

# 모바일프로그래밍(2) -Mobile Programming(2)-

2020. 03. 22.

Tues.

Prepared by DaeKyeong Kim  
Ph.D.





# Agenda

## ❖ 교과목 개요 및 특징

- 본 강의는 '모바일 프로그래밍1'에서 습득한 안드로이드에 관한 기본적인 지식과 앱 개발 능력을 고도화하는 단계로서 기본적인 모바일 콘텐츠 개발 이외에 네트워킹과 DB, GPS 등의 이용 기법들을 숙지하여 실생활에 이용되는 다양한 콘텐츠 및 모바일 게임 등의 앱 개발 능력을 키운다.

## ❖ 교과목표(학습목표)

- 객체지향 언어인 JAVA의 기본 개념을 학습하고, 실전 프로그래밍 기법을 학습하고 애플레이터에서 실행하여 확인함으로써 모바일 기기를 이용한 정보 공유 및 콘텐츠를 개발할 수 있다.

# 학습일정 및 내용



| 주차 | 기간          | 수업내용 및 학습활동                     | 비고               |
|----|-------------|---------------------------------|------------------|
| 1  | 03/07-03/11 | · 강의 오리엔테이션<br>· 연구실 안전 정기교육 실시 | · 주교재 01장-04장 복습 |
| 2  | 03/14-03/18 | · 프로그램 작성하기                     | · 주교재 01장-04장 복습 |
| 3  | 03/21-03/25 | · 기본문법 활용하기<br>· 언어특성 활용하기      | · 주교재 01장-04장 복습 |
| 4  | 03/28-04/01 | · UI 요구사항 확인하기                  | · 주교재 05장        |
| 5  | 04/04~04/08 | · UI 요구사항 확인하기                  | · 주교재 06-07장     |
| 6  | 04/11~04/15 | · UI 설계하기                       | · 주교재 08장        |
| 7  | 04/18~04/22 | · UI 설계하기                       | · 주교재 09장        |
| 8  | 04/25~04/29 | · 중간고사                          |                  |

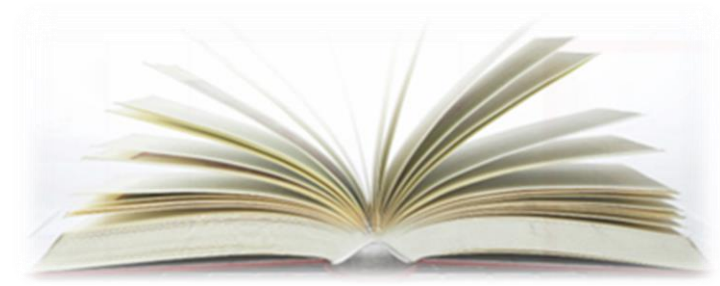
# 학습일정 및 내용



| 주차 | 기간          | 수업내용 및 학습활동             | 비고        |
|----|-------------|-------------------------|-----------|
| 9  | 05/02~05/06 | · 멀티미디어 연동하기/라이브러리 활용하기 | · 주교재 10장 |
| 10 | 05/09~05/13 | · 멀티미디어 연동하기/라이브러리 활용하기 | · 주교재 11장 |
| 11 | 05/16~05/20 | · 멀티미디어 연동하기/라이브러리 활용하기 | · 주교재 12장 |
| 12 | 05/23~06/03 | · 멀티미디어 연동하기/라이브러리 활용하기 | · 주교재 13장 |
| 13 | 06/06~06/10 | · 산출물 작성하기              | -         |
| 14 | 06/13~06/17 | · 산출물 작성하기              | -         |
| 15 | 06/20~06/24 | 기말고사                    |           |
|    |             |                         |           |

# Contents

---

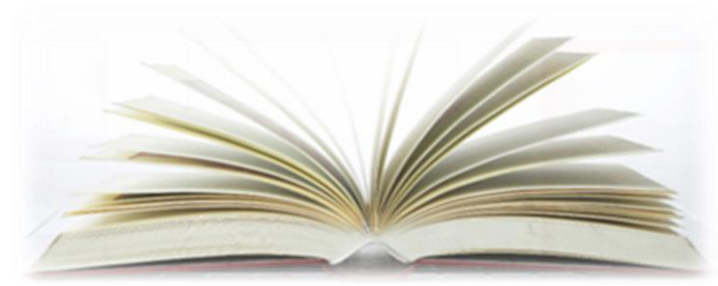


## Section 1 위젯

..... 3

# Section 1

## 위젯



1. 텍스트뷰

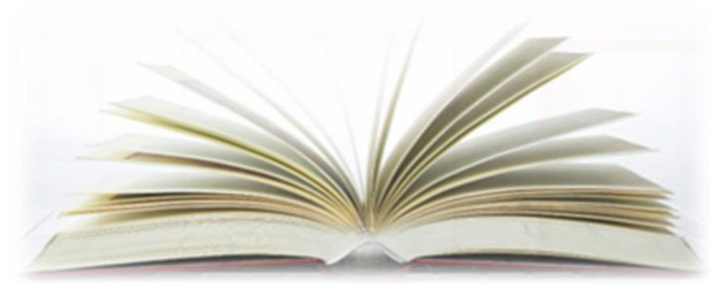
2. 버튼

3. 이미지뷰

4. 프로그레스바

# 학습목표

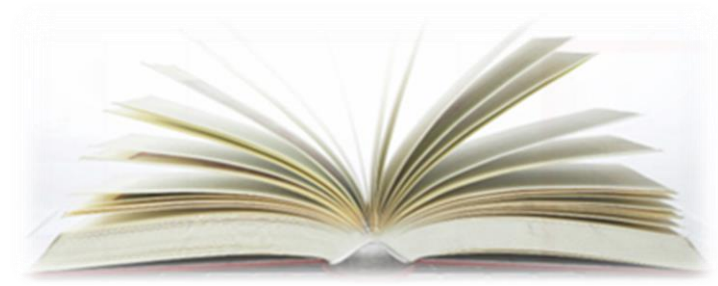
---



- ❖ 이 워크샵에서는 위젯의 클래스 계층 관계를 이해한다.
- ❖ 텍스트뷰의 속성과 사용법을 익힌다.
- ❖ 버튼의 속성과 사용법을 익힌다.
- ❖ 이미지뷰의 속성과 사용법을 익힌다.
- ❖ 프로그레스바의 속성과 사용법을 익힌다.

# Subsection 1

---



텍스트뷰



- ❖ 텍스트를 표시하는 기본 기능
- ❖ 속성을 추가하여 텍스트를 편집
- ❖ TextView: 자식 위젯에 공통 기능을 제공하므로 가장 먼저 학습
- ❖ CheckedTextView: 텍스트에 체크 표시 기능을 추가
- ❖ EditText: 텍스트를 편집하는 속성을 추가한 것
- ❖ Button과 그 이하 위젯: 서로 다른 특징을 가진 버튼 기능을 제공

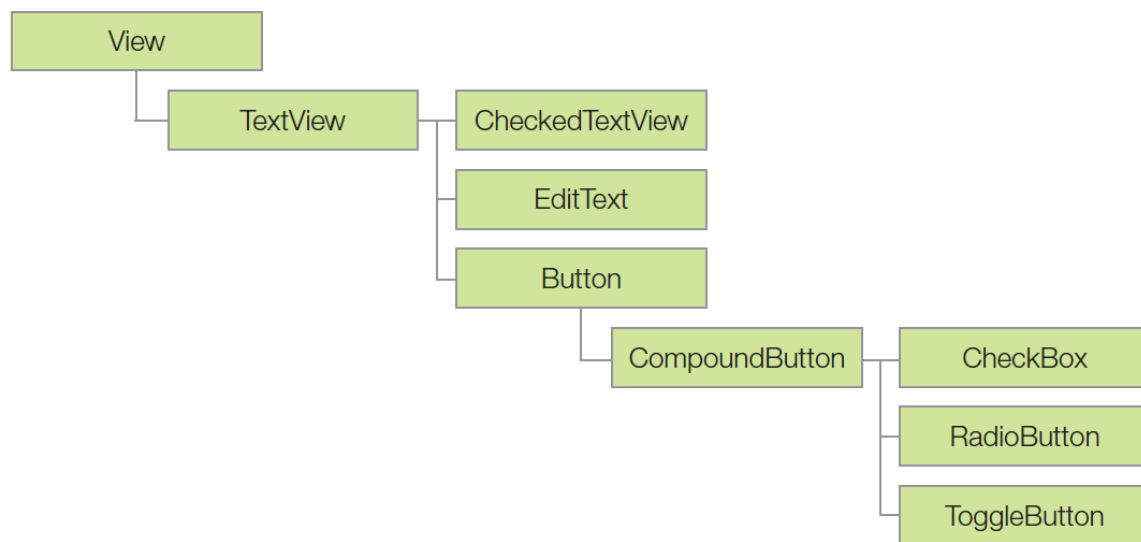


그림 3-1 텍스트뷰 관련 위젯들의 클래스 계층도

## ❖ TextView 속성

- View 클래스의 속성을 모두 가짐
- TextView만의 고유한 속성을 추가로 제공

표 3-1 TextView 속성 - 편집 기능 제외

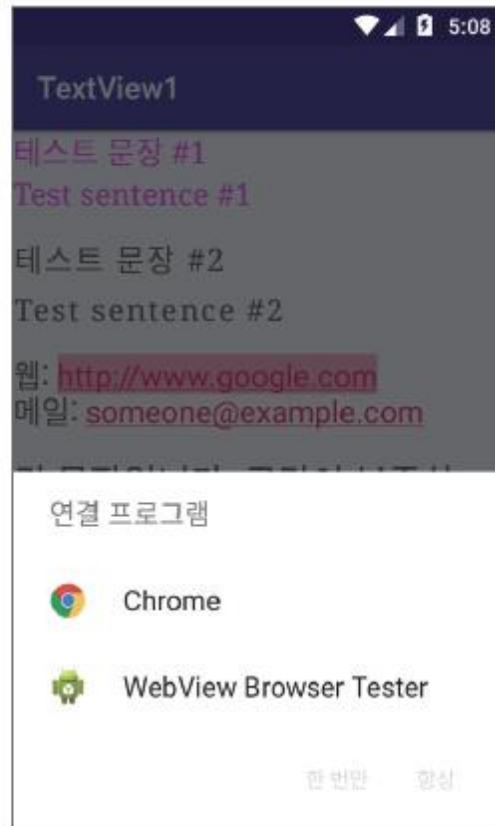
| XML 속성                        | 관련 메서드              | 기능   |
|-------------------------------|---------------------|--|
| android:text                  | setText()           | 표시할 텍스트를 지정한다.   |
| android:textColor             | setTextColor()      | 텍스트 색상을 지정한다.  |
| android:textSize              | setTextSize()       | 텍스트 크기를 지정한다.  |
| android:typeface              | setTypeface()       | 텍스트 서체를 "normal", "sans", "serif", "monospace" 중 하나로 지정한다.                   |
| android:textStyle             |                     | 텍스트 스타일을 "bold", "italic", "bolditalic" 중 하나로 지정한다.                          |
| android:textAppearance        | setTextAppearance() | 텍스트의 기본 색상/크기/서체/스타일을 한꺼번에 지정한다.   |
| android:letterSpacing         | setLetterSpacing()  | 글자 간격을 실숫값(예: 0.1)으로 지정한다. 기본값은 "0"이며, 음수도 가능하다.                             |
| android:lineSpacingExtra      | setLineSpacing()    | 줄 간격에 더할 값을 px/mm/in/pt/dp/sp 단위로 지정한다. 기본값은 "0dp"이며, 음수도 가능하다.              |
| android:lineSpacingMultiplier |                     | 줄 간격에 곱할 값을 실수(예: 1.2)로 지정한다. 기본값은 "1"이며, 음수는 안 된다.                          |
| android:gravity               | setGravity()        | TextView 내부에 텍스트를 표시할 위치를 지정한다. 예를 들어 속성값으로 "right"를 주면 텍스트가 오른쪽에 정렬되어 표시된다. |

|  |   |   |
|--|---|---|
| android:autoLink   | setAutoLinkMask()                         | 텍스트에 포함된 웹 주소, 이메일 주소, 전화 번호, 거리 주소(미국식)가 자동으로 링크되어 동작한다. 클릭하면 웹사이트가 열리거나 이메일을 보내거나 전화를 걸거나 지도 앱이 실행된다.   |
| android:singleLine   | setSingleLine()                           | 텍스트를 무조건 한 줄로 표시한다.   |
| android:ellipsize  | setEllipsize()                            | android:singleLine 속성을 준 상태에서 표시할 텍스트가 너무 길면 “...”을 표시한다. 속성값에 따라 앞(“start”), 중간(“middle”), 끝(“end”)에 “...”이 표시된다. “marquee” 값을 주면 전광판처럼 텍스트가 자동 스크롤된다. |
| android:drawableTop<br>android:drawableBottom<br>android:drawableLeft<br>android:drawableRight | setCompoundDrawablesWithIntrinsicBounds() | 텍스트의 위쪽/아래쪽/왼쪽/오른쪽에 그림을 표시한다.   |
| android:drawablePadding  | setCompoundDrawablePadding()              | 텍스트와 그림 사이에 여백을 준다. 기본값은 “0dp”이며, 음수도 가능하다.   |

## ❖ TextView의 다양한 속성

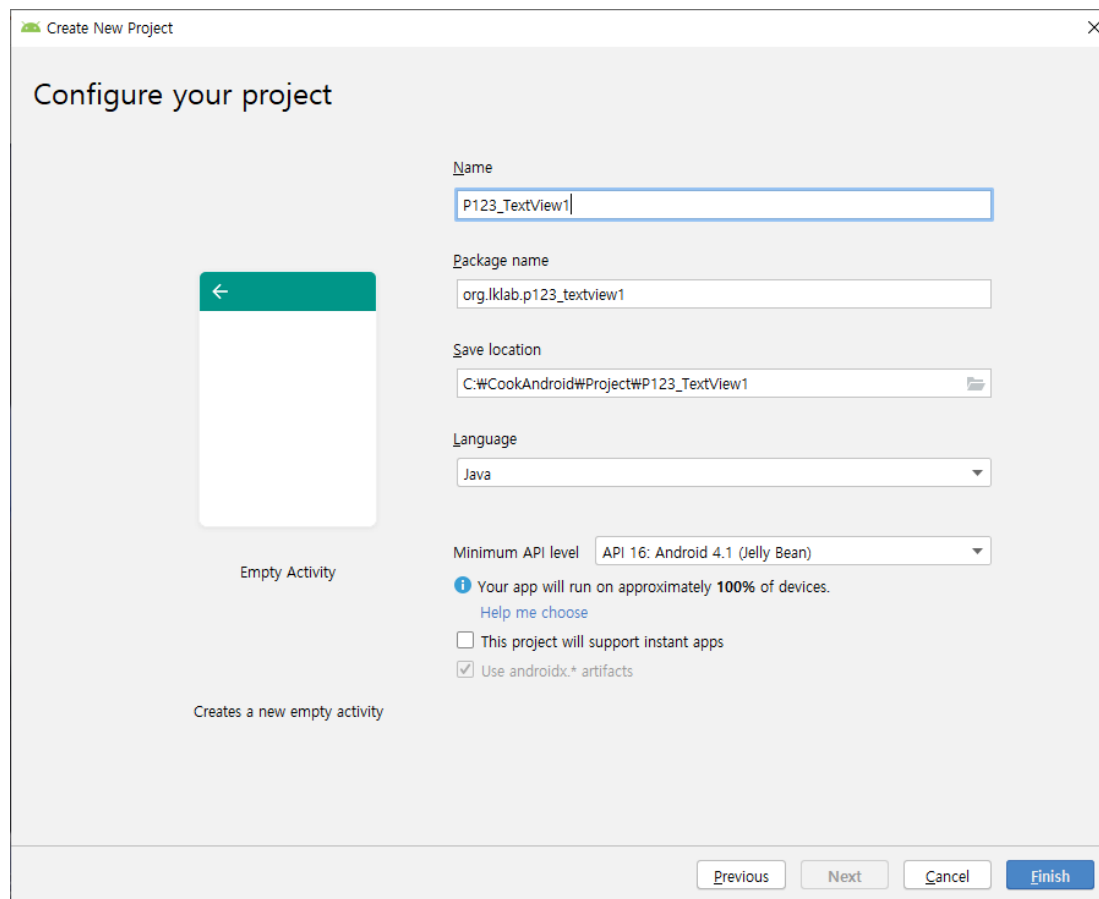


(a) 초기 화면



(b) 웹 주소 클릭 시

그림 3-2 실행 화면



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:text="테스트 문장 #1\nTest sentence #1"
        android:textColor="#ff00ff"
        android:textSize="20sp"
        android:typeface="serif"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:letterSpacing="0.05"
        android:lineSpacingMultiplier="1.2"
        android:text="테스트 문장 #2\nTest sentence #2"
        android:textSize="20sp"
        android:typeface="serif"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"
        android:autoLink="web | email"
        android:text="웹: http://www.google.com\n메일: someone@example.com"
        android:textSize="20sp"/>
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"
    android:ellipsize="end"
    android:singleLine="true"
    android:text="긴 문장입니다. 공간이 부족하면 생략 표시가 되는지 확인합니다."
    android:textSize="24sp"/>
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:drawablePadding="4dp"
    android:drawableTop="@mipmap/ic_launcher"
    android:text="앱 아이콘입니다."/>
</LinearLayout>
```

```
package org.lklab.p123_textview1;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

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



### ❖ CheckedTextView

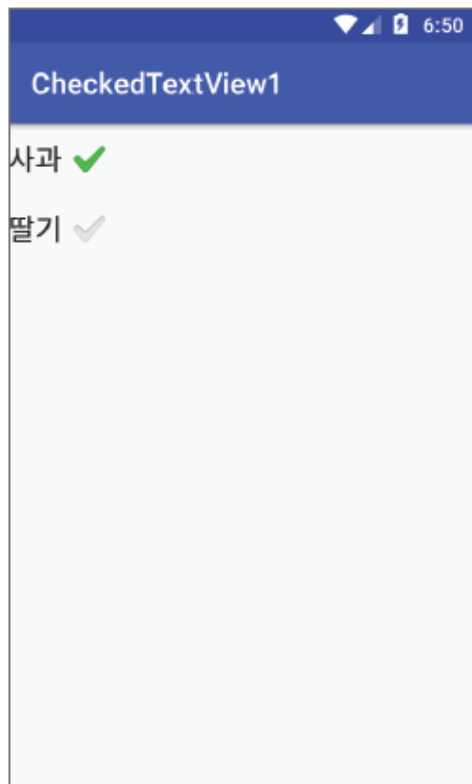
- TextView에 체크 표시 기능을 추가한 것

### ❖ 주요 속성

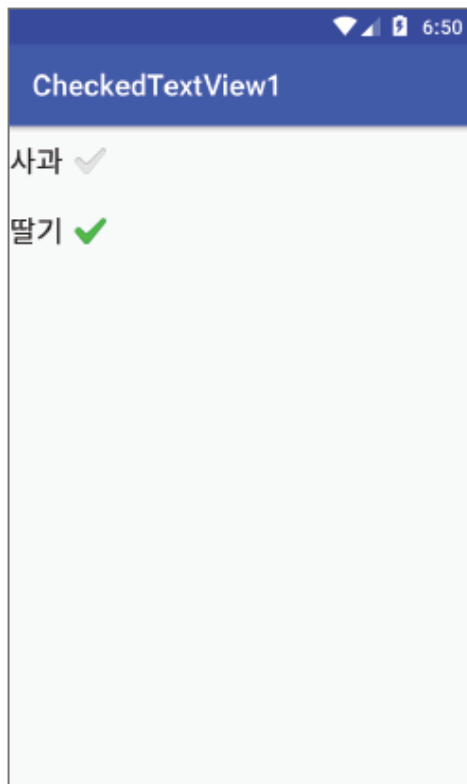
표 3-2 CheckedTextView 속성

| XML 속성            | 관련 메서드                 | 기능                                       |
|-------------------|------------------------|--|
| android:checked   | setChecked()           | 값이 "true"이면 체크 상태로 시작한다. 기본값은 "false"이다. |
| android:checkMark | setCheckMarkDrawable() | 체크 표시에 사용할 그림을 지정한다.                     |

## ❖ CheckedTextView를 이용한 출력과 입력 방법

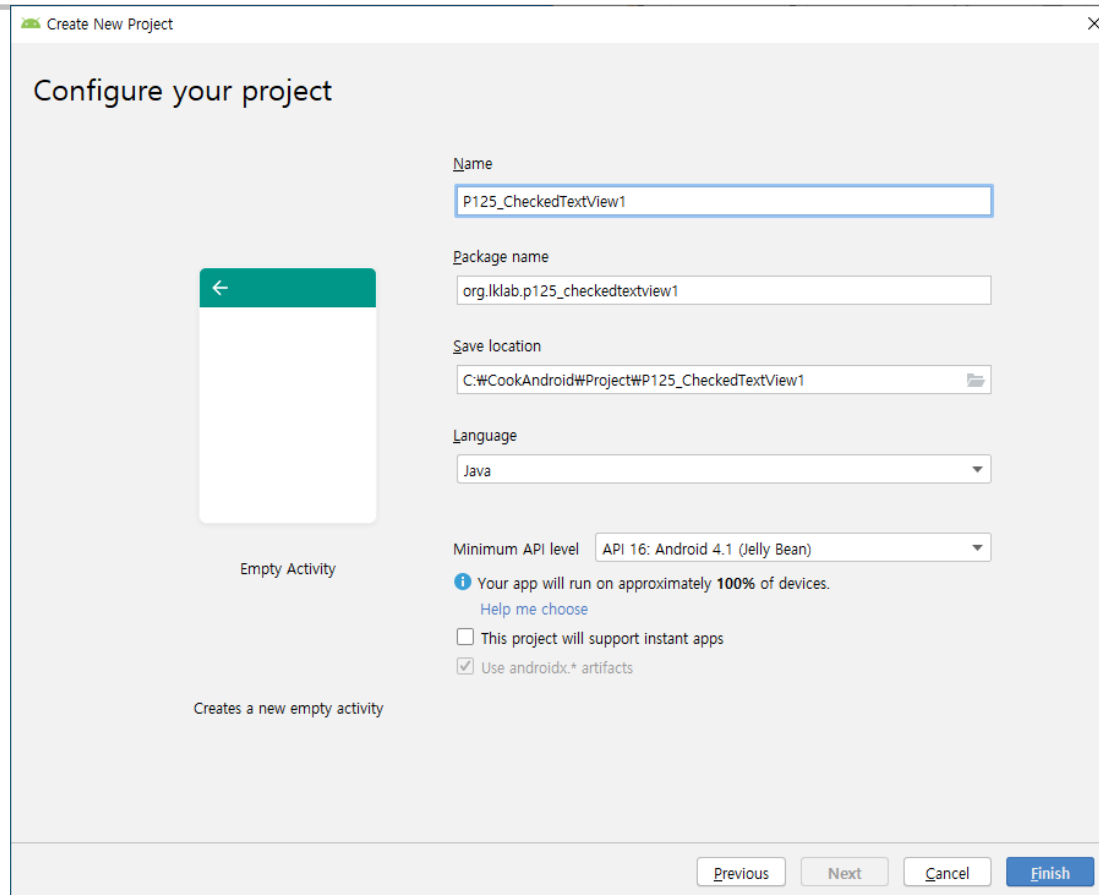


(a) 초기 화면



(b) 사과와 딸기를 한 번씩 클릭

그림 3-3 실행 화면



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <CheckedTextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkMark="?android:attr/textCheckMark"
        android:checked="true"
        android:gravity="center_vertical"
        android:onClick="mOnClick"
        android:text="사고 "
        android:textSize="20dp"/>
    <CheckedTextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkMark="?android:attr/textCheckMark"
        android:gravity="center_vertical"
        android:onClick="mOnClick"
        android:text="딸기 "
        android:textSize="20dp"/>
</LinearLayout>
```

```
package org.lklab.p125_checkedtextview1;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;
import android.widget.CheckedTextView;

public class MainActivity extends AppCompatActivity {

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

    public void mOnClick(View v) {
        CheckedTextView ctv = (CheckedTextView) v;
        ctv.setChecked(!ctv.isChecked());
        //ctv.toggle();
    }
}
```

## ❖ EditText

- 텍스트를 편집할 수 있도록 TextView의 기본 속성을 변경한 것
- 사실상 속성만 다른 TextView

표 3-3 TextView(= EditText) 속성 - 편집 기능 관련

| XML 속성                   | 관련 메서드                              | 기능   |
|--------------------------|-------------------------------------|--|
| android:hint             | setHint()                           | 텍스트가 비어 있을 때 대신 표시할 텍스트를 지정한다.   |
| android:ems              | setEms()                            | TextView의 폭을 지정한다. em은 M을 발음대로 쓴 것인데 대략 글자 한 개의 폭을 의미한다. 예를 들어 android:ems="10"이면 글자 열 개의 폭이다.   |
| android:lines            | setLines()                          | TextView의 높이를 줄 수로 지정한다.   |
| android:maxLength        | setFilters()                        | 입력 가능한 최대 글자 수를 지정한다.  |
| android:selectAllOnFocus | setSelectAllOnFocus()               | 속성값이 "true"이면 TextView를 선택 시 텍스트 전체가 선택되어 하이라이트 표시된다.  |
| android:digits           | setKeyListener()                    | 지정된 숫자만 입력받는다. 예를 들어 "12345"를 속성값으로 주면 1,2,3,4,5 숫자만 입력할 수 있다.   |
| android:inputType        | setRawInputType() 또는 setInputType() | 입력 텍스트의 형식을 지정한다. 예를 들어 이메일 주소나 숫자만 입력할 수 있도록 제한할 수 있다. [그림 3-4]에서 보듯이 안드로이드 스튜디오의 Design 탭에서 Text를 선택하면 inputType만 다른 다양한 EditText를 볼 수 있다. |

## ❖ EditText

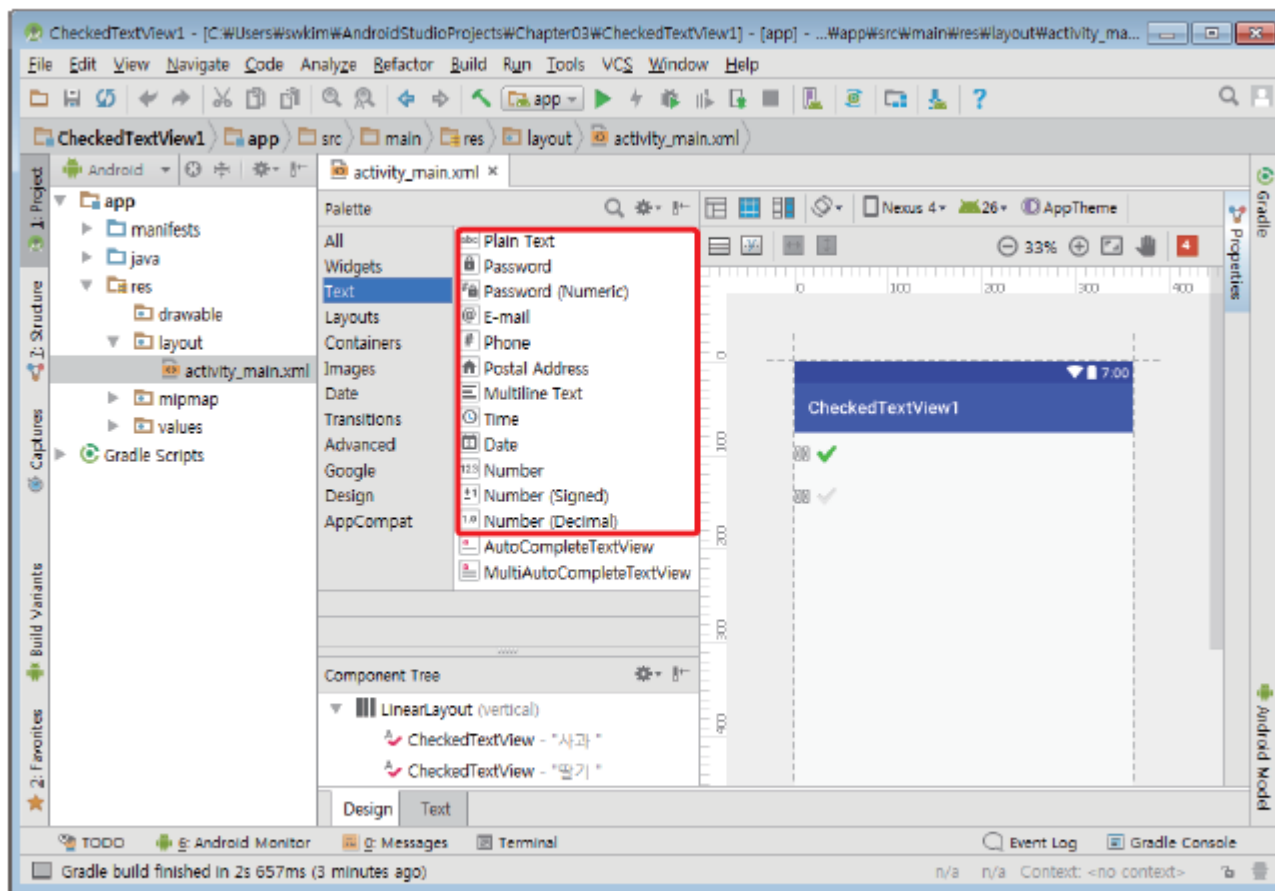


그림 3-4 inputType 속성에 따른 다양한 EditText

## ❖ requestFocus

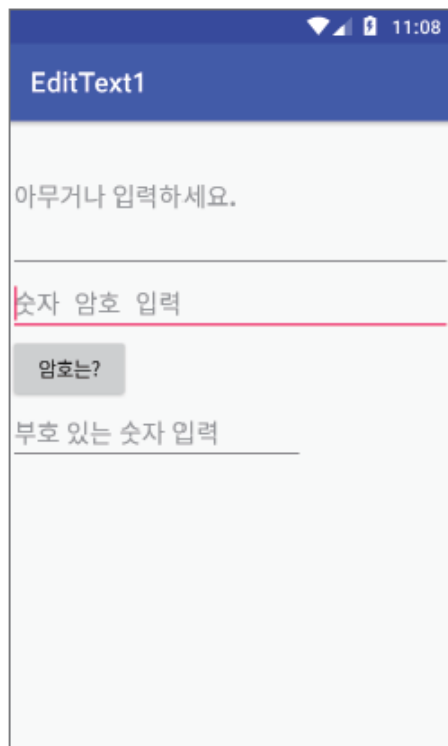
- XML 엘리먼트로, EditText와 더불어 유용하게 사용
- 여러 개의 EditText 중 키 입력을 최우선으로 받고 싶은 것이 있다면 requestFocus 엘리먼트를 자식으로 포함

```
<EditText
    android:id="@+id/editName"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <requestFocus/>
</EditText>
```

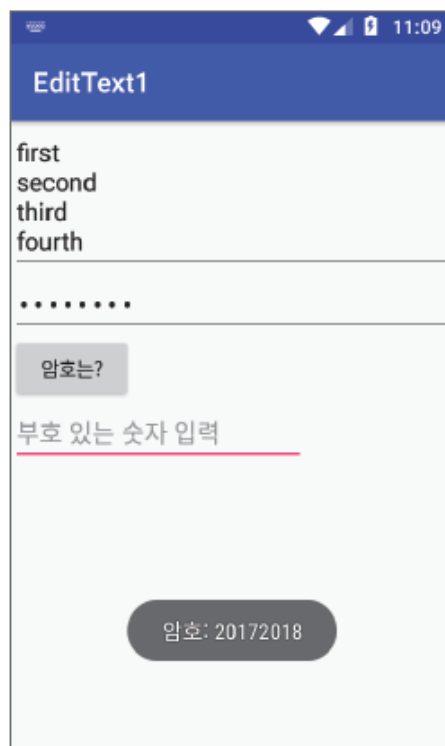
- EditText에 사용자가 입력한 내용은 자바 코드에서 getText ( ) 메서드로 획득
- 내용 변경 시 setText ( ) 메서드를 사용



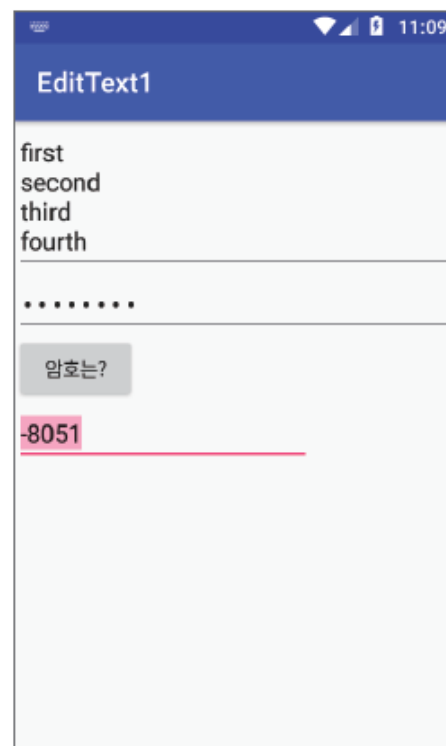
## ❖ EditText (= TextView)의 다양한 속성



(a) 초기 화면



(b) 텍스트 입력 테스트 (1)

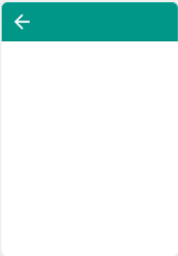


(c) 텍스트 입력 테스트 (2)

그림 3-5 실행 화면

Create New Project

Configure your project

 Empty Activity

Creates a new empty activity


Name  
P129\_EditText1

Package name  
org.klab.p129\_edittext1

Save location  
C:\CookAndroid\ProjectWP129\_EditText1

Language  
Java

Minimum API level  
API 16: Android 4.1 (Jelly Bean)

 Your app will run on approximately **100%** of devices.  
[Help me choose](#)

☐ This project will support instant apps

☒ Use android.\* artifacts

Previous Next Cancel Finish

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="아무거나 입력하세요."
        android:lines="4"/>
    <EditText
        android:id="@+id/editPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="숫자 암호 입력"
        android:inputType="numberPassword"
        android:maxLength="10">
        <requestFocus/>
    </EditText>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="mOnClick"
        android:text="암호는?"/>
    <EditText
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="부호 있는 숫자 입력"
        android:inputType="numberSigned"
        android:selectAllOnFocus="true"/>
</LinearLayout>
```

```
package org.lklab.p129_edittext1;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    private EditText mEditPassword;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mEditPassword = (EditText) findViewById(R.id.editPassword);
    }

    public void mOnClick(View v) {
        String password = mEditPassword.getText().toString();
        if (password.length() > 0) {
            Toast.makeText(this, "암호: " + password, Toast.LENGTH_SHORT).show();
        } else {
            Toast.makeText(this, "암호를 먼저 입력하세요!", Toast.LENGTH_SHORT).show();
        }
    }
}
```

- ❖ 텍스트뷰의 일종으로 출력보다는 입력을 위해 주로 사용
- ❖ Button 클래스: TextView에 추가 속성이 없는 거의 동일한 클래스
- ❖ CompoundButton 클래스: 추상 클래스로서 체크와 언체크 두 가지 상태를 추가한 것
- ❖ CompoundButton의 서브클래스들: 모양만 조금 다를 뿐 추가 속성은 없다.

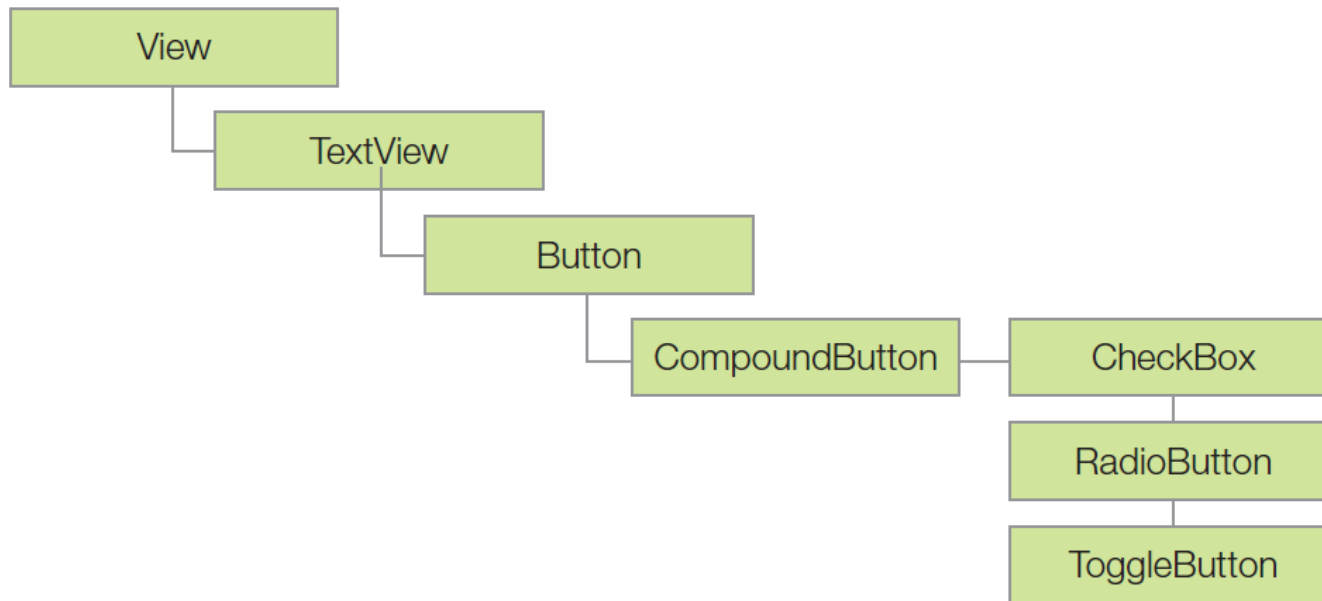


그림 3-6 텍스트뷰와 버튼 위젯들의 클래스 계층도

❖ 두 개의 CheckBox를 이용해서 텍스트의 색상이나 배경을 변경하기



(a) 초기 화면



(b) 첫 번째 CheckBox만 체크

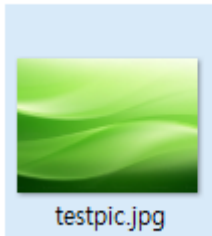


(c) 두 번째 CheckBox도 체크

그림 3-7 실행 화면

## ❖ 두 개의 CheckBox를 이용해서 텍스트의 색상이나 배경을 변경하기

| > res > drawable



res (generated)

▼ res

▼ drawable

ic\_launcher\_background.xml

ic\_launcher\_foreground.xml (v24)

testpic.jpg (v24)

▶ layout

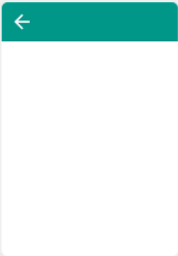
▶ mipmap

▶ values

res (generated)

Create New Project

Configure your project

 Empty Activity

Creates a new empty activity


Name  
P133\_CheckBox1

Package name  
org.lklab.p133\_checkbox1

Save location  
C:\CookAndroid\Project\WP133\_CheckBox1

Language  
Java

Minimum API level  
API 16: Android 4.1 (Jelly Bean)

 Your app will run on approximately **100%** of devices.  
[Help me choose](#)

☐ This project will support instant apps

☒ Use android.\* artifacts

Previous Next Cancel Finish



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <CheckBox
        android:id="@+id/checkColor"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="mOnClick"
        android:text="색상 교체"/>
    <CheckBox
        android:id="@+id/checkBackground"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="mOnClick"
        android:text="배경 교체"/>
    <TextView
        android:id="@+id/textTest"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center"
        android:text="테스트"
        android:textColor="#000000"
        android:textSize="40sp"
        android:textStyle="bold"/>
</LinearLayout>
```

```
package org.lklab.p133_checkbox1;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;
import android.widget.CheckBox;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    private CheckBox mCheckColor, mCheckBackground;
    private TextView mTextTest;

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

        mCheckColor = (CheckBox) findViewById(R.id.checkColor);
        mCheckBackground = (CheckBox) findViewById(R.id.checkBackground);
        mTextTest = (TextView) findViewById(R.id.textTest);
    }
}
```

```
public void mOnClick(View v) {  
    switch (v.getId()) {  
        case R.id.checkColor:  
            if (mCheckColor.isChecked())  
                mTextTest.setTextColor(0xffff0000);  
            else  
                mTextTest.setTextColor(0xff000000);  
            break;  
        case R.id.checkBackground:  
            if (mCheckBackground.isChecked())  
                mTextTest.setBackgroundResource(R.drawable.testpic);  
            else  
                mTextTest.setBackgroundResource(0);  
            break;  
        }  
    }  
}
```

❖ 라디오 그룹에 여러 개의 RadioButton을 묶어서 사용



(a) 초기 화면

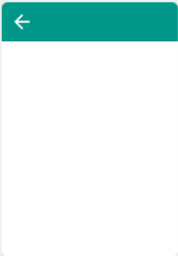


(b) '파랑'을 클릭하여 선택

그림 3-8 실행 화면

Create New Project

Configure your project

 Empty Activity

Creates a new empty activity


Name  
P137\_RadioButton1

Package name  
org.lklab.p137\_radiobutton1

Save location  
C:\CookAndroid\Project\WP137\_RadioButton1

Language  
Java

Minimum API level  
API 16: Android 4.1 (Jelly Bean)

 Your app will run on approximately **100%** of devices.  
[Help me choose](#)

☐ This project will support instant apps

☒ Use android.\* artifacts

Previous Next Cancel Finish

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/layoutMain"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <RadioGroup
        android:id="@+id/groupColor"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <RadioButton
            android:id="@+id/radioRed"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:onClick="mOnClick"
            android:text="빨강"/>
        <RadioButton
            android:id="@+id/radioGreen"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:onClick="mOnClick"
            android:text="초록"/>
        <RadioButton
            android:id="@+id/radioBlue"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:onClick="mOnClick"
            android:text="파랑"/>
    </RadioGroup>
</LinearLayout>
```

```
package org.lklab.p137_radiobutton1;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;
import android.widget.LinearLayout;
import android.widget.RadioGroup;

public class MainActivity extends AppCompatActivity {

    private LinearLayout mLayoutMain;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mLayoutMain = (LinearLayout) findViewById(R.id.layoutMain);

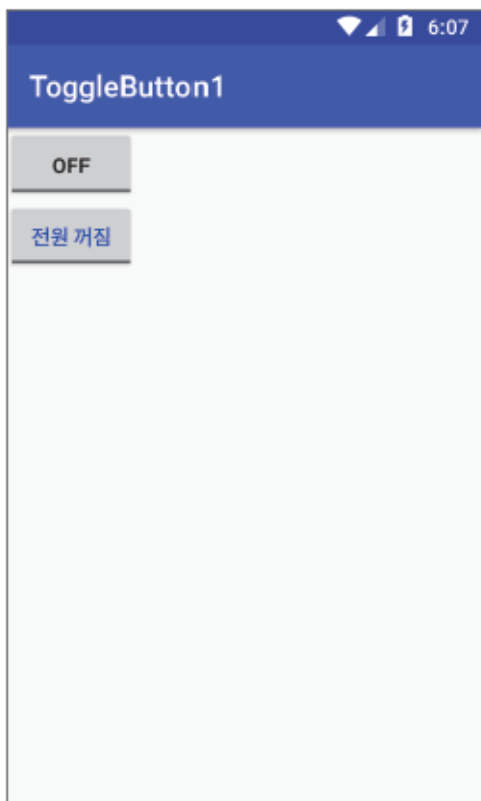
        int id = R.id.radioGreen; // 초록을 기본값으로 선택
        RadioGroup ColorGroup = (RadioGroup) findViewById(R.id.groupColor);
        ColorGroup.check(id); // 라디오 그룹에서 초록을 선택
        changeColor(id); // 초록을 클릭한 것처럼 처리
    }

    public void mOnClick(View v) {
        changeColor(v.getId());
    }
}
```

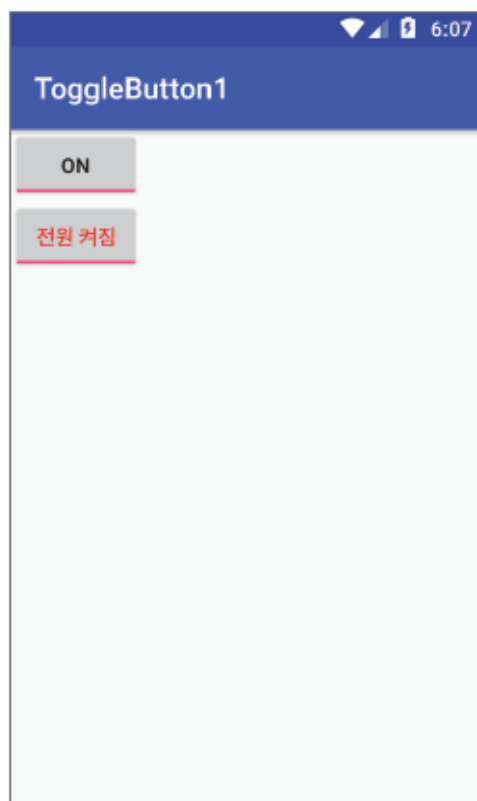
```
private void changeColor(int id) {  
    switch (id) {  
        case R.id.radioRed:  
            mLayoutMain.setBackgroundColor(0xffff0000);  
            break;  
        case R.id.radioGreen:  
            mLayoutMain.setBackgroundColor(0xff00ff00);  
            break;  
        case R.id.radioBlue:  
            mLayoutMain.setBackgroundColor(0xff0000ff);  
            break;  
    }  
}
```



❖ ToggleButton은 On/Off 두 가지 상태를 가지는 버튼



(a) 초기 화면

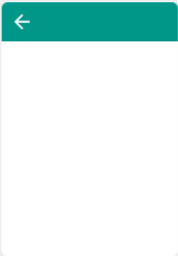


(b) 두 ToggleButton 클릭 후

그림 3-9 실행 화면

Create New Project

Configure your project

 Empty Activity

Creates a new empty activity


Name  
P139\_ToggleButton1

Package name  
org.klab.p139\_togglebutton1

Save location  
C:\CookAndroid\Project\WP139\_ToggleButton1

Language  
Java

Minimum API level  
API 16: Android 4.1 (Jelly Bean)

 Your app will run on approximately **100%** of devices.  
[Help me choose](#)

☐ This project will support instant apps

☒ Use android.\* artifacts

Previous Next Cancel Finish

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <ToggleButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
    <ToggleButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="mOnClick"
        android:textColor="#0000ff"
        android:textOff="전원 꺼짐"
        android:textOn="전원 켜짐"/>
</LinearLayout>
```

```
package org.lklab.p139_togglebutton1;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;
import android.widget.ToggleButton;

public class MainActivity extends AppCompatActivity {

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

    public void mOnClick(View v) {
        ToggleButton btn = (ToggleButton) v;
        if (btn.isChecked())
            btn.setTextColor(0xffff0000);
        else
            btn.setTextColor(0xff0000ff);
    }
}
```

- ❖ 다양한 형식의 이미지를 화면에 출력하는 위젯
- ❖ `ImageView` 클래스: 이미지 출력에 필요한 공통의 속성과 기능을 모아놓은 것
- ❖ `ImageButton` 클래스: 이미지가 버튼처럼 동작하는 기능을 추가한 것

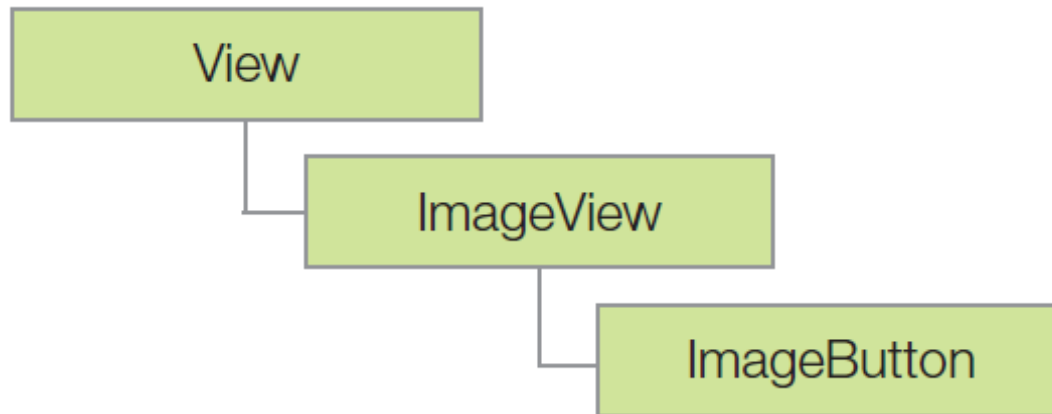


그림 3-10 이미지뷰 위젯들의 클래스 계층도

## ❖ ImageView

- 리소스, 로컬 파일, Bitmap, Drawable을 출력할 수 있는 다재다능한 클래스이다.
- ❖ 레이아웃 디자인에 자주 사용되는 순서의 주요 속성

표 3-4 ImageView 속성

| XML 속성                   | 관련 메서드                | 기능  |
|--------------------------|-----------------------|---|
| android:src              | setImageResource()    | ImageView에 출력할 리소스를 지정한다.   |
| android:scaleType        | setScaleType()        | ImageView에 출력할 이미지의 크기와 출력 위치를 제어한다. 상세 사항은 아래에서 추가 설명한다.   |
| android:adjustViewBounds | setAdjustViewBounds() | 속성값이 "true"이면 이미지의 종횡비에 맞춰 ImageView의 크기가 자동으로 조정된다. 예를 들어 이미지가 3:2 비율이면 이미지뷰도 3:2 비율이 된다. 기본값은 "false"이다.                                      |
| android:maxWidth         | setMaxWidth()         | ImageView의 최대 폭을 지정한다.  |
| android:maxHeight        | setMaxHeight()        | ImageView의 최대 높이를 지정한다.   |
| android:cropToPadding    | setCropToPadding()    | 속성값이 "true"이면 이미지의 출력이 항상 패딩 영역 내부로 제한된다. 속성값이 기본값인 "false"여도 이미지의 출력이 패딩 영역 내부로 여전히 제한되는데, 이미지를 원본 크기(1:1)로 출력할 경우에 한해 패딩 영역 내부로 제한되지 않고 출력된다. |

### ❖ 이미지뷰가 액티비티 화면 전체를 차지

activity\_main.xml

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <ImageView xmlns:android="http://schemas.android.com/apk/res/android"
3      android:layout_width="match_parent"
4      android:layout_height="match_parent"
5      android:scaleType="속성값"
6      android:src="@drawable/testpic"/>
```

## ❖ scaleType 속성값에 따른 ScaleType 예제의 실행 화면



(a) matrix



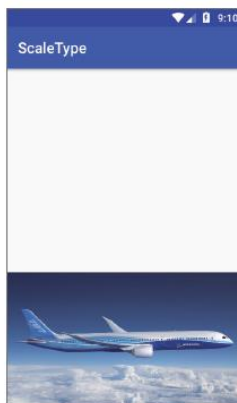
(b) fitXY



(c) fitStart



(d) fitCenter



(e) fitEnd



(f) center



(g) centerCrop

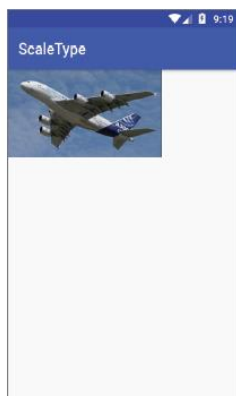


(h) centerInside

그림 3-11 scaleType 속성값에 따른 이미지 출력 결과(이미지가 큰 경우)



## ❖ scaleType 속성값에 따른 ScaleType 예제의 실행 화면



(a) matrix



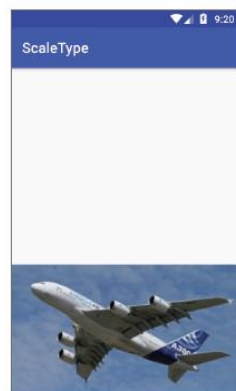
(b) fitXY



(c) fitStart



(d) fitCenter



(e) fitEnd



(f) center



(g) centerCrop



(h) centerInside

그림 3-12 scaleType 속성값에 따른 이미지 출력 결과(이미지가 작은 경우)

## ❖ scaleType 속성값

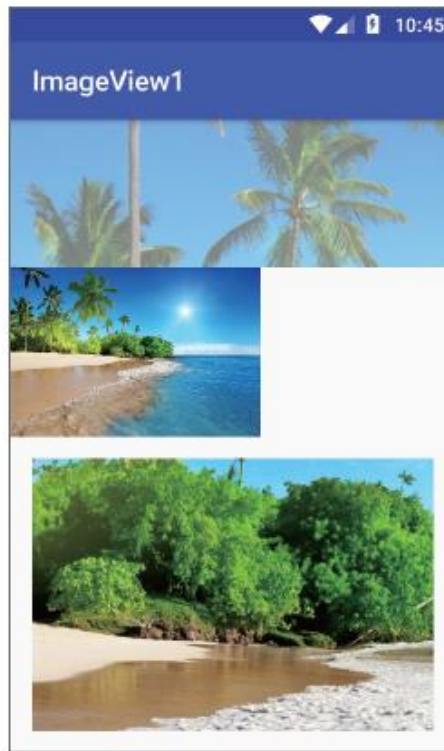
표 3-5 scaleType 속성값에 따른 이미지 출력 방식

| scaleType 속성값 | 기능  |
|---------------|---|
| matrix        | 뷰의 좌상단을 기준으로 출력하되 setImageMatrix() 메서드로 행렬(matrix)을 조작하지 않으면 1:1, 즉 원본 그대로 출력한다.  |
| fitXY         | 이미지의 종횡비를 유지하지 않으면서 뷰 화면을 가득 채운다.   |
| fitStart      | 뷰의 좌상단을 기준으로 출력하되 이미지의 종횡비를 유지하면서 가득 채운다.   |
| fitCenter     | 뷰의 정중앙을 기준으로 출력하되 이미지의 종횡비를 유지하면서 가득 채운다.   |
| fitEnd        | 뷰의 우하단을 기준으로 출력하되 이미지의 종횡비를 유지하면서 가득 채운다.   |
| center        | 이미지가 뷰의 정중앙에 위치하되 1:1, 즉 원본 그대로 출력한다.   |
| centerCrop    | 이미지가 뷰의 정중앙에 위치하되 이미지의 폭과 높이가 뷰의 폭과 높이보다 크거나 같도록 (필요시) 크기를 조정한다. 이때 이미지의 종횡비는 유지한다.<br><b>참고</b> 이미지가 뷰보다 크면 어느 한쪽 면이 뷰에 닿을 때까지 이미지를 줄이고, 이미지가 뷰보다 작으면 어느 한쪽 면이 뷰에 닿을 때까지 이미지를 키운다. |
| centerInside  | 이미지가 뷰의 정중앙에 위치하되 이미지의 폭과 높이가 뷰의 폭과 높이보다 작거나 같도록 (필요시) 크기를 조정한다. 이때 이미지의 종횡비는 유지한다.<br><b>참고</b> 이미지가 뷰보다 크면 모든 면이 뷰의 내부에 들어올 때까지 이미지를 줄이고, 이미지가 뷰보다 작으면 그대로 둔다.                    |

## ❖ 실행 화면



(a) 초기 화면



(b) 아래로 스크롤한 후

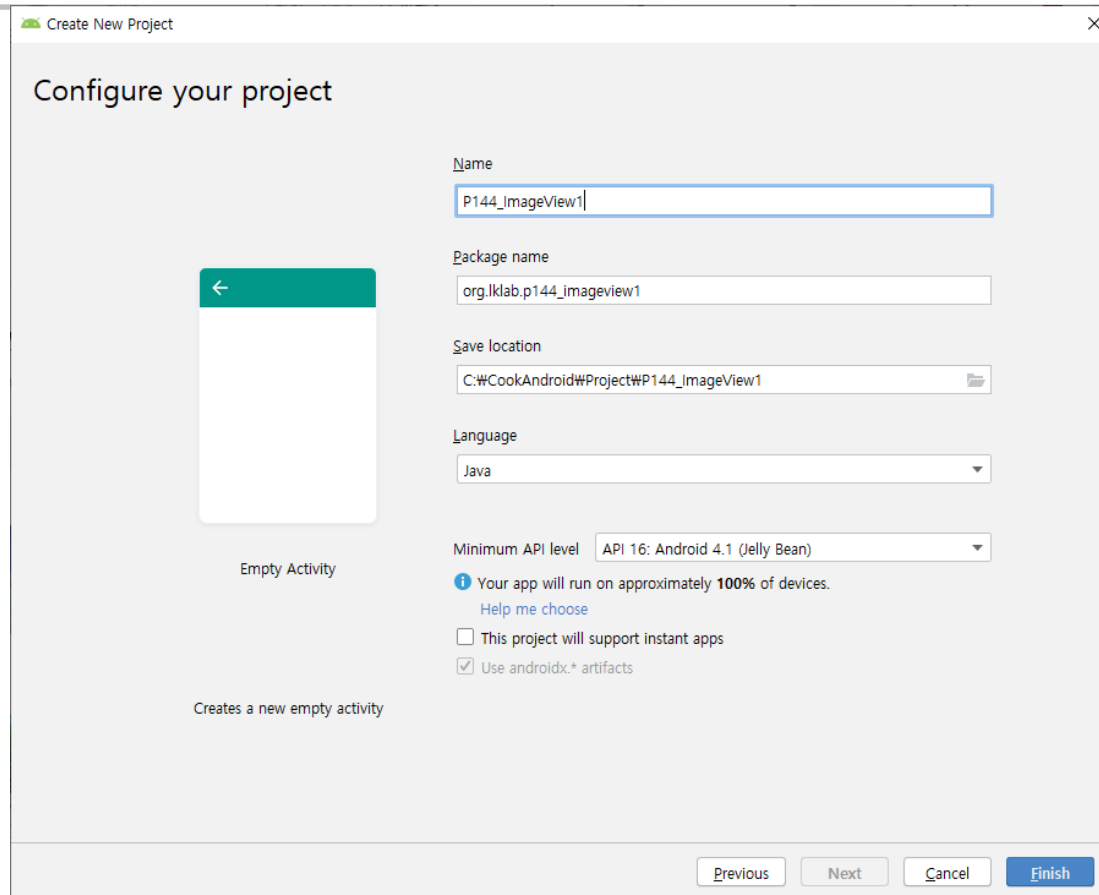
그림 3-13 실행 화면

## ❖ 실행 화면

main &gt; res &gt; drawable-nodpi



- ▼ res
  - ▼ drawable
    - ic\_launcher\_background.xml
    - ic\_launcher\_foreground.xml (v24)
    - testpic.jpg
  - ▶ layout
  - ▶ mipmap



```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">
        <ImageView
            android:id="@+id/imageTest"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:adjustViewBounds="true"
            android:scaleType="matrix"
            android:src="@drawable/testpic"/>
        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:adjustViewBounds="true"
            android:maxWidth="180dp"
            android:scaleType="fitXY"
            android:src="@drawable/testpic"/>
        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:adjustViewBounds="true"
            android:cropToPadding="true"
            android:padding="16dp"
            android:scaleType="center"
            android:src="@drawable/testpic"/>
    </LinearLayout>
</ScrollView>
```

```
package org.lklab.p144_imageview1;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {

    private ImageView mImageTest;

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

        mImageTest = (ImageView) findViewById(R.id.imageTest);
        mImageTest.setAlpha(128); /* 안드로이드 1.0부터 사용 가능 */
        //mImageTest.setImageAlpha(128); /* 안드로이드 4.1부터 사용 가능 */
    }
}
```

- ❖ 버튼처럼 누르고 떼는 효과를 ImageView에 추가한 것
- ❖ 추가 속성은 없으며 ImageView처럼 src 속성으로 출력할 대상을 지정

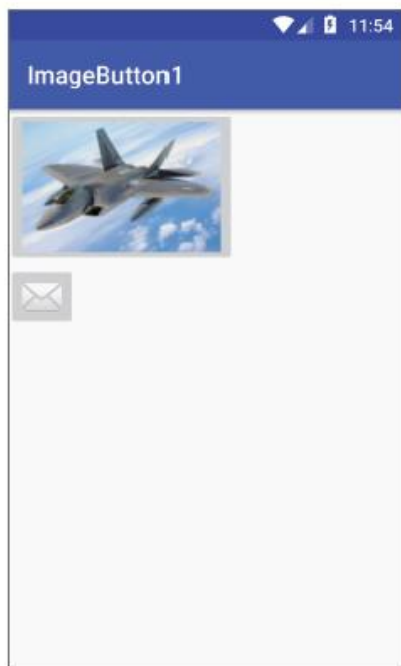
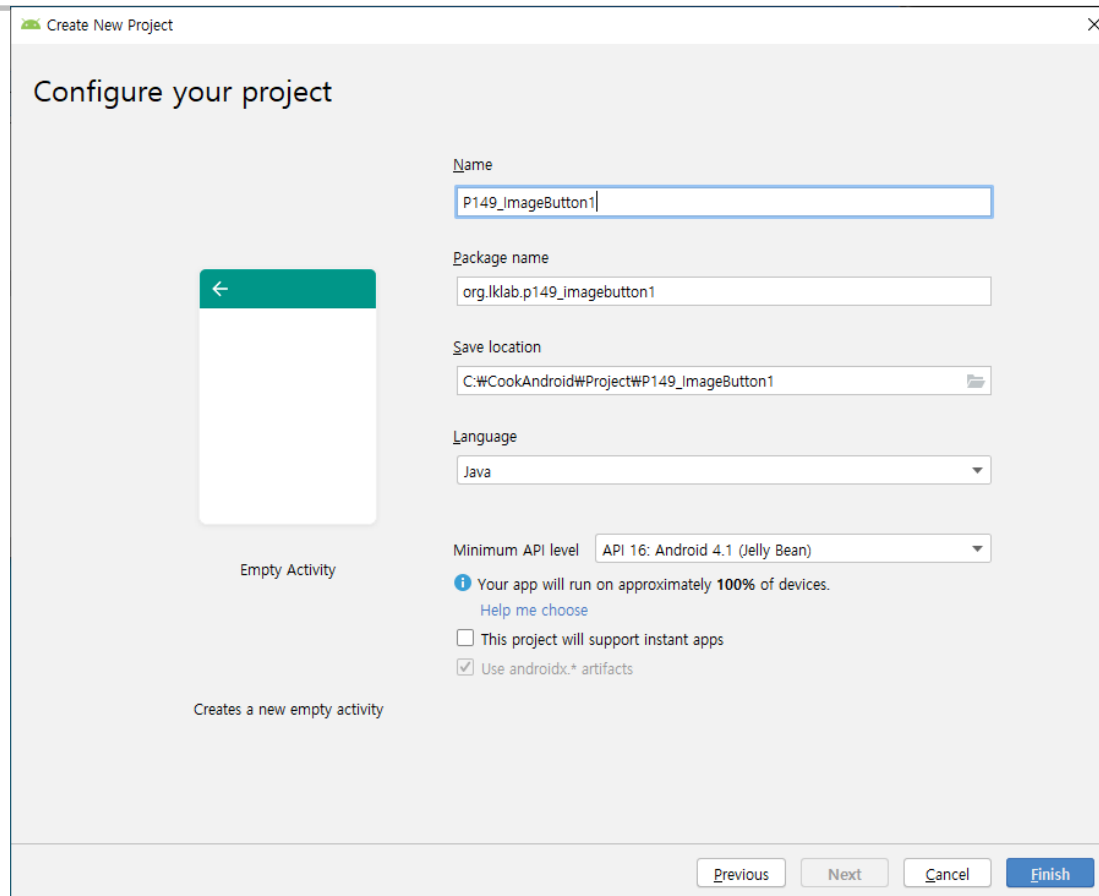


그림 3-15 실행 화면







```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <ImageButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/testpic"/>
    <ImageButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@android:drawable/ic_dialog_email"/>
</LinearLayout>
```

```
package org.lklab.p149_imagebutton1;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

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

- ❖ 작업의 진행 상태를 표시하는 것이 주 기능
- ❖ 대표 클래스인 ProgressBar는 주로 출력용으로 사용
- ❖ 추상 클래스인 AbsSeekBar에서 파생된 RatingBar와 SeekBar 클래스는 주로 입력용

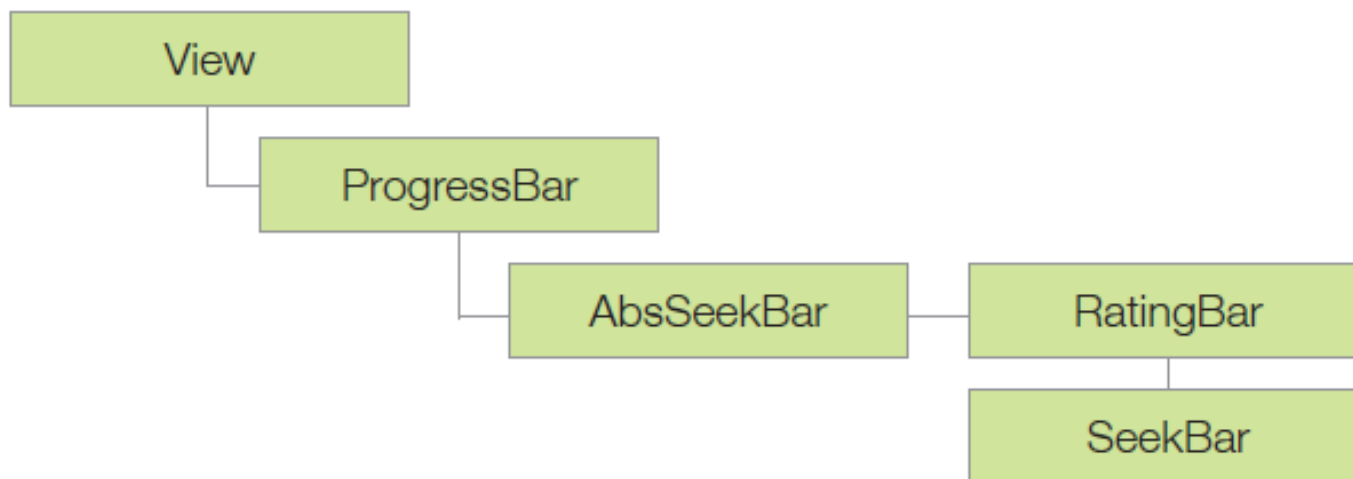


그림 3-16 프로그레스바 위젯들의 클래스 계층도

- ❖ 작업의 진행 상황을 회전 바퀴나 수평 막대로 보여줌
- ❖ 확정(indeterminate) 모드와 불확정(indeterminate) 모드가 있음
- ❖ 불확정 모드에서는 진행값을 보여주지 않고 애니메이션으로만 표시
- ❖ 수평 막대는 확정 모드에서 최대 두 개의 진행값을 보여줌
- ❖ 각각 기본 진행값(default progress value)과 2차 진행값(secondary progress value)이라 부름



(a) 회전 바퀴: 확정/불확정 모드 동일



(b) 수평 막대: 확정 모드

(기본 진행값을 50%, 2차 진행값을 75%로 설정한 예)



(c) 수평 막대: 불확정 모드

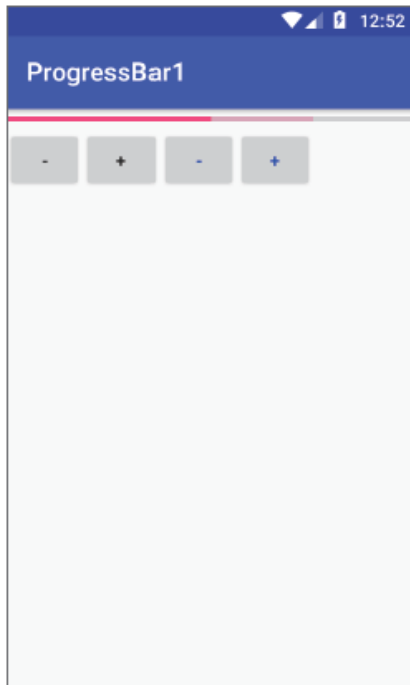
그림 3-17 ProgressBar 형태

## ❖ ProgressBar의 주요 속성

표 3-6 ProgressBar 속성

| XML 속성                    | 관련 메서드                 | 기능   |
|---------------------------|------------------------|--|
| android:max               | setMax()               | 최대 진행값을 지정한다. 예를 들어 값이 "255"면 진행값은 0~255 범위가 된다. |
| android:progress          | setProgress()          | 기본 진행값을 지정한다.                                    |
| android:secondaryProgress | setSecondaryProgress() | 2차 진행값을 지정한다.                                    |
| android:indeterminate     | setIndeterminate()     | 값이 "true"이면 불확정 모드가 된다.                          |

- ❖ ProgressBar를 확정 모드로 사용하면서 진행값을 변경하는 방법
  - 좌측의 [-/+ ] 버튼은 기본 진행값을 증감
  - 우측의 [-/+ ] 버튼은 2 차 진행값을 증감



activity\_main.xml

```

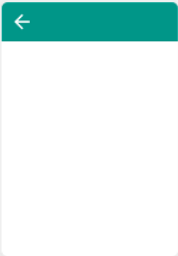
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:layout_width="match_parent"
4      android:layout_height="match_parent"
5      android:orientation="vertical">
6      <ProgressBar
7          android:id="@+id/progStatus"

```

그림 3-18 실행 화면

Create New Project

Configure your project

 Empty Activity

Creates a new empty activity


Name  
P152\_ProgressBar1

Package name  
org.lklab.p152\_progressbar1

Save location  
C:\CookAndroid\Project\WP152\_ProgressBar1

Language  
Java

Minimum API level  
API 16: Android 4.1 (Jelly Bean)

 Your app will run on approximately **100%** of devices.  
[Help me choose](#)

☐ This project will support instant apps

☒ Use android.\* artifacts

Previous Next Cancel Finish



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <ProgressBar
        android:id="@+id/progStatus"
        style="?android:attr/progressBarStyleHorizontal"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:max="120"
        android:progress="60"
        android:secondaryProgress="90"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <Button
            android:id="@+id/btnDecrease1"
            android:layout_width="60dp"
            android:layout_height="wrap_content"
            android:onClick="mOnClick"
            android:text="-"/>
        <Button
            android:id="@+id/btnIncrease1"
            android:layout_width="60dp"
            android:layout_height="wrap_content"
            android:onClick="mOnClick"
            android:text="+"/>
    </LinearLayout>
</LinearLayout>
```

```
<Button
    android:id="@+id/btnDecrease2"
    android:layout_width="60dp"
    android:layout_height="wrap_content"
    android:onClick="mOnClick"
    android:text="-"
    android:textColor="#0000ff"/>
<Button
    android:id="@+id/btnIncrease2"
    android:layout_width="60dp"
    android:layout_height="wrap_content"
    android:onClick="mOnClick"
    android:text="+"
    android:textColor="#0000ff"/>
</LinearLayout>
</LinearLayout>
```

```
package org.lklab.p152_progressbar1;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;
import android.widget.ProgressBar;

public class MainActivity extends AppCompatActivity {

    private ProgressBar mProgStatus;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mProgStatus = (ProgressBar) findViewById(R.id.progStatus);
    }

    public void mOnClick(View v) {
        switch (v.getId()) {
            case R.id.btnDecrease1:
                mProgStatus.setProgress(mProgStatus.getProgress() - 5);
                break;
            case R.id.btnIncrease1:
                mProgStatus.setProgress(mProgStatus.getProgress() + 5);
                break;
            case R.id.btnDecrease2:
                mProgStatus.setSecondaryProgress(mProgStatus.getSecondaryProgress() - 5);
                break;
            case R.id.btnIncrease2:
                mProgStatus.setSecondaryProgress(mProgStatus.getSecondaryProgress() + 5);
                break;
        }
    }
}
```

- ❖ RatingBar는 별의 개수로 평점을 입력받을 때 사용
- ❖ 주요 속성

표 3-7 RatingBar 속성

| XML 속성              | 관련 메서드           | 기능   |
|---------------------|------------------|--|
| android:numStars    | setNumStars()    | 표시할 별의 개수를 지정한다.   |
| android:rating      | setRating()      | 기본 평점을 지정한다.   |
| android:stepSize    | setStepSize()    | 스텝 크기를 지정한다. 예를 들어 값이 "0.5"이면 한 단계 값을 증감할 때 별 1/2만큼 평점이 증감한다.        |
| android:isIndicator | setIsIndicator() | "true"이면 사용자 입력을 받을 수 없고 출력만 한다.<br>기본값은 "false"이므로 사용자 입력을 받을 수 있다. |

## ❖ 실행 화면

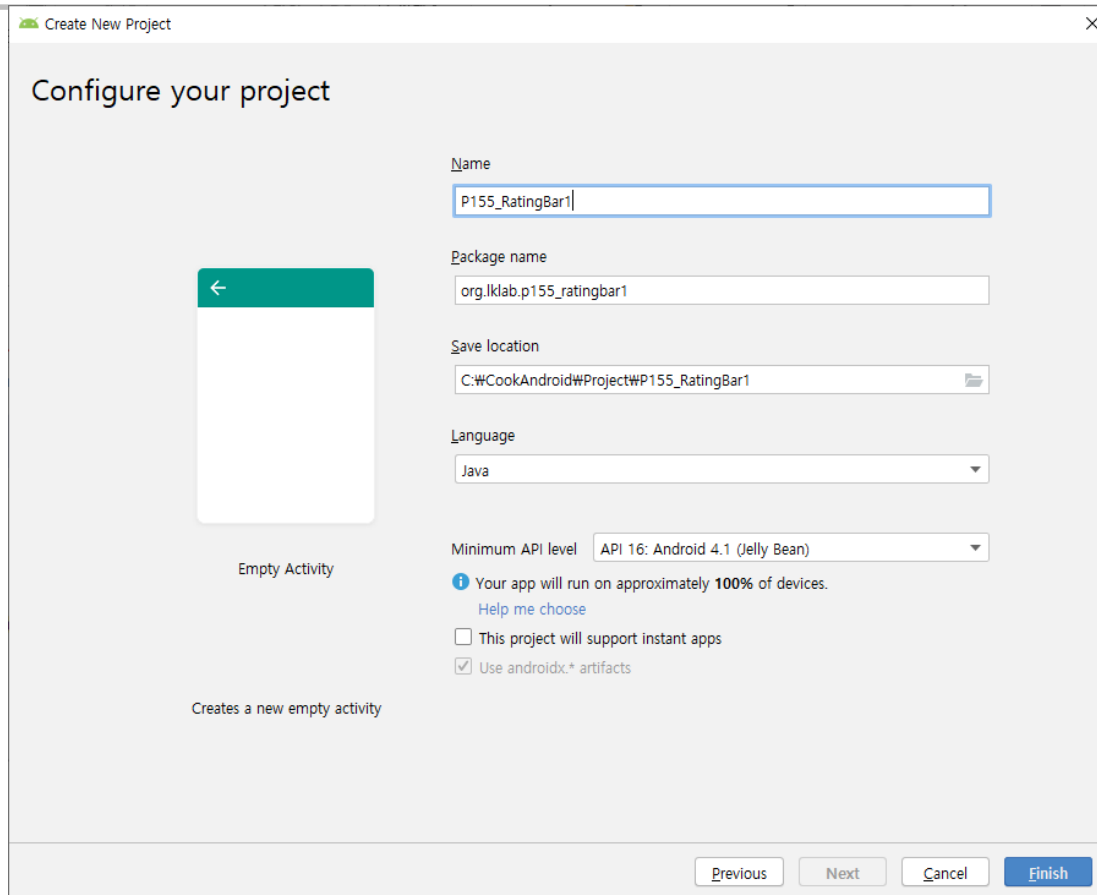


(a) 초기 화면



(b) 평점 1/2 증가 후 [투표 결과는?] 클릭

그림 3-19 실행 화면



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <RatingBar
        android:id="@+id/ratingVote"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:numStars="5"
        android:rating="3"
        android:stepSize="0.5"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <Button
            android:id="@+id/btnDec"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:onClick="mOnClick"
            android:text="-"/>
        <Button
            android:id="@+id/btnInc"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:onClick="mOnClick"
            android:text="+"/>
        <Button
            android:id="@+id/btnResult"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:onClick="mOnClick"
            android:text="투표 결과는?"/>
    </LinearLayout>
</LinearLayout>
```

```
package org.lklab.p155_ratingbar1;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;
import android.widget.RatingBar;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    private RatingBar mRatingVote;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mRatingVote = (RatingBar) findViewById(R.id.ratingVote);
    }

    public void mOnClick(View v) {
        switch (v.getId()) {
            case R.id.btnDec:
                mRatingVote.incrementProgressBy(-1);
                break;
            case R.id.btnInc:
                mRatingVote.incrementProgressBy(1);
                break;
            case R.id.btnResult:
                Toast.makeText(this, "현재값 = " + mRatingVote.getRating(),
                    Toast.LENGTH_SHORT).show();
                break;
        }
    }
}
```



### ❖ SeekBar

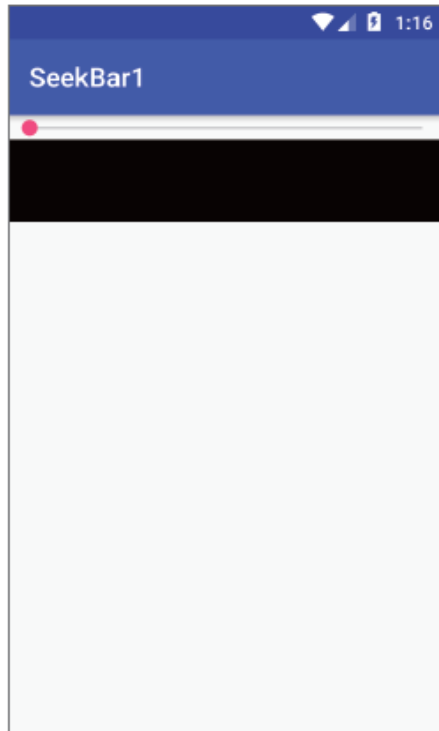
- 프로그레스바의 일종
- 썸(thumb)을 이용해 사용자가 진행값을 조작
- 모양이나 동작이 윈도우즈 운영 체제의 스크롤바(scroll bar)와 거의 같음
- 썸을 조작하면 이벤트가 발생하는데 SeekBar.OnSeekBarChangeListener 인터페이스 객체를 통해 처리

### ❖ SeekBar.OnSeekBarChangeListener 인터페이스에 포함된 추상 메서드

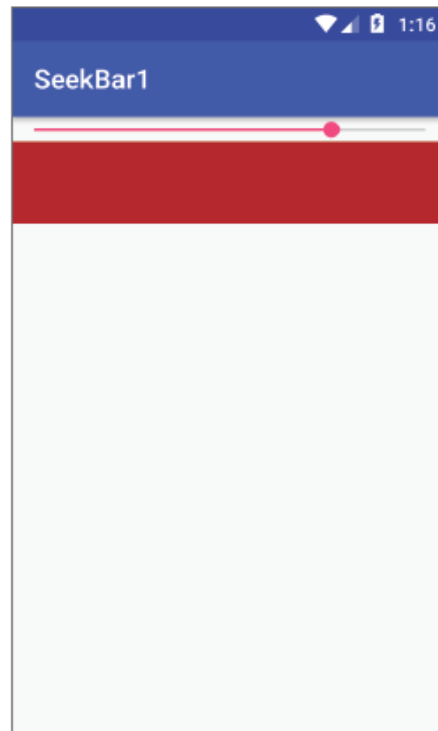
표 3-8 SeekBar.OnSeekBarChangeListener 인터페이스

| 추상 메서드  | 의미  |
|---|---|
| void onStartTrackingTouch(SeekBar seekBar)                              | 사용자가 터치하여 썸을 끌기를 시작했다.  |
| void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) | 썸의 위치가 변경되었다. 현재 위치값은 progress이며 사용자의 행위가 아닌 자바 코드로 위치값을 변경한 경우에는 fromUser의 값이 "false"가 된다. |
| void onStopTrackingTouch(SeekBar seekBar)                               | 사용자가 손을 떼어 썸을 끌기를 중단했다.   |

## ❖ 실행 화면



(a) 초기 화면

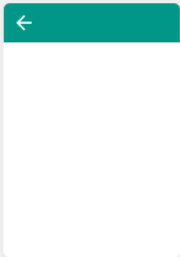


(b) 썸 조작 후

그림 3-20 실행 화면

Create New Project

Configure your project

 Empty Activity

Creates a new empty activity


Name  
P159\_SeekBar1

Package name  
org.lklab.p159\_seekbar1

Save location  
C:\CookAndroid\ProjectWP159\_SeekBar1

Language  
Java

Minimum API level  
API 16: Android 4.1 (Jelly Bean)

 Your app will run on approximately **100%** of devices.  
[Help me choose](#)

☐ This project will support instant apps

☒ Use android.\* artifacts

Previous Next Cancel Finish

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">
    <SeekBar
        android:id="@+id/seekRed"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:max="255"/>
    <View
        android:id="@+id/viewColor"
        android:layout_width="match_parent"
        android:layout_height="60dp"/>
</LinearLayout>
```

```
package org.lklab.p159_seekbar1;

import androidx.appcompat.app.AppCompatActivity;

import android.graphics.Color;
import android.os.Bundle;

import android.view.View;
import android.widget.SeekBar;

public class MainActivity extends AppCompatActivity {

    private SeekBar mSeekBar;
    private View mViewColor;
```

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

    mSeekBar = (SeekBar) findViewById(R.id.seekBar);
    mViewColor = findViewById(R.id.viewColor);

    mViewColor.setBackgroundColor(Color.rgb(255, mSeekBar.getProgress(), 0, 0));

    mSeekBar.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {

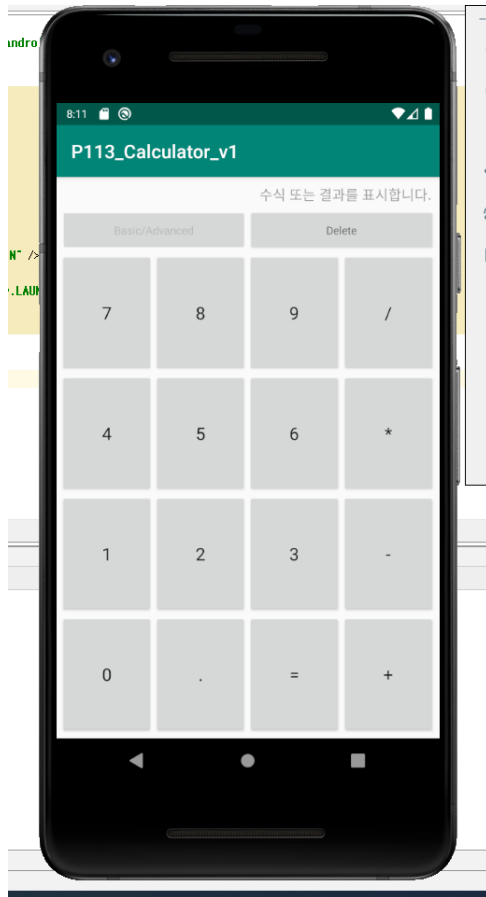
        @Override
        public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
            if (fromUser) {
                mViewColor.setBackgroundColor(Color.rgb(255, progress, 0, 0));
            }
        }

        @Override
        public void onStartTrackingTouch(SeekBar seekBar) {
        }

        @Override
        public void onStopTrackingTouch(SeekBar seekBar) {
        }
    });
}
```



## ❖ 계산기





Create New Project

Configure your project

Name  
P164\_Calculator\_v2

Package name  
org.lklab.p164\_calculator\_v2

Save location  
C:\CookAndroid\Project\WP164\_Calculator\_v2

Language  
Java

Minimum API level  
API 16: Android 4.1 (Jelly Bean)

Empty Activity

Creates a new empty activity

☐ This project will support instant apps

☒ Use android.\* artifacts

[Help me choose](#)

Previous Next Cancel Finish

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="4dp">
    <TextView
        android:id="@+id/textDisplay"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ellipsize="start"
        android:gravity="right | center_vertical"
        android:hint="수식 또는 결과를 표시합니다."
        android:padding="4dp"
        android:singleLine="true"
        android:textAppearance="?android:attr/textAppearanceMedium"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <CheckBox
            android:id="@+id/btnMode"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="2"
            android:checked="false"
            android:onClick="mOnClick"
            android:text="Advanced"
            android:textAppearance="?android:attr/textAppearanceSmall"/>
```

```
<Button
```

```
    android:id="@+id/btnPer"  
    android:layout_width="0dp"  
    android:layout_height="wrap_content"  
    android:layout_weight="0.8"  
    android:onClick="mOnClick"  
    android:text="%"  
    android:textAppearance="?android:attr/textAppearanceSmall"  
    android:visibility="invisible"/>
```

```
<Button
```

```
    android:id="@+id/btnPow"  
    android:layout_width="0dp"  
    android:layout_height="wrap_content"  
    android:layout_weight="0.8"  
    android:onClick="mOnClick"  
    android:text="^"  
    android:textAppearance="?android:attr/textAppearanceSmall"  
    android:visibility="invisible"/>
```

```
<Button
```

```
    android:id="@+id/btnDel"  
    android:layout_width="0dp"  
    android:layout_height="wrap_content"  
    android:layout_weight="2"  
    android:onClick="mOnClick"  
    android:text="Delete"  
    android:textAppearance="?android:attr/textAppearanceSmall"/>
```

```
</LinearLayout>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1">
    <Button
        android:id="@+id/btn7"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="7"
        android:textAppearance="?android:attr/textAppearanceLarge"/>
    <Button
        android:id="@+id/btn8"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="8"
        android:textAppearance="?android:attr/textAppearanceLarge"/>
    <Button
        android:id="@+id/btn9"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="9"
        android:textAppearance="?android:attr/textAppearanceLarge"/>
    <Button
        android:id="@+id/btnDiv"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="/"
        android:textAppearance="?android:attr/textAppearanceLarge"/>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1">
    <Button
        android:id="@+id/btn4"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="4"
        android:textAppearance="?android:attr/textAppearanceLarge"/>
    <Button
        android:id="@+id/btn5"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="5"
        android:textAppearance="?android:attr/textAppearanceLarge"/>
    <Button
        android:id="@+id/btn6"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="6"
        android:textAppearance="?android:attr/textAppearanceLarge"/>
    <Button
        android:id="@+id/btnMul"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="*"
        android:textAppearance="?android:attr/textAppearanceLarge"/>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1">
    <Button
        android:id="@+id/btn1"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="1"
        android:textAppearance="?android:attr/textAppearanceLarge"/>
    <Button
        android:id="@+id/btn2"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="2"
        android:textAppearance="?android:attr/textAppearanceLarge"/>
    <Button
        android:id="@+id/btn3"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="3"
        android:textAppearance="?android:attr/textAppearanceLarge"/>
    <Button
        android:id="@+id/btnSub"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="-"
        android:textAppearance="?android:attr/textAppearanceLarge"/>
</LinearLayout>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1">
    <Button
        android:id="@+id/btn0"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="0"
        android:textAppearance="?android:attr/textAppearanceLarge"/>
    <Button
        android:id="@+id/btnDot"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="."
        android:textAppearance="?android:attr/textAppearanceLarge"/>
    <Button
        android:id="@+id/btnEq1"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="="
        android:textAppearance="?android:attr/textAppearanceLarge"/>
    <Button
        android:id="@+id/btnAdd"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:onClick="mOnClick"
        android:text="+"
        android:textAppearance="?android:attr/textAppearanceLarge"/>
```

```
package org.lklab.p164_calculator_v2;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    private TextView mTextDisplay;
    private CheckBox mBtnMode;
    private Button mBtnPer;
    private Button mBtnPow;
    private StringBuilder mMathExpr;

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

        mTextDisplay = (TextView) findViewById(R.id.textDisplay);
        mBtnMode = (CheckBox) findViewById(R.id.btnMode);
        mBtnPer = (Button) findViewById(R.id.btnPer);
        mBtnPow = (Button) findViewById(R.id.btnPow);

        mMathExpr = new StringBuilder(256);
```



```
Button BtnDel = (Button) findViewById(R.id.btnDel);
BtnDel.setOnLongClickListener(new View.OnLongClickListener() {
    @Override
    public boolean onLongClick(View v) {
        mMathExpr.delete(0, mMathExpr.length());
        mTextDisplay.setText(mMathExpr);
        return true;
    }
});


public void mOnClick(View v) {
    switch (v.getId()) {
        case R.id.btnMode:
            if (mBtnMode.isChecked()) {
                mBtnPer.setVisibility(View.VISIBLE);
                mBtnPow.setVisibility(View.VISIBLE);
            } else {
                mBtnPer.setVisibility(View.INVISIBLE);
                mBtnPow.setVisibility(View.INVISIBLE);
            }
            break;
        case R.id.btn0:
            mMathExpr.append("0");
            break;
        case R.id.btn1:
            mMathExpr.append("1");
            break;
        case R.id.btn2:
            mMathExpr.append("2");
            break;
    }
}
```

```
case R.id.btn3:
    mMathExpr.append("3");
    break;
case R.id.btn4:
    mMathExpr.append("4");
    break;
case R.id.btn5:
    mMathExpr.append("5");
    break;
case R.id.btn6:
    mMathExpr.append("6");
    break;
case R.id.btn7:
    mMathExpr.append("7");
    break;
case R.id.btn8:
    mMathExpr.append("8");
    break;
case R.id.btn9:
    mMathExpr.append("9");
    break;
case R.id.btnDot:
    mMathExpr.append(".");
    break;
case R.id.btnAdd:
    mMathExpr.append("+");
    break;
case R.id.btnSub:
    mMathExpr.append("-");
    break;
```

```
case R.id.btnMul:
    mMathExpr.append("*");
    break;
case R.id.btnDiv:
    mMathExpr.append("/");
    break;
case R.id.btnPer:
    mMathExpr.append("%");
    break;
case R.id.btnPow:
    mMathExpr.append("^");
    break;
case R.id.btnDel:
    if (mMathExpr.length() > 0)
        mMathExpr.deleteCharAt(mMathExpr.length() - 1);
    break;
case R.id.btnEq:
    if (mMathExpr.length() == 0)
        return;
    // 중위 표기법 → 후위 표기법 → 스택으로 계산 → 결과 리턴
    String result = NumCalc.calc(mMathExpr.toString());
    if (result != null) {
        mMathExpr.delete(0, mMathExpr.length());
        mMathExpr.append(result);
    }
    break;
}
mTextDisplay.setText(mMathExpr);
}
```

## ❖ NumCalc.java

New Android Activity

 **Configure Activity**  
Android Studio

**Creates a new empty activity**

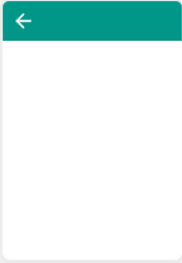
Activity Name:

☐ Generate Layout File

☐ Launcher Activity

Package name:

Source Language:



If true, a layout file will be generated

```
package org.lklab.p164_calculator_v2;

import java.util.ArrayList;
import java.util.Stack;

public class NumCalc {

    private static final int NUM = 0; // 숫자
    private static final int LEFT = 1; // 왼쪽 괄호
    private static final int RIGHT = 2; // 오른쪽 괄호
    private static final int OP = 3; // 연산자

    private static int getType(String expr) {
        if (expr.equals("("))
            return LEFT;
        if (expr.equals(")")
            return RIGHT;
        if (expr.equals("+"))
            return OP;
        if (expr.equals("-"))
            return OP;
        if (expr.equals("*"))
            return OP;
        if (expr.equals("/"))
            return OP;
        if (expr.equals("%"))
            return OP;
        if (expr.equals("^"))
            return OP;
        return NUM;
    }
}
```

```
private static ArrayList<String> parse(String expr) {
    if (getType(expr.substring(expr.length() - 1)) == OP)
        return null;

    ArrayList<String> result = new ArrayList<String>();
    StringBuilder buf = new StringBuilder(32);

    for (int i = 0; i < expr.length(); i++) {
        char ch = expr.charAt(i);
        if ((ch >= '0' && ch <= '9') || ch == '.' || ch == 'E') {
            buf.append(ch);
        } else if (ch == '(' || ch == ')') {
            if (buf.length() > 0) {
                result.add(buf.toString());
                buf.delete(0, buf.length());
            }
            buf.append(ch);
            result.add(buf.toString());
            buf.delete(0, buf.length());
        } else if (ch == '-') {
            if (i == 0) {
                buf.append(ch);
            } else {
                char c = expr.charAt(i - 1);
                if (c == '(' || c == '*' || c == '/' || c == '%' || c == '^' || c == 'E')
                    buf.append(ch);
                else {
                    if (buf.length() > 0) {
                        result.add(buf.toString());
                        buf.delete(0, buf.length());
                    }
                    buf.append(ch);
                    result.add(buf.toString());
                    buf.delete(0, buf.length());
                }
            }
        }
    }
}
```

```
} else if (ch == '+' || ch == '*' || ch == '/' || ch == '%' || ch == '^') {  
    if (buf.length() > 0) {  
        result.add(buf.toString());  
        buf.delete(0, buf.length());  
    }  
    buf.append(ch);  
    result.add(buf.toString());  
    buf.delete(0, buf.length());  
}  
}  
result.add(buf.toString());  
return result;  
}
```

```
private static int getPrecedence(String op) {  
    if (op.equals("+"))  
        return 1;  
    else if (op.equals("-"))  
        return 1;  
    else if (op.equals("*"))  
        return 2;  
    else if (op.equals("/"))  
        return 2;  
    else if (op.equals("%"))  
        return 3;  
    else if (op.equals("^"))  
        return 3;  
    return 0;  
}
```

```
private static ArrayList<String> postfix(ArrayList<String> expr) {
    ArrayList<String> result = new ArrayList<String>();
    Stack<String> stack = new Stack<String>();

    for (String str : expr) {
        if (getType(str) == NUM)
            result.add(str);
        else if (getType(str) == LEFT) {
            stack.push(str);
        } else if (getType(str) == OP) {
            if (stack.isEmpty())
                stack.push(str);
            else {
                while (!stack.isEmpty()) {
                    if (getPrecedence(stack.lastElement()) >= getPrecedence(str))
                        result.add(stack.pop());
                    else
                        break;
                }
                stack.push(str);
            }
        } else if (getType(str) == RIGHT) {
            while (!stack.isEmpty() && (getType(stack.lastElement()) != LEFT)) {
                result.add(stack.pop());
            }
            stack.pop();
        }
    }
    while (!stack.isEmpty()) {
        result.add(stack.pop());
    }
    return result;
}
```



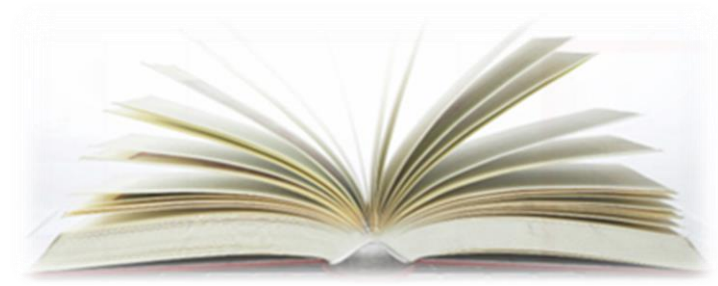
```
private static String calcRPN(ArrayList<String> expr) {
    Stack<Double> stack = new Stack<Double>();
    double n1, n2, result;

    for (String str : expr) {
        if (getType(str) == NUM) {
            stack.push(Double.parseDouble(str));
        } else if (getType(str) == OP) {
            n2 = stack.pop();
            n1 = stack.pop();
            if (str.equals("+")) {
                result = n1 + n2;
            } else if (str.equals("-")) {
                result = n1 - n2;
            } else if (str.equals("*")) {
                result = n1 * n2;
            } else if (str.equals("/")) {
                if (n2 == 0.0)
                    return null;
                else
                    result = n1 / n2;
            } else if (str.equals("%")) {
                result = n1 * n2 / 100.0;
            } else if (str.equals("^")) {
                result = Math.pow(n1, n2);
            } else {
                return null;
            }
            stack.push(result);
        }
    }
    return stack.pop().toString();
}
```

```
public static String calc(String expr) {  
    ArrayList<String> parsed_expr;  
    String result = null;  
  
    parsed_expr = NumCalc.parse(expr);  
    if (parsed_expr != null) {  
        result = NumCalc.calcRPN(NumCalc.postfix(parsed_expr));  
        if (result != null) {  
            if (result.endsWith(".0")) {  
                result = result.substring(0, result.length() - 2);  
            }  
            if (result.endsWith("-0")) {  
                result = result.replace("-0", "0");  
            }  
        }  
    }  
    return result;  
}
```

# Unit A

---



## 참고자료

- ❖ <http://www.ncs.go.kr>
- ❖ IT CookBook, 단계별로 배우는 안드로이드 프로그래밍 김선우 / 한빛아카데미 / 2017년 12월
- ❖ IT CookBook, Android Studio를 활용한 안드로이드 프로그래밍(7판) Android 12.0(S) 지원 우재남 외 1명 / 한빛아카데미 / 2022년 01월
- ❖ IT CookBook, 코틀린을 활용한 안드로이드 프로그래밍 우재남 외 1명 / 한빛아카데미 / 2020년 06월
- ❖ 기타 서적 및 웹 사이트 자료 다수 참조



감사합니다.

- ❖ Mobile: 010-9591-1401
- ❖ E-mail: [onlooker2zip@naver.com](mailto:onlooker2zip@naver.com)