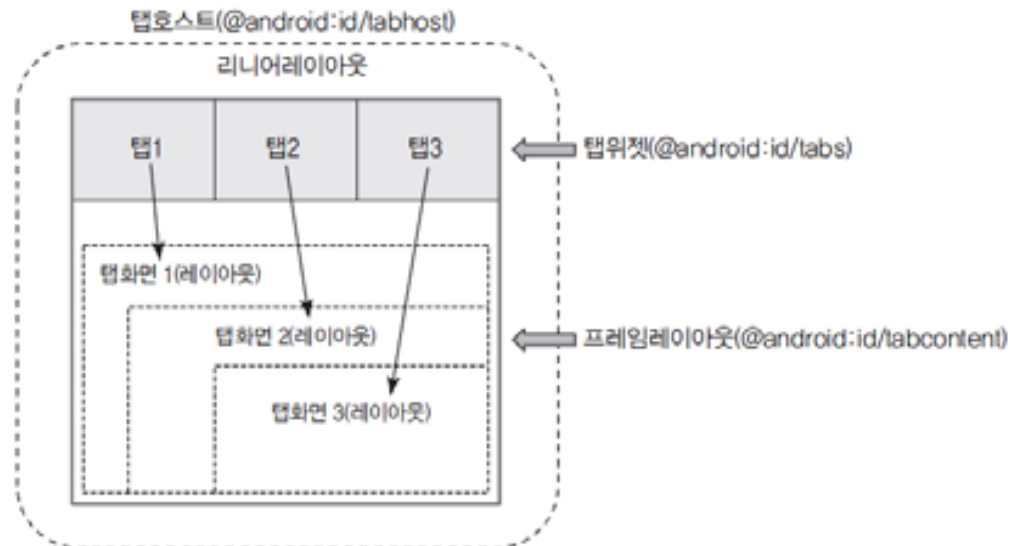
복잡한 기능의 뷰 컨테이너 - 탭 호스트

- ▶ 여러 탭을 두고 각 탭을 클릭할 때 해당 화면 전환

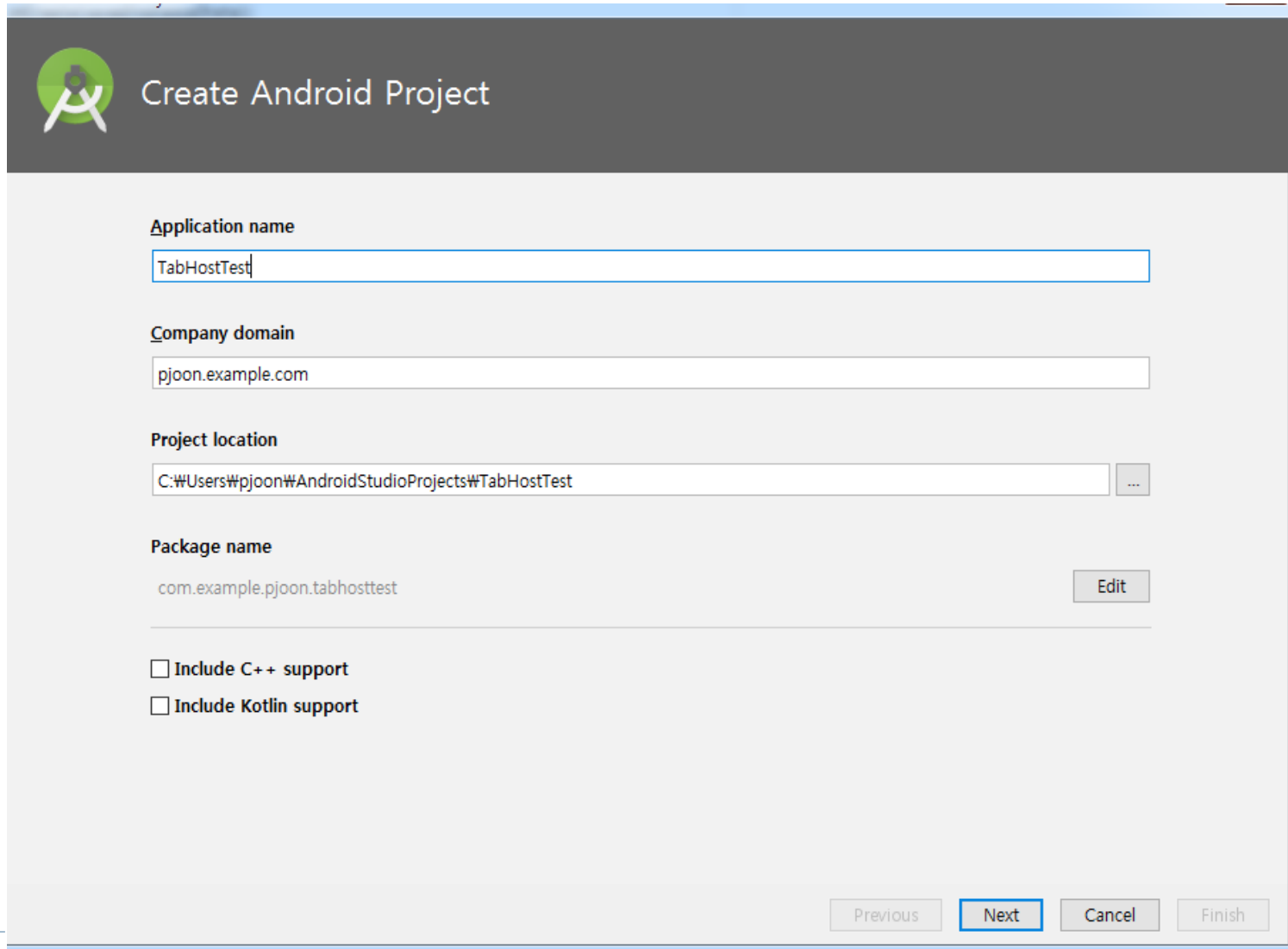
```
<탭호스트 id="@android:id/tabhost">  
  <리니어레이아웃>  
    <탭위젯 id="@android:id/tabs" />  
    <프레임레이아웃 id="@android:id/tabcontent">
```


// 여기에 각 탭스펙에 대응할 탭화면(레이아웃)을 3개 넣음

```
</프레임레이아웃>  
</리니어레이아웃>  
</탭호스트>
```



복잡한 기능의 뷰 컨테이너 - 탭 호스트



 Create Android Project

Application name
TabHostTest

Company domain
pjoon.example.com

Project location
C:\Users\pjoon\AndroidStudioProjects\TabHostTest ...

Package name
com.example.pjoon.tabhosttest Edit

☐ Include C++ support
☐ Include Kotlin support

Previous Next Cancel Finish

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <TabHost xmlns:android="http://schemas.android.com/apk/res/android"
3         xmlns:tools="http://schemas.android.com/tools"
4         android:id="@android:id/tabhost"
5         android:layout_width="match_parent"
6         android:layout_height="match_parent" >
7
8     <LinearLayout
9         android:id="@+id/linearLayout1"
10        android:layout_width="match_parent"
11        android:layout_height="match_parent"
12        android:orientation="vertical" >
13
14        <TabWidget
15            android:id="@android:id/tabs"
16            android:layout_width="match_parent"
17            android:layout_height="wrap_content" >
18        </TabWidget>
19
20        <FrameLayout
21            android:id="@android:id/tabcontent"
22            android:layout_width="match_parent"
23            android:layout_height="match_parent" >
```

```
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
```

```
<LinearLayout
    android:id="@+id/tabSong"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#f00000"
    android:orientation="vertical">
</LinearLayout>

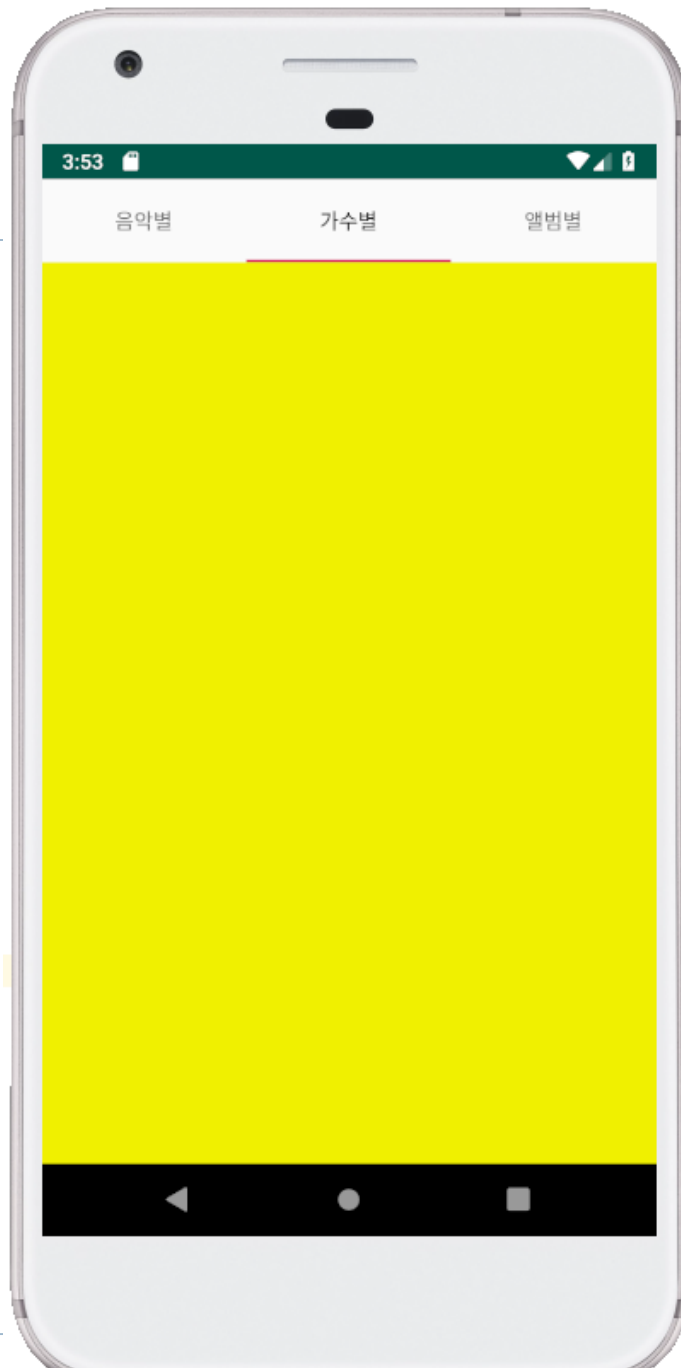
<LinearLayout
    android:id="@+id/tabArtist"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#ff0000"
    android:orientation="vertical">
</LinearLayout>

<LinearLayout
    android:id="@+id/tabAlbum"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#ff00ff"
    android:orientation="vertical">
</LinearLayout>
</FrameLayout>
</LinearLayout>
</TabHost>
```

```

7
8 public class MainActivity extends TabActivity {
9
10     @Override
11     protected void onCreate(Bundle savedInstanceState) {
12         super.onCreate(savedInstanceState);
13         setContentView(R.layout.activity_main);
14
15         TabHost tabHost = getTabHost();
16
17         TabHost.TabSpec tabSpecSong = tabHost.newTabSpec( tag: "SONG").setIndicator("음악별");
18         tabSpecSong.setContent(R.id.tabSong);
19         tabHost.addTab(tabSpecSong);
20
21         TabHost.TabSpec tabSpecArtist = tabHost.newTabSpec( tag: "ARTIST")
22             .setIndicator("가수별");
23         tabSpecArtist.setContent(R.id.tabArtist);
24         tabHost.addTab(tabSpecArtist);
25
26         TabHost.TabSpec tabSpecAlbum = tabHost.newTabSpec( tag: "ALBUM").setIndicator("앨범별");
27         tabSpecAlbum.setContent(R.id.tabAlbum);
28         tabHost.addTab(tabSpecAlbum);
29
30         tabHost.setCurrentTab(0);
31
32     }
33 }

```



복잡한 기능의 뷰 컨테이너 - 웹뷰

- ▶ 사용자가 웹브라우저 기능을 앱 안에 포함시킬 수 있는 위젯
- ▶ 에디트텍스트에 URL 입력 → 해당 링크로 이동하는 웹브라우저 앱





Create Android Project

Application name**Company domain****Project location****Package name**☐ Include C++ support☐ Include Kotlin support


```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:tools="http://schemas.android.com/tools"
4     android:layout_width="fill_parent"
5     android:layout_height="fill_parent"
6     android:orientation="vertical" >
7
8     <LinearLayout
9         android:id="@+id/linearLayout1"
10        android:layout_width="match_parent"
11        android:layout_height="wrap_content" >
12
13        <EditText
14            android:id="@+id/edtUri"
15            android:layout_width="wrap_content"
16            android:layout_height="wrap_content"
17            android:layout_weight="1"
18            android:hint="URL을 입력하세요."
19            android:singleLine="true" >
20
21        </EditText>
22
23        <Button
24            android:id="@+id/btnGo"
25            android:layout_width="wrap_content"
26            android:layout_height="wrap_content"
27            android:text="이동" />
28
29        <Button
30            android:id="@+id/btnBack"
31            android:layout_width="wrap_content"
32            android:layout_height="wrap_content"
33            android:text="이전" />
34
35    </LinearLayout>
36
37    <WebView
38        android:id="@+id/webView1"
39        android:layout_width="match_parent"
40        android:layout_height="match_parent"
41        android:clickable="true" />
42
43</LinearLayout>
```

WebViewTest

URL을 입력하세요.

이동

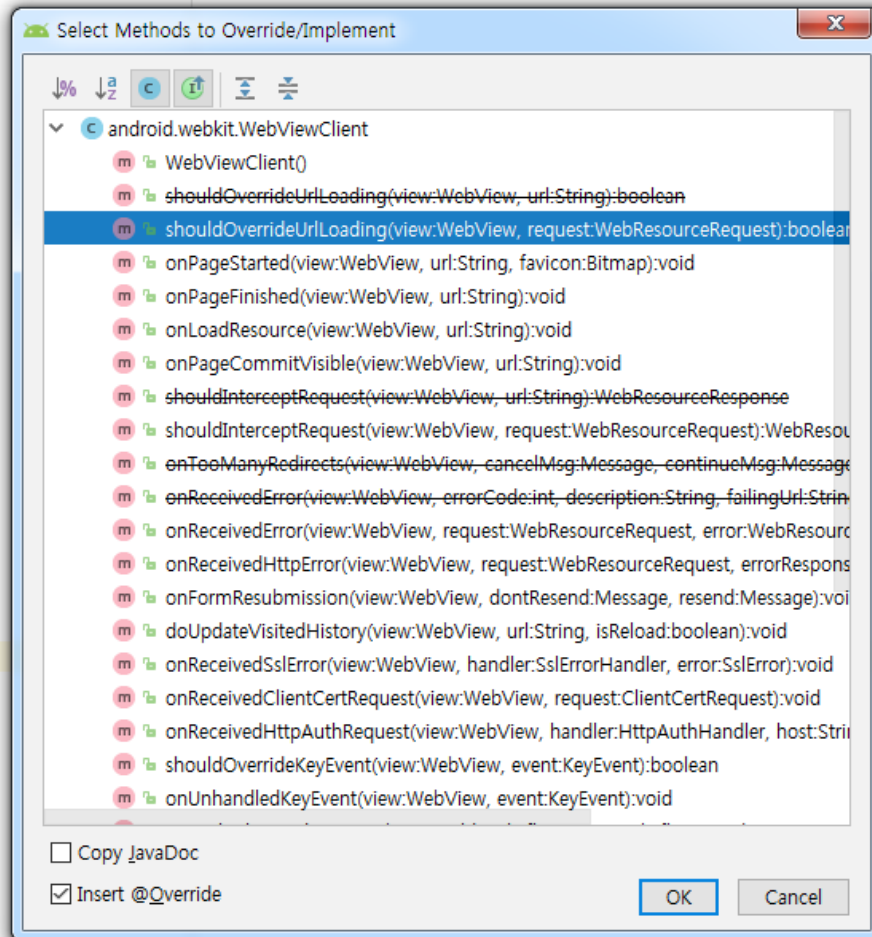
이전








WebView

```

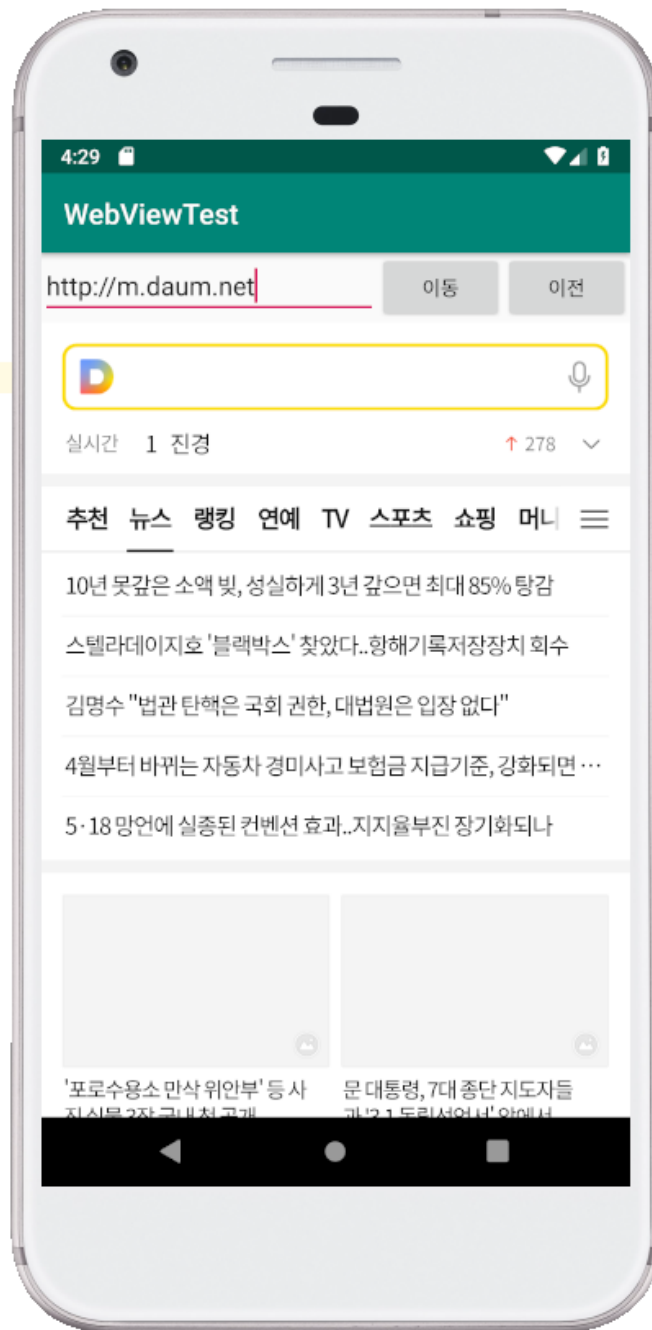
1  package com.example.pjoon.webviewtest;
2
3  import android.support.v7.app.AppCompatActivity;
4  import android.os.Bundle;
5  import android.webkit.WebView;
6  import android.webkit.WebViewClient;
7  import android.widget.Button;
8  import android.widget.EditText;
9
10 public class MainActivity extends AppCompatActivity {
11
12     EditText edtUrl;
13     Button btnGo, btnBack;
14     WebView web;
15
16     @Override
17     protected void onCreate(Bundle savedInstanceState) {
18         super.onCreate(savedInstanceState);
19         setContentView(R.layout.activity_main);
20
21         getSupportActionBar().setDisplayHomeAsUpEnabled(true);
22
23         edtUrl = (EditText) findViewById(R.id.edtUrl);
24         btnGo = (Button) findViewById(R.id.btnGo);
25         btnBack = (Button) findViewById(R.id.btnBack);
26         web = (WebView) findViewById(R.id.webView1);
27     }
28
29     class testClient extends WebViewClient {
30
31     }
32
33 }
34
35
36
37

```



```
20   protected void onCreate(Bundle savedInstanceState) {  
21     super.onCreate(savedInstanceState);  
22     setContentView(R.layout.activity_main);  
23  
24     getSupportActionBar().setDisplayHomeAsUpEnabled(true);  
25  
26     edtUrl = (EditText) findViewById(R.id.edtUrl);  
27     btnGo = (Button) findViewById(R.id.btnGo);  
28     btnBack = (Button) findViewById(R.id.btnBack);  
29     web = (WebView) findViewById(R.id.webView1);  
30  
31     web.setWebViewClient(new testClient());  
32  
33     WebSettings webSet = web.getSettings();  
34     webSet.setBuiltInZoomControls(true);  
35  
36     btnGo.setOnClickListener(new View.OnClickListener() {  
37           public void onClick(View v) {  
38             web.loadUrl(edtUrl.getText().toString());  
39         }  
40     });  
41  
42     btnBack.setOnClickListener(new View.OnClickListener() {  
43           public void onClick(View v) {  
44             web.goBack();  
45         }  
46     });  
47  
48  }
```

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="com.example.pjoon.webviewtest">
4
5     <uses-permission android:name="android.permission.INTERNET" />
6
7
8     <application
9         android:allowBackup="true"
10        android:icon="@mipmap/ic_launcher"
11        android:label="@string/webviewtest"
12        android:roundIcon="@mipmap/ic_launcher_round"
13        android:supportRtl="true"
14        android:theme="@style/AppTheme">
15        <activity android:name=".MainActivity">
16            <intent-filter>
17                <action android:name="android.intent.action.MAIN" />
18
19                <category android:name="android.intent.category.LAUNCHER" />
20            </intent-filter>
21        </activity>
22    </application>
23
24 </manifest>
```



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:tools="http://schemas.android.com/tools"
4   package="com.example.pjoon.webviewtest">
5
6   <uses-permission android:name="android.permission.INTERNET" />
7
8
9   <application
10     android:allowBackup="true"
11     android:icon="@mipmap/ic_launcher"
12     android:label="@string/webviewtest"
13     android:roundIcon="@mipmap/ic_launcher_round"
14     android:supportRtl="true"
15     android:usesCleartextTraffic="true"
16     android:theme="@style/AppTheme"
17     tools:targetApi="m"
18     tools:ignore="GoogleAppIndexingWarning">
19     <activity android:name=".MainActivity">
20       <intent-filter>
21         <action android:name="android.intent.action.MAIN" />
22
23         <category android:name="android.intent.category.LAUNCHER" />
24       </intent-filter>
25     </activity>
26   </application>
27
28 </manifest>
```