

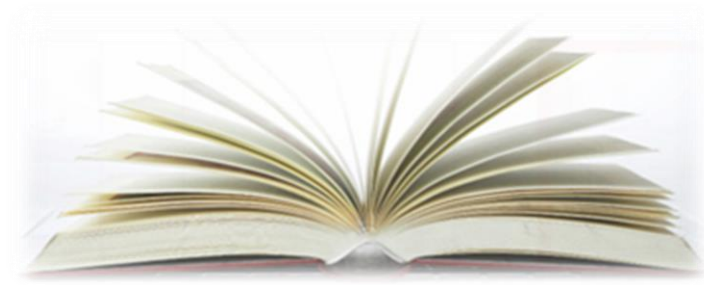
모바일프로그래밍(2) -Mobile Programming(2)-

2020. 03. 15.

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



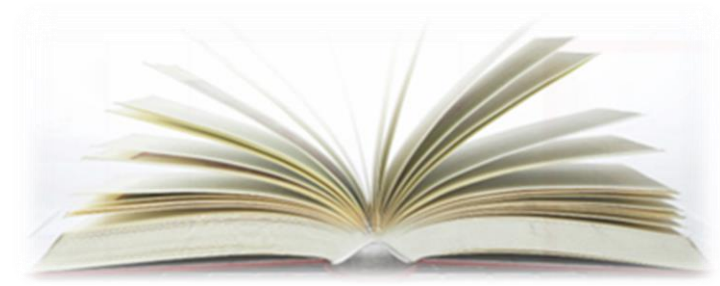
주차	기간	수업내용 및 학습활동	비고
1	03/07-03/11	· 강의 오리엔테이션 · 연구실 안전 정기교육 실시	· 주교재 01장-04장 복습
2	03/14-03/18	· 프로그램 작성하기	· 주교재 01장-04장 복습
3	03/21-03/25	· 기본문법 활용하기 · 언어특성 활용하기	· 주교재 01장-04장 복습
4	03/28-04/01	· UI 요구사항 확인하기	· 주교재 05장
5	04/04~04/08	· UI 요구사항 확인하기	· 주교재 06-07장
6	04/11~04/15	· UI 설계하기	· 주교재 08장
7	04/18~04/22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

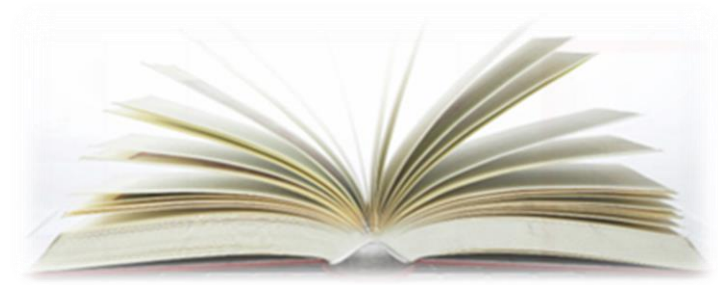
Contents



Section 1 레이아웃

..... 3

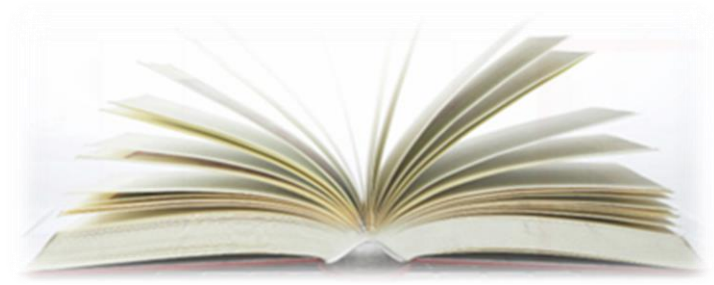
Section 1



레이아웃

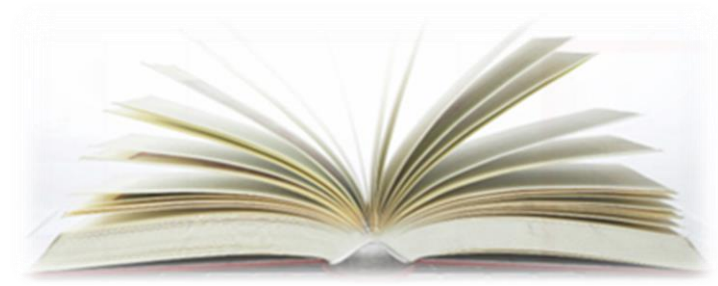
1. 뷰 = 레이아웃 + 위젯
2. 레이아웃

학습목표



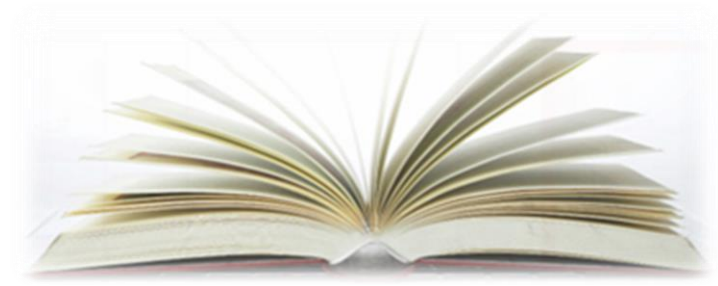
- ❖ 이 워크샵에서는 뷰의 공통 속성을 이해하고 활용한다.
- ❖ 뷰의 크기와 여백 지정 방법을 익힌다.
- ❖ 대표적인 레이아웃의 사용법을 학습한다

Subsection 1



뷰 = 레이아웃 + 위젯

Subsection 2



레이아웃

❖ 뷰(View)

- 클래스 혹은 그 서브클래스로 만든 객체를 뜻함
- 액티비티 화면은 한 개 이상의 뷰로 구성
- 크게 두 종류(레이아웃과 위젯)로 나눔

java.lang.Object

↳ android.view.View

↳ android.view.ViewGroup

↳ android.support.constraint.ConstraintLayout

(a) ConstraintLayout 클래스

그림 2-1 Hello 앱 화면을 구성하는 뷰의 클래스 계층

java.lang.Object

↳ android.view.View

↳ android.widget.TextView

↳ android.widget.Button

(b) Button 클래스

❖ 레이아웃(Layout)

- ViewGroup의 서브클래스.
- 여러 개의 뷰('자식 뷰'라 부름)를 규칙대로 화면에 배치

❖ 위젯(Widget)

- View 또는 ViewGroup의 서브클래스로서 단독으로 사용

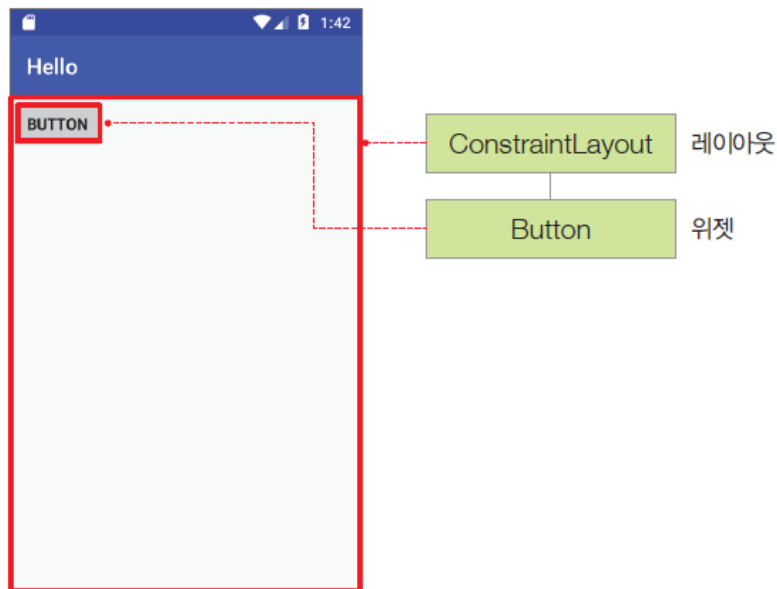
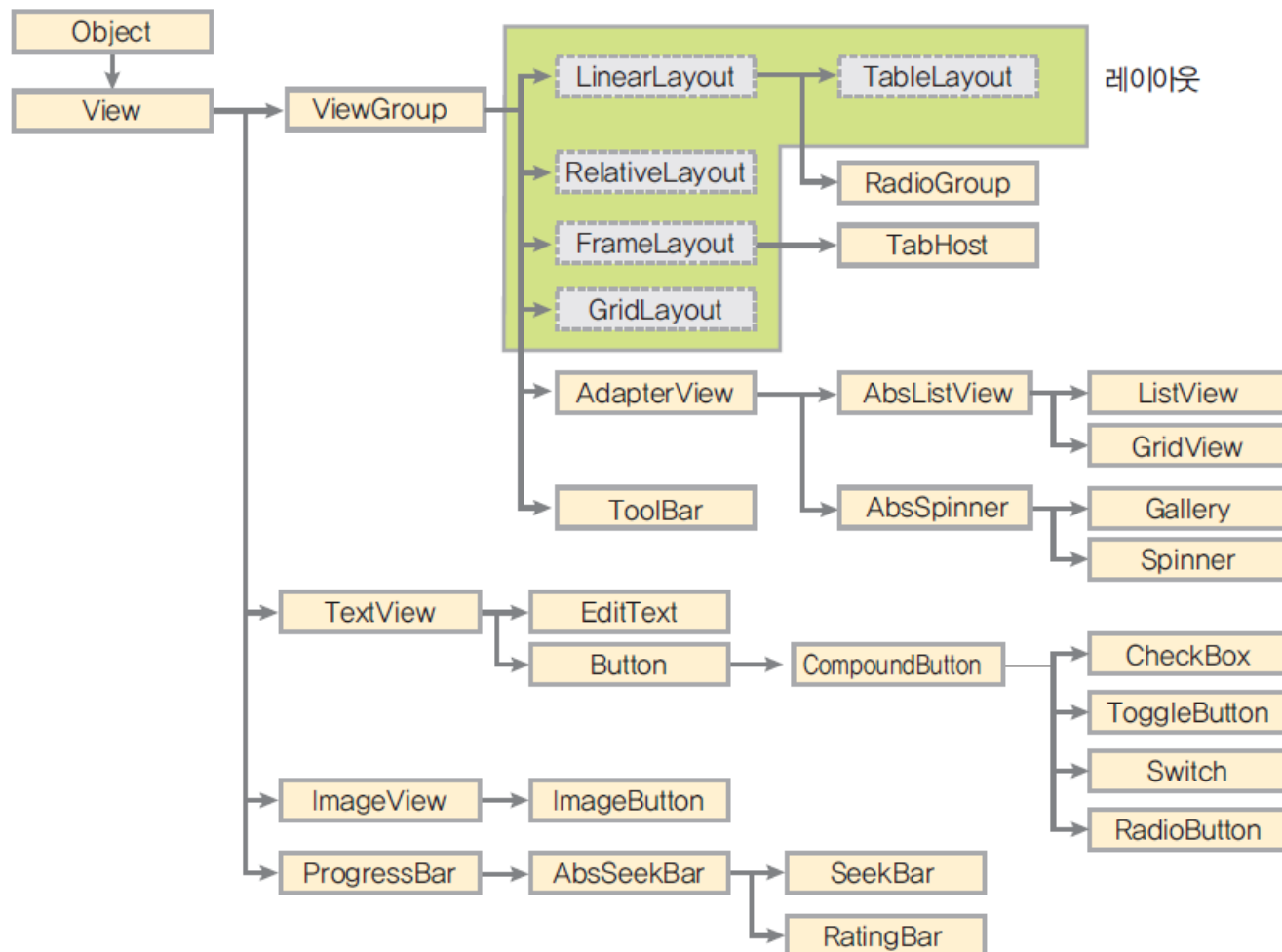


그림 2-2 Hello 앱의 화면을 구성하는 뷰 객체 = 레이아웃 + 위젯

❖ View 클래스 계층도



클래스 상속관계

<https://developer.android.com/reference/android/widget/Button>

developers

PlatformAndroid StudioGoogle PlayJetpackKotlinDocsGames

Search

한국어로그인

DOCUMENTATION

OverviewGuidesReferenceSamplesDesign & Quality

Filter

android.view.translation

android.webkit

android.widget

OverviewAnnotationsInterfacesClasses

AbsListView

AbsListView.LayoutParams

AbsoluteLayout

AbsoluteLayout.LayoutParams

AbsSeekBar

AbsSpinner

ActionMenuView

ActionMenuView.LayoutParams

AdapterView

AdapterView.AdapterContextMenuInfo

AdapterViewAnimator

Android Developers > Docs > Reference

도움이 되었나요?

Button

Added in API level 1

KotlinJava

public class Button
extends TextView

java.lang.Object
↳ android.view.View
↳ android.widget.TextView
↳ android.widget.Button

Known direct subclasses
CompoundButton

Known indirect subclasses
CheckBox, RadioButton, Switch, ToggleButton

A user interface element the user can tap or click to perform an action.

To display a button in an activity, add a button to the activity's layout XML file:

이 페이지의 내용

Summary

Inherited XML attributes

Inherited constants

Inherited fields

Public constructors

Public methods

Inherited methods

Public constructors

Button

Button

Button

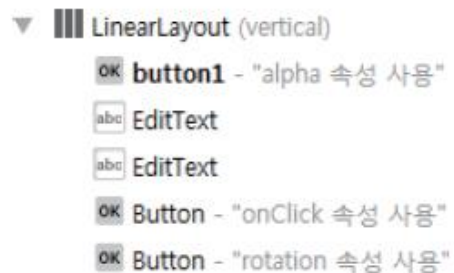
Button

Public methods

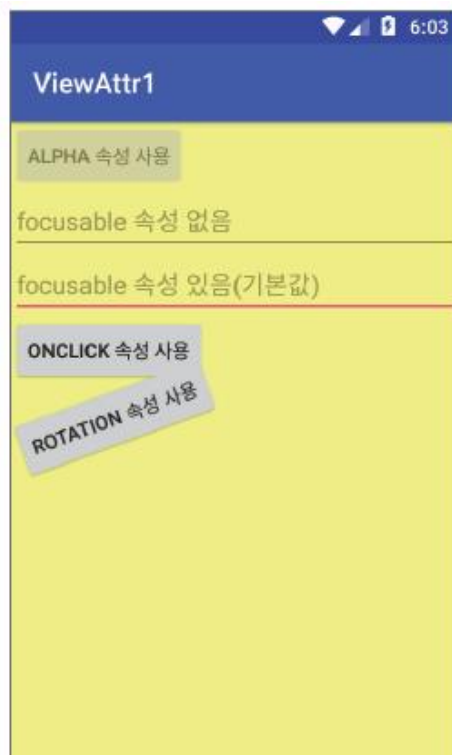
getAccessibilityClassName

onResolvePointerIcon

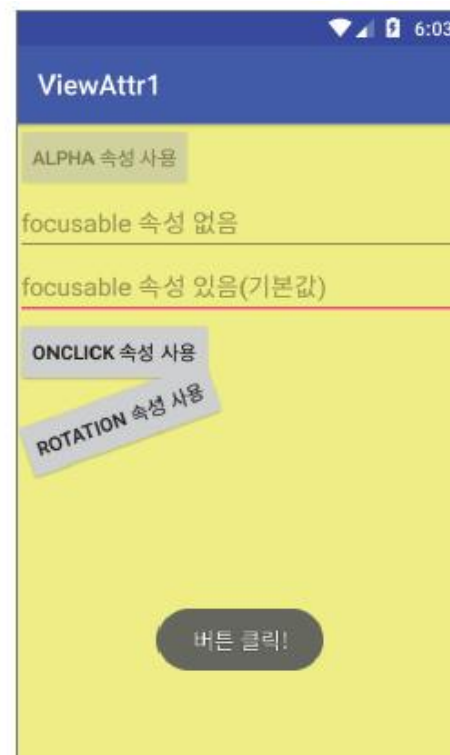
❖ 뷰의 공통 속성 분석



(a) 컴포넌트 트리



(b) 초기 화면



(c) 'ONCLICK 속성 사용' 클릭

그림 2-3 컴포넌트 트리와 실행 화면

Create New Project

Configure your project

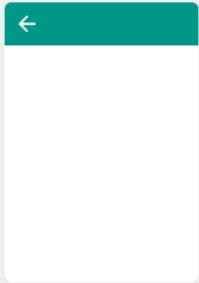
Name
P76_ViewAttr1

Package name
org.lklab.p76_viewattr1

Save location
C:\CookAndroid\Project\P76_ViewAttr1

Language
Java

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

 Empty Activity

☐ This project will support instant apps

☒ Use androidx.* artifacts

[Help me choose](#)

Creates a new empty activity

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:background="#ffff77"
    android:orientation="vertical">
    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:alpha="0.5"
        android:text="alpha 속성 사용"/>
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:focusable="false"
        android:hint="focusable 속성 없음"/>
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="focusable 속성 있음(기본값)"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="mOnClick"
        android:text="onClick 속성 사용"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:rotation="-20"
        android:text="rotation 속성 사용"/>
</LinearLayout>
```



```
package org.lklab.p76_viewattr1;

import androidx.appcompat.app.AppCompatActivity;

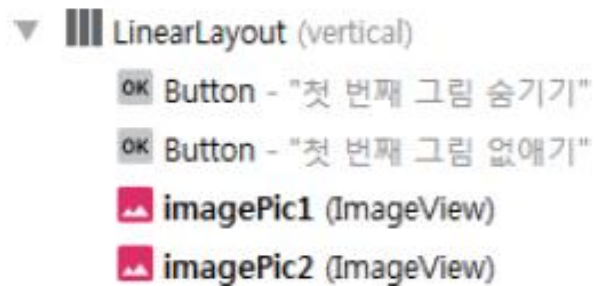
import android.os.Bundle;
import android.view.View;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

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

    public void mOnClick(View v) {
        Toast.makeText(this, "버튼 클릭!", Toast.LENGTH_SHORT).show();
    }
}
```

❖ 컴포넌트 트리와 초기 화면

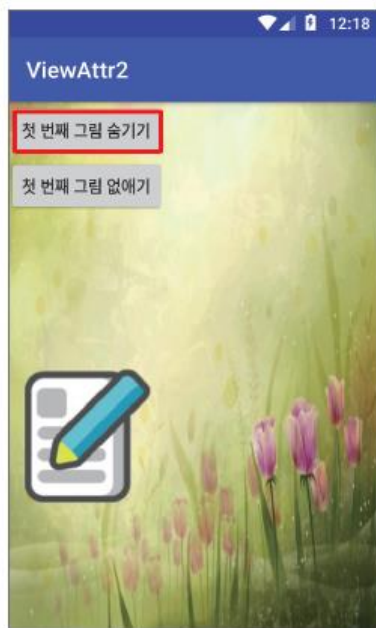


(a) 컴포넌트 트리

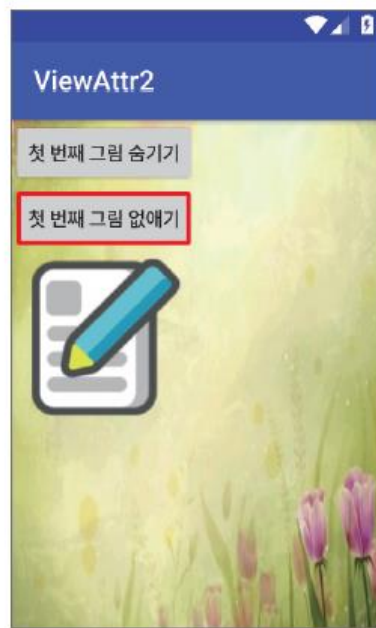


(b) 초기 화면

- ❖ (c)는 '첫 번째 그림 숨기기'를 클릭하여 INVISIBLE 속성을 적용한 결과
- ❖ (d)는 '첫 번째 그림 없애기'를 클릭하여 GONE 속성을 적용한 결과



(c) '첫 번째 그림 숨기기' 클릭

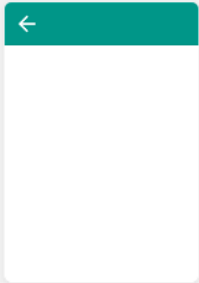


(d) '첫 번째 그림 없애기' 클릭

그림 2-4 컴포넌트 트리과 실행 화면

Create New Project

Configure your project



Empty Activity

Creates a new empty activity

Name
P78_ViewAttr2

Package name
org.lklab.p78_viewattr2

Save location
C:\CookAndroid\Project\P78_ViewAttr2

Language
Java

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

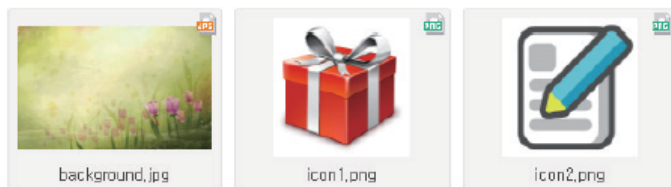
i Your app will run on approximately **100%** of devices.
[Help me choose](#)

☐ This project will support instant apps

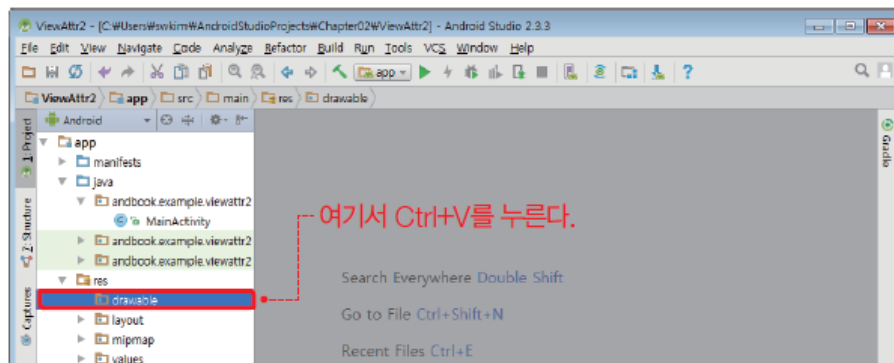
☒ Use androidx.* artifacts

Previous Next Cancel Finish

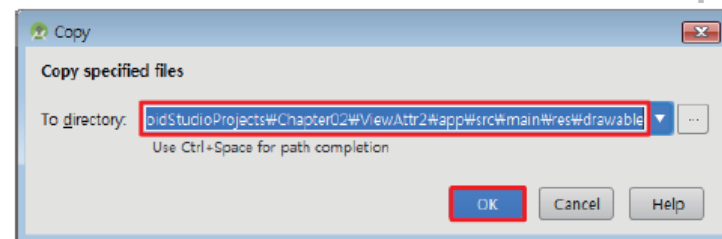
❖ 준비한 그림 배치



(a) 준비한 그림 파일



(b) 그림 파일 붙여넣기



(c) Copy 대화상자

그림 2-5 drawable 리소스 준비

```
<?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:background="@drawable/background"
    android:orientation="vertical">
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="mOnClick1"
        android:text="첫 번째 그림 숨기기"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="mOnClick2"
        android:text="첫 번째 그림 없애기"/>
    <ImageView
        android:id="@+id/imagePic1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/icon1"/>
    <ImageView
        android:id="@+id/imagePic2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/icon2"/>
</LinearLayout>
```

```
package org.lklab.p78_viewattr2;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;
import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {

    private ImageView mImagePic1;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mImagePic1 = (ImageView) findViewById(R.id.imagePic1);
    }

    public void mOnClick1(View v) {
        mImagePic1.setVisibility(View.INVISIBLE);
    }

    public void mOnClick2(View v) {
        mImagePic1.setVisibility(View.GONE);
    }
}
```

❖ 제공되는 크기 지정 단위

- 레이아웃 리소스에서 뷰의 크기를 지정할 때 숫자 뒤에 붙여서 사용

표 2-2 크기 지정 단위

단위	의미
px	픽셀(pixel)이다. 특수한 경우를 제외하고는 사용하지 않는 편이 좋다.
mm in	각각 밀리미터(millimeter)와 인치(inch)이다. 다양한 디바이스에서 일관된 크기로 출력할 수 있다는 점에서 픽셀 단위보다는 좋지만 세밀한 크기를 지정하려면 소수점 이하 숫자가 필요한 경우가 많아서 사용을 권장하지 않는다.
pt	1/72인치다. 인쇄 기술에서 유래한 단위로 주로 글자 크기를 지정할 때 사용된다.
dp 또는 dip (사용 권장)	<p>밀도 독립적 픽셀(Density-Independent Pixel)이다. 160dp로 출력하면 논리적 1인치가 되도록 한다는 약속된 단위다. 여기서 논리적 1인치는 실제 1인치 길이와는 조금 다를 수 있다. 실제 출력 시에는 디바이스에 따라 물리적 픽셀 수가 결정된다. 다양한 디바이스에서 일관된 크기로 출력할 수 있고 세밀한 크기를 지정할 때 소수점 이하 숫자가 필요한 경우가 적다는 장점이 있다.</p> <p>참고 예를 들어 160dp로 출력한 결과는 160dpi 디바이스에서는 160픽셀로, 320dpi 디바이스에서는 320픽셀로 출력된다. 여기서 dpi(dot-per-inch)는 디스플레이의 물리적 특징이 아니라 안드로이드 플랫폼에서 정한 값이다. 안드로이드 7.0부터는 이전 버전과 달리 시스템 설정에서 dpi를 사용자가 변경할 수 있다. 안드로이드 에뮬레이터의 [설정]-[디스플레이]-[디스플레이 크기]로 들어가 살펴보자.</p>
sp (사용 권장)	스케일 독립적 픽셀(Scale-Independent Pixel)이다. dp와 기본 정의는 같지만 안드로이드 시스템 전체의 글꼴 크기를 설정에서 증감하면 그에 맞게 크기가 증감한다는 차이가 있다. 시력이 좋지 않은 사용자를 고려하여 화면을 디자인할 때 (특히 글자 크기를 지정할 때) 사용을 권장한다.

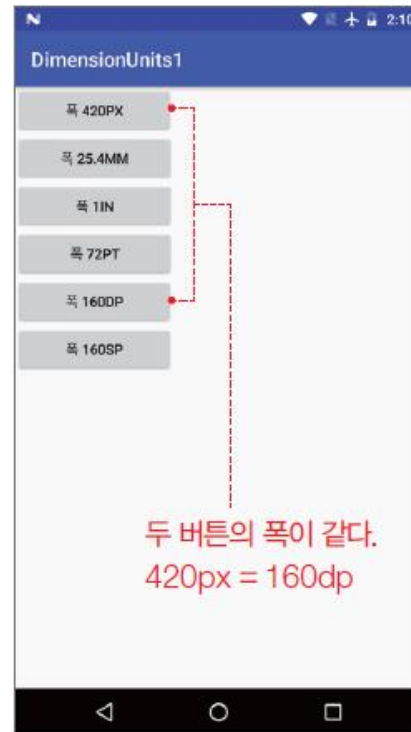
❖ 크기 지정 단위를 버튼의 폭에 적용한 결과



(a) 에뮬레이터(넥서스 S)

→ 480*800 해상도, 240dpi

그림 2-6 실행 화면

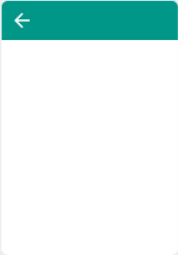


(b) 넥서스 5X

→ 1080*1920 해상도, 420dpi

Create New Project

Configure your project

 Empty Activity

Creates a new empty activity


Name
P83_DimensionUnits1

Package name
org.lklab.p83_dimensionunits1

Save location
C:\CookAndroid\ProjectWP83_DimensionUnits1

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 androidx.* 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">
    <Button
        android:layout_width="240px"
        android:layout_height="wrap_content"
        android:text="폭 240px"/>
    <Button
        android:layout_width="25.4mm"
        android:layout_height="wrap_content"
        android:text="폭 25.4mm"/>
    <Button
        android:layout_width="1in"
        android:layout_height="wrap_content"
        android:text="폭 1in"/>
    <Button
        android:layout_width="72pt"
        android:layout_height="wrap_content"
        android:text="폭 72pt"/>
    <Button
        android:layout_width="160dp"
        android:layout_height="wrap_content"
        android:text="폭 160dp"/>
    <Button
        android:layout_width="160sp"
        android:layout_height="wrap_content"
        android:text="폭 160sp"/>
</LinearLayout>
```

```
package org.lklab.p83_dimensionunits1;

import androidx.appcompat.app.AppCompatActivity;

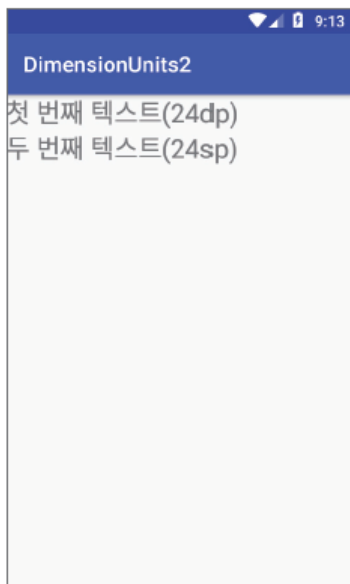
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

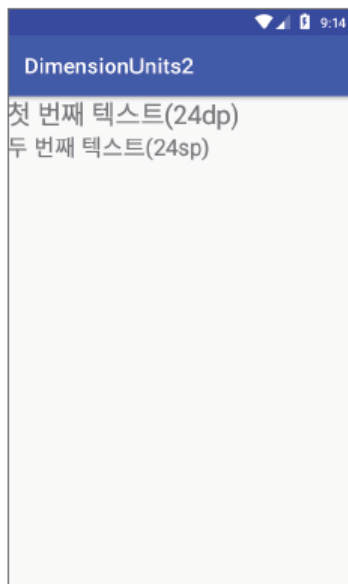
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

❖ dp와 sp의 차이

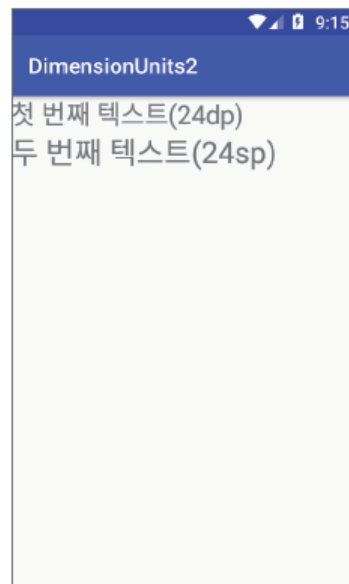
- 시스템 설정의 [디스플레이]-[글꼴 크기] 메뉴에서 시스템 전체 글꼴의 크기를 변경하면 sp 단위로 출력한 텍스트의 크기만 변경



(a) 초기 화면(기본 글꼴)



(b) 작은 글꼴 선택 시

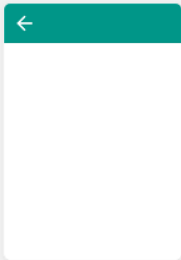


(c) 큰 글꼴 선택 시

그림 2-7 실행 화면

Create New Project

Configure your project



Empty Activity

Creates a new empty activity


Name
P85_DimensionUnits2

Package name
org.lklab.p85_dimensionunits2

Save location
C:\CookAndroid\Project\P85_DimensionUnits2

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 androidx.* 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">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="첫 번째 텍스트(24dp)"
        android:textSize="24dp"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="두 번째 텍스트(24sp)"
        android:textSize="24sp"/>
</LinearLayout>
```

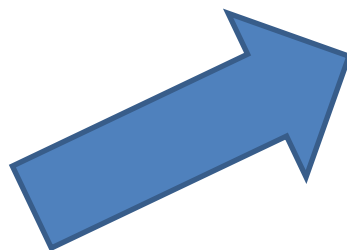
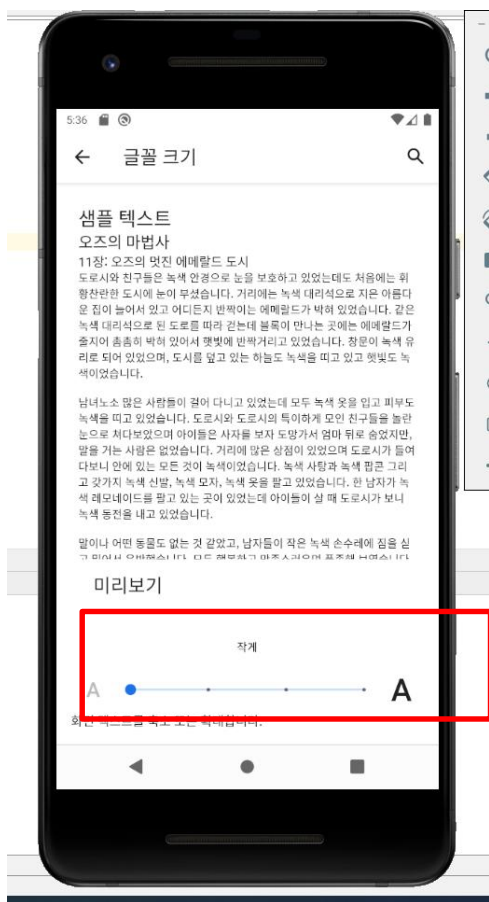
```
package org.lklab.p85_dimensionunits2;

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);
    }
}
```

sp 단위로
출력한 텍스
트의 크기만
변경



❖ 마진margin

- 외부 여백, 즉 뷰와 부모 뷰 사이의 공간
- 마진으로 지정한 여백은 뷰 자신의 영역에 포함되지 않음

❖ 패딩padding

- 내부 여백, 즉 뷰와 뷰의 내용물 사이의 공간이다. 패딩으로 지정한 여백은 뷰 자신의 영역에 포함된다.

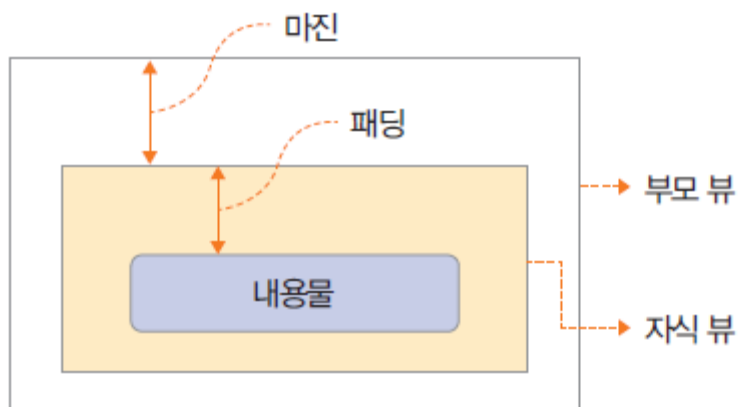
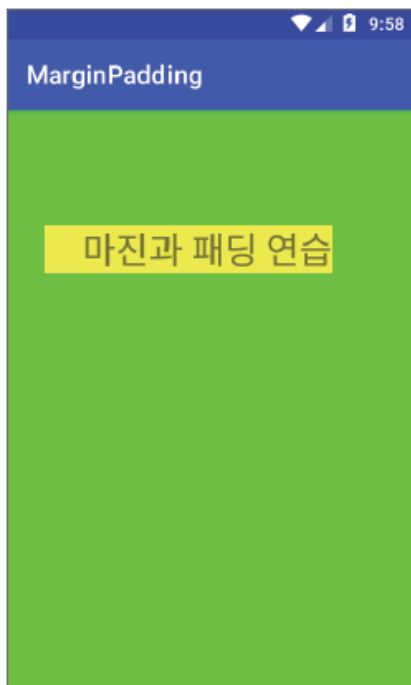


그림 2-8 마진과 패딩

❖ 마진과 패딩을 부모 뷰와 자식 뷰에 적용한 예



(a) 실행 화면



(b) 여백 분석

그림 2-9 실행 화면과 여백 분석

Create New Project

Configure your project

Name
P87_MarginPadding

Package name
org.lklab.p87_marginpadding

Save location
C:\CookAndroid\Project\WP87_MarginPadding

Language
Java

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

Empty Activity

Creates a new empty activity

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

☐ This project will support instant apps

☒ Use androidx.* 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:background="#00ff00"
    android:orientation="vertical"
    android:padding="30dp">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="60dp"
        android:background="#ffff00"
        android:paddingLeft="30dp"
        android:text="마진과 패딩 연습"
        android:textSize="28sp"/>
</LinearLayout>
```

```
package org.lklab.p87_marginpadding;

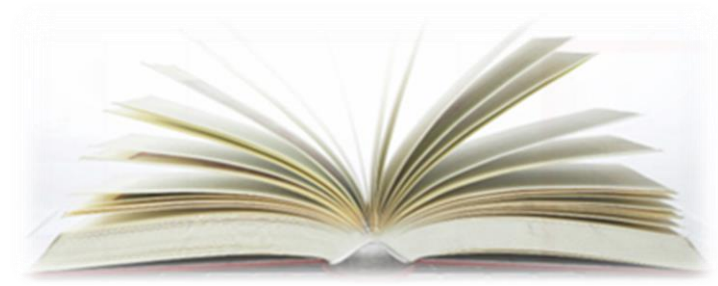
import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

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

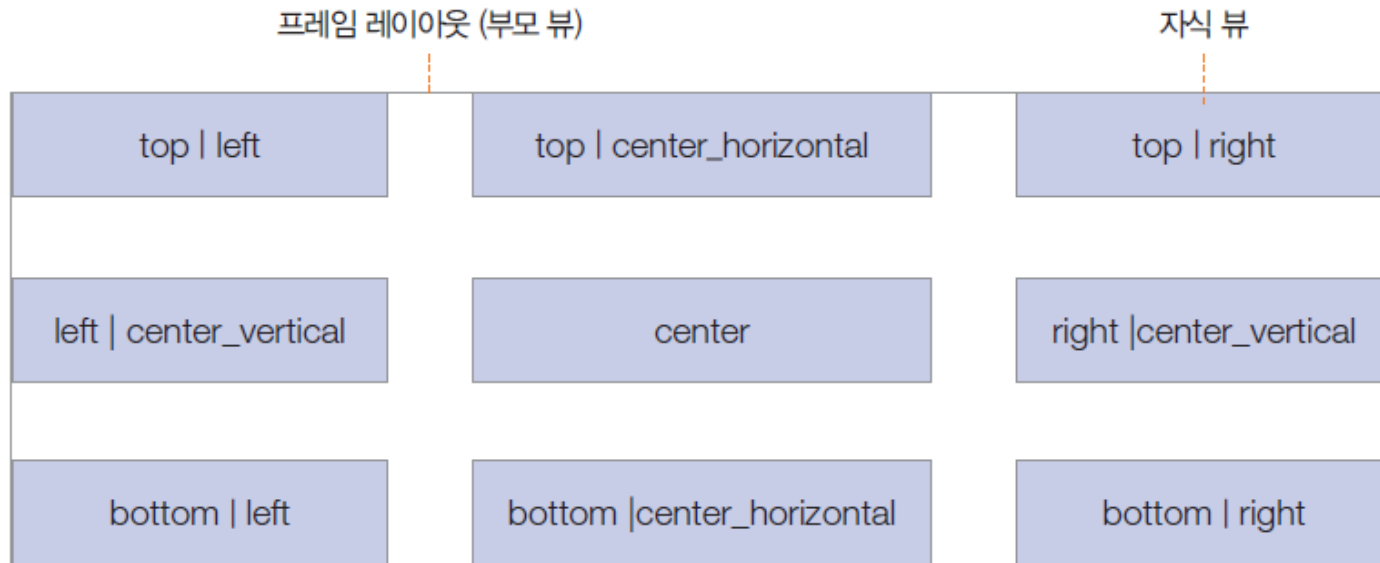
Subsection 2



레이아웃

❖ 프레임 레이아웃(FrameLayout)

- 화면의 일정 영역을 차지하는 틀(frame) 안에 자식 뷰를 배치하는 뷰 그룹
- 기본 동작은 자식 뷰를 프레임 레이아웃의 좌측 상단에 배치하는 것
- 그라비티(layout_gravity)속성을 사용하면 자식 뷰의 위치를 변경 가능



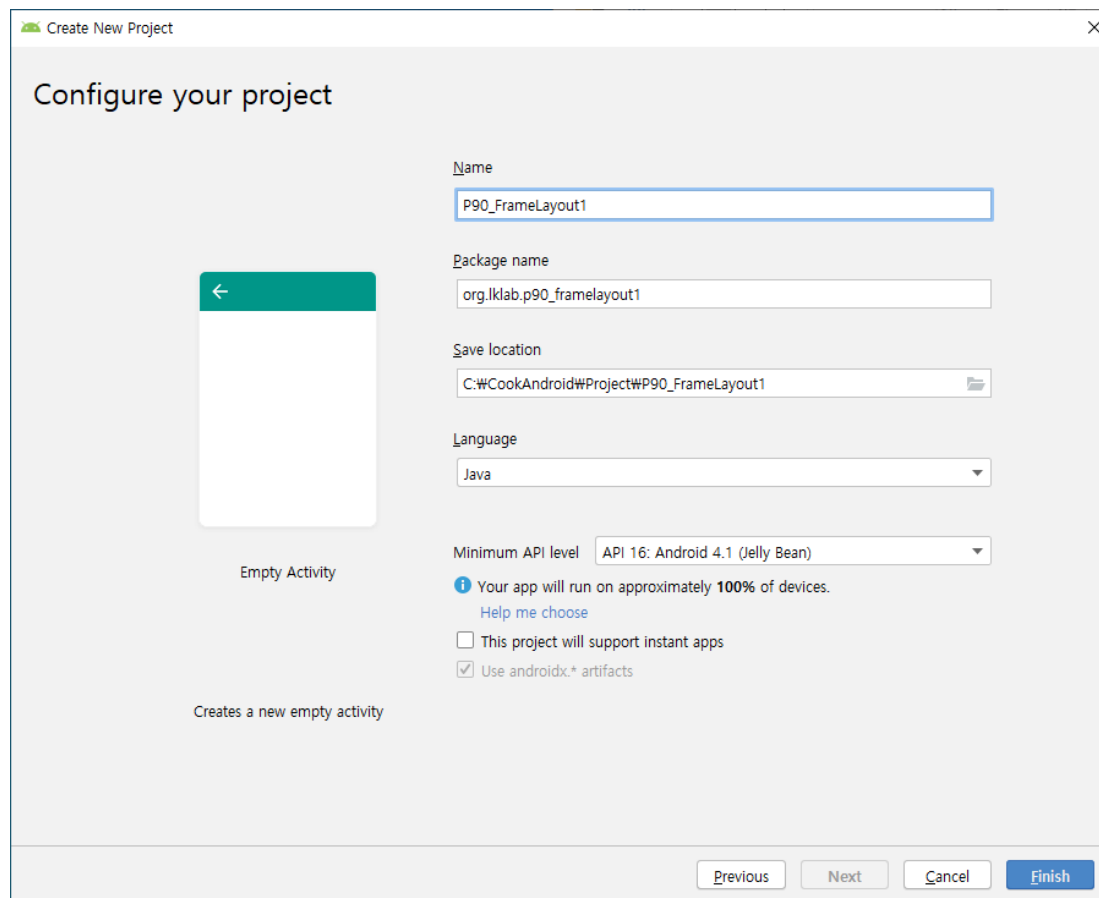
- top과 left는 기본값이므로 생략 가능
- center = center_horizontal | center_vertical

그림 2-10 그라비티 속성값에 따른 자식 뷰의 위치

- ❖ 프레임 레이아웃 안에 자식 뷰를 그래비티 속성값을 바꿔가며 배치
- ❖ 우측 상단의 텍스트뷰와 아날로그 시계는 겹쳐서 출력



그림 2-11 실행 화면



```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <AnalogClock
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="right"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="right"
        android:text="현재 시각을 보여줍니다."
        android:textColor="#ff0000"
        android:textSize="18sp"/>
    <DigitalClock
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:textColor="#0000ff"
        android:textSize="24sp"/>
</FrameLayout>
```

```
package org.lklab.p90_framelayout1;

import androidx.appcompat.app.AppCompatActivity;

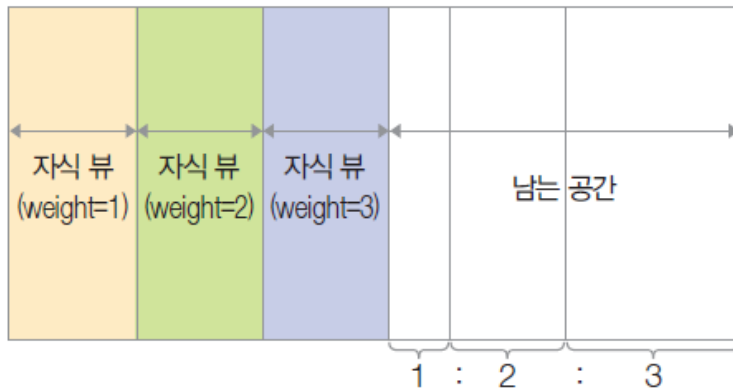
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

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

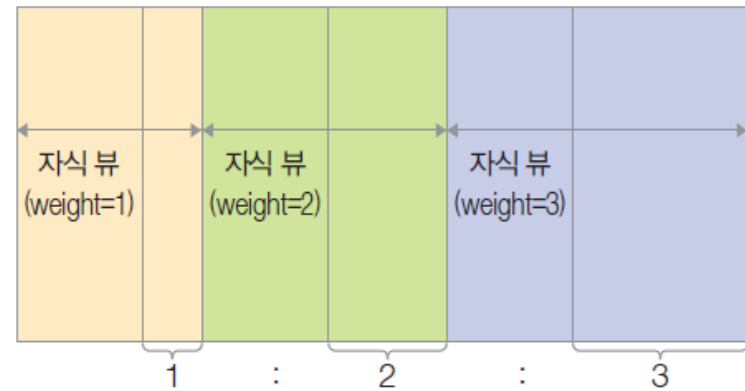
❖ 리니어 레이아웃(LinearLayout)

- 자식 뷰를 가로 또는 세로 방향으로 일렬로 배치하는 뷰 그룹
- 기본값은 가로 방향이며 orientation 속성값을 "vertical"로 하면 세로 방향이 됨
- 리니어 레이아웃에 자식 뷰를 차례로 배치하다 남는 공간이 생기면, layout_weight 속성을 자식 뷰에 적용하여 일정 비율로 나누어줄 수 있음
- 남는 공간이 없으면 layout_weight 속성이 제대로 동작하지 않으므로 주의



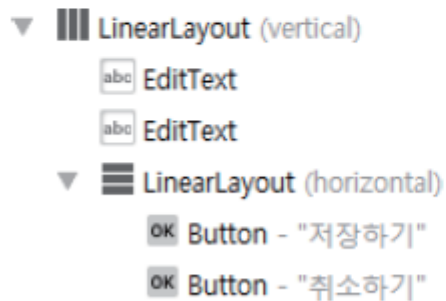
(a) 리니어 레이아웃 내부의 남는 공간

그림 2-12 layout_weight 속성

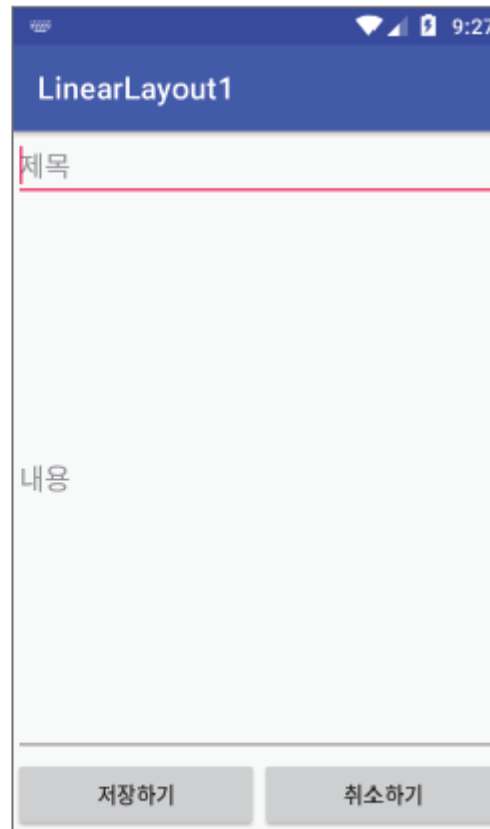


(b) 자식 뷰에 1:2:3으로 나누어준다.

❖ 리니어 레이아웃의 기본 사용법

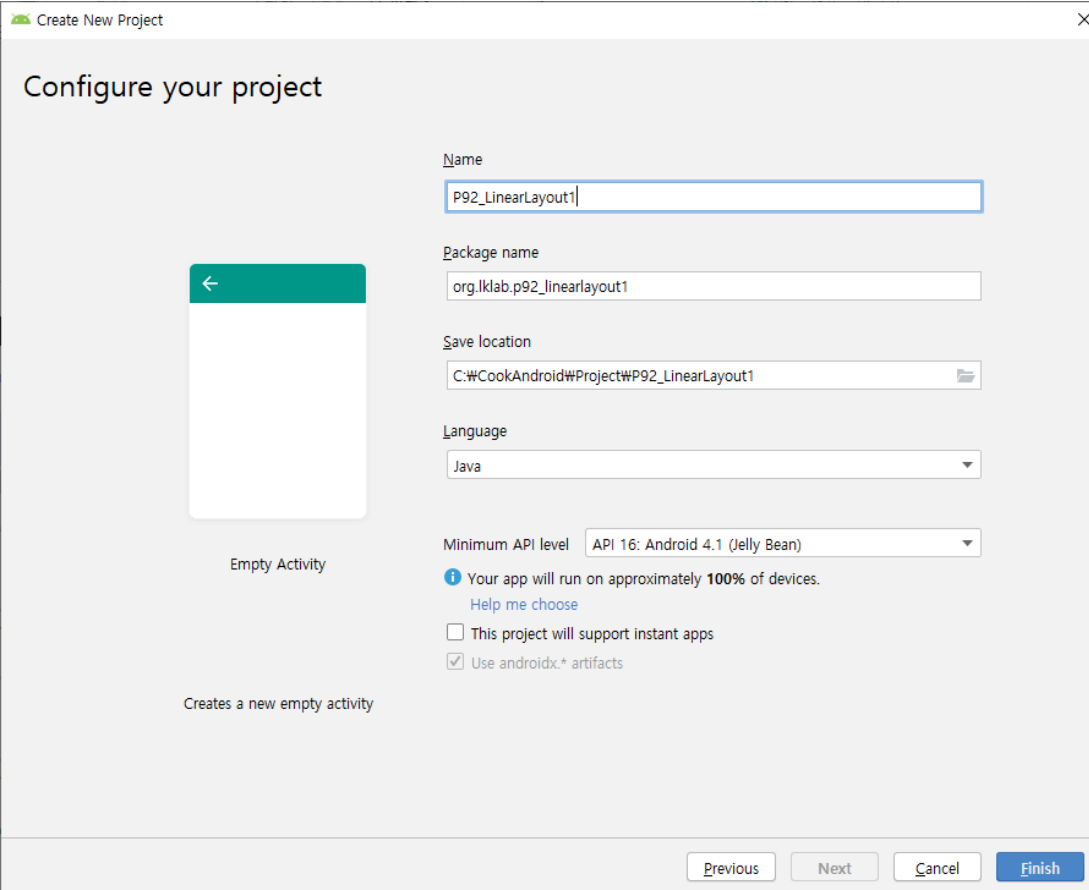


(a) 컴포넌트 트리



(b) 실행 화면

그림 2-13 컴포넌트 트리와 실행 화면



Create New Project

Configure your project

Name
P92_LinearLayout1

Package name
org.lklab.p92_linearlayout1

Save location
C:\\CookAndroid\\Project\\P92_LinearLayout1

Language
Java

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

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

☐ This project will support instant apps

☒ Use androidx.* artifacts

Empty Activity

Creates a new empty activity

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="제목"/>
    <EditText
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:hint="내용"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <Button
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="저장하기"/>
        <Button
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="취소하기"/>
    </LinearLayout>
</LinearLayout>
```



```
package org.lklab.p92_linearlayout1;

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);
    }
}
```

❖ **baselineAligned의 기능**

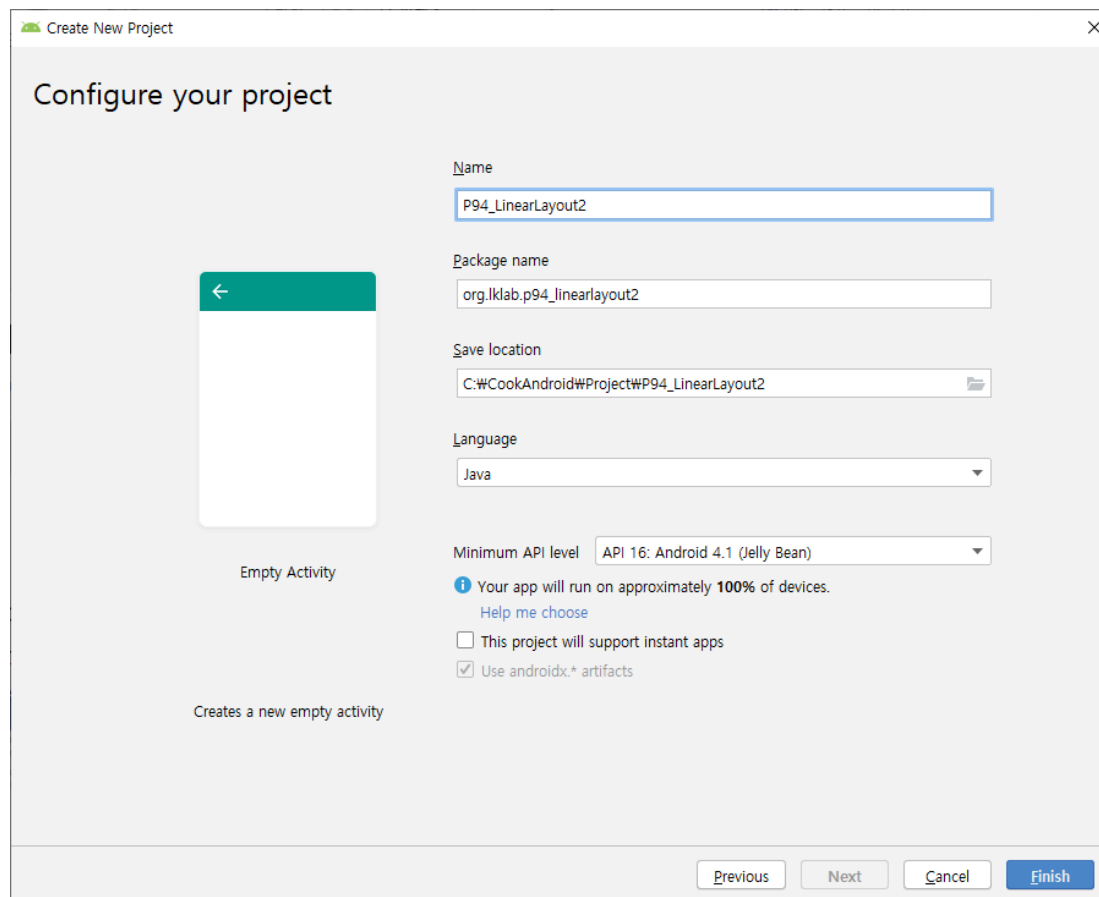
- 텍스트를 가로 방향으로 배치할 때 기준선(baseline)에 맞출지를 결정하는 값
- 기본값은 "true"
- 속성값을 "false"로 변경하면 기준선이 맞지 않아 부자연스러움
- 한글은 기준선 개념은 없지만 baselineAligned 속성의 영향을 받음



그림 2-14 기준선 개념



그림 2-15 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">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Jellybean"
            android:textSize="14sp"/>
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Lollipop"
            android:textSize="24sp"/>
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Programming"
            android:textSize="20sp"/>
    </LinearLayout>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:baselineAligned="false">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Jellybean"
        android:textSize="14sp"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Lollipop"
        android:textSize="24sp"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Programming"
        android:textSize="20sp"/>
</LinearLayout>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="젤리빈"
        android:textSize="14sp"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="롤리팝"
        android:textSize="24sp"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Programming"
        android:textSize="20sp"/>
</LinearLayout>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:baselineAligned="false">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="젤리 빈"
        android:textSize="14sp"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="롤리팝"
        android:textSize="24sp"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Programming"
        android:textSize="20sp"/>
</LinearLayout>
</LinearLayout>
```

```
package org.lklab.p94_linearlayout2;

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);
    }
}
```


❖ 렐러티브 레이아웃(RelativeLayout)

- 자식 뷰를 부모(parent) 뷰나 형제(sibling) 뷰를 기준으로 상대적인 위치를 주어 배치하는 뷰 그룹
- 자식 뷰의 기본 위치: 부모 뷰의 좌측 상단이지만 다양한 layout_~ 속성이 마련되어 있어 자유롭게 위치 지정

❖ 부모 뷰를 기준으로 위치 지정

- 자식 뷰의 속성 이름으로 layout_alignParent~ 또는 layout_center~ 형태를 사용하며 속성값으로 "true" 또는 "false"를 지정한다.

❖ 부모 뷰를 기준으로 위치 지정

표 2-5 부모 뷰를 기준으로 자식 뷰의 위치 지정

XML 속성	기능
layout_alignParentTop	부모 뷰의 윗면과 자식 뷰의 윗면을 일치시킨다.
layout_alignParentBottom	부모 뷰의 아랫면과 자식 뷰의 아랫면을 일치시킨다.
layout_alignParentLeft	부모 뷰의 왼쪽 면과 자식 뷰의 왼쪽 면을 일치시킨다.
layout_alignParentRight	부모 뷰의 오른쪽 면과 자식 뷰의 오른쪽 면을 일치시킨다.
layout_alignWithParentIfMissing	<p>앵커가 없으면 부모 뷰를 기준으로 자식 뷰를 정렬한다.</p> <p>참고 이 속성은 설명만으로 이해하기 어려우므로 잠시 후 RelativeLayout2 예제로 테스트할 것이다.</p> <p>참고 자식 뷰의 위치 지정 시 기준이 되는 형제 뷰를 안드로이드 개발자 문서에서 앵커(anchor)라 한다.</p>
layout_centerHorizontal	부모 영역 내에서 수평으로 중앙에 자식 뷰를 배치한다.
layout_centerVertical	부모 영역 내에서 수직으로 중앙에 자식 뷰를 배치한다.
layout_centerInParent	부모 영역 내의 정중앙에 자식 뷰를 배치한다.

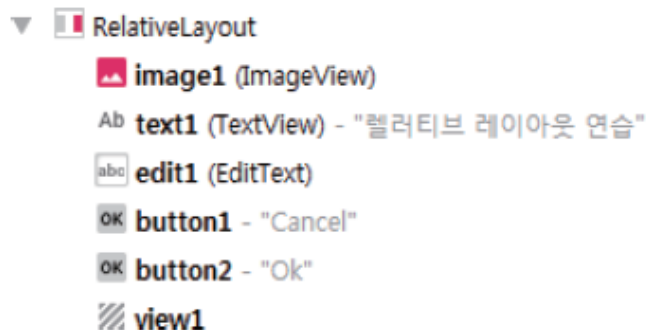
❖ 부모 뷰를 기준으로 위치 지정

- 자식 뷰의 속성 이름으로 `layout_align~` 또는 `layout_{above|below}` 또는 `layout_to{Left|Right}Of` 형태를 사용
- 속성값으로 “@+id/식별자” 또는 “@id/식별자”를 지정

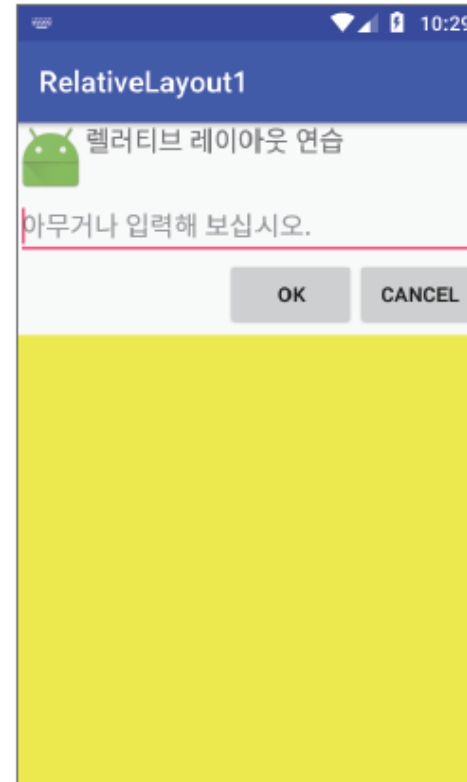
표 2-6 형제 뷰를 기준으로 자식 뷰의 위치 지정

XML 속성	기능
<code>layout_alignTop</code>	앵커의 윗면과 자식 뷰의 윗면을 일치시킨다.
<code>layout_alignBottom</code>	앵커의 아랫면과 자식 뷰의 아랫면을 일치시킨다.
<code>layout_alignLeft</code>	앵커의 왼쪽 면과 자식 뷰의 왼쪽 면을 일치시킨다.
<code>layout_alignRight</code>	앵커의 오른쪽 면과 자식 뷰의 오른쪽 면을 일치시킨다.
<code>layout_alignBaseline</code>	앵커의 기준선과 자식 뷰의 기준선을 일치시킨다. 참고 이 속성도 잠시 후 <code>RelativeLayout2</code> 예제로 테스트할 것이다.
<code>layout_above</code>	앵커의 위쪽에 자식 뷰를 배치한다.
<code>layout_below</code>	앵커의 아래쪽에 자식 뷰를 배치한다.
<code>layout_toLeftOf</code>	앵커의 왼쪽에 자식 뷰를 배치한다.
<code>layout_toRightOf</code>	앵커의 오른쪽에 자식 뷰를 배치한다.

❖ 렐러티브 레이아웃의 기본 사용법



(a) 컴포넌트 트리

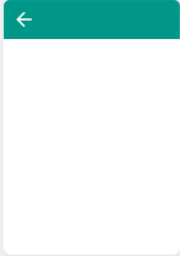


(b) 실행 화면

그림 2-16 컴포넌트 트리와 실행 화면

Create New Project

Configure your project



Empty Activity

Creates a new empty activity


Name
P97_RelativeLayout1

Package name
org.lklab.p97_relativelayout1

Save location
C:\CookAndroid\ProjectWP97_RelativeLayout1

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"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <ImageView
        android:id="@+id/image1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@mipmap/ic_launcher"/>
    <TextView
        android:id="@+id/text1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_toRightOf="@+id/image1"
        android:text="렐러티브 레이아웃 연습"
        android:textSize="18sp"/>
    <EditText
        android:id="@+id/edit1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_below="@+id/image1"
        android:hint="아무거나 입력해 보십시오."/>
    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:layout_below="@+id/edit1"
        android:text="Cancel"/>
```

```
<Button
    android:id="@+id/button2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/edit1"
    android:layout_toLeftOf="@+id/button1"
    android:text="Ok"/>
<View
    android:id="@+id/view1"
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_alignParentBottom="true"
    android:layout_below="@+id/button2"
    android:layout_marginTop="4dp"
    android:background="#ffff00"/>
</RelativeLayout>
```

```
package org.lklab.p97_relativelayout1;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

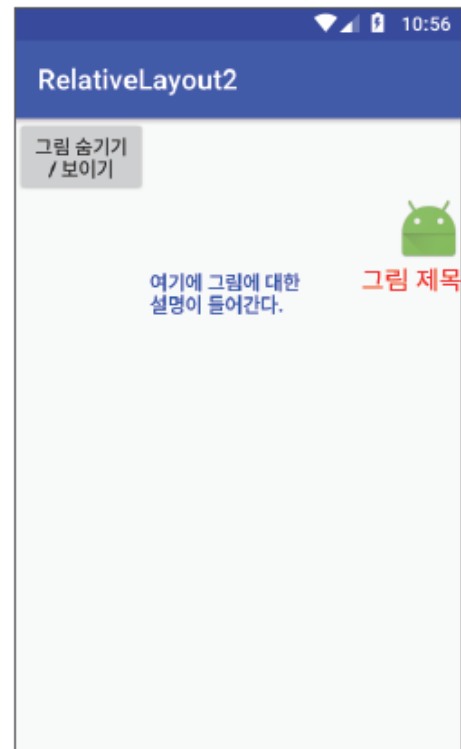
public class MainActivity extends AppCompatActivity {

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


❖ alignBaseline과 alignWithParentIfMissing 속성 사용



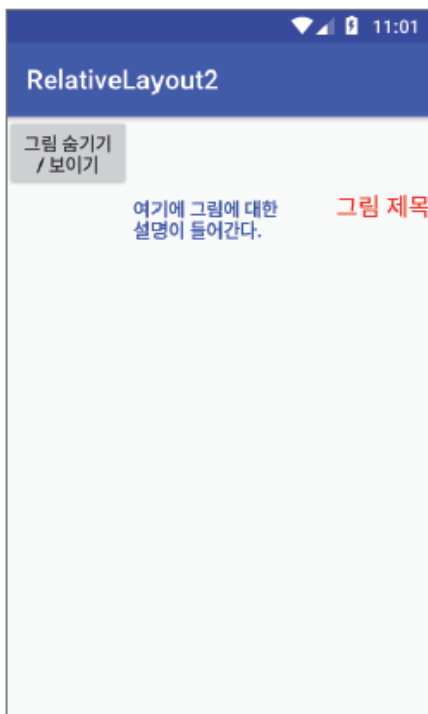
(a) 컴포넌트 트리



(b) 실행 화면

그림 2-17 컴포넌트 트리와 실행 화면

❖ 버튼을 클릭하면 앵커 역할을 하는 이미지뷰가 화면에서 없어짐



(a) `alignWithParentIfMissing="true"` (b) `alignWithParentIfMissing="false"`

그림 2-18 앵커가 없어졌을 때 `alignWithParentIfMissing` 속성값에 따른 차이

Create New Project

Configure your project

Name
P99_RelativeLayout2

Package name
org.lklab.p99_relativelayout2

Save location
C:\CookAndroid\Project\P99_RelativeLayout2

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 androidx.* artifacts

[Help me choose](#)

Previous Next Cancel Finish

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="mOnClick"
        android:text="그림 숨기기\n/ 보이기"/>
    <ImageView
        android:id="@+id/image1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:layout_below="@+id/button1"
        android:src="@mipmap/ic_launcher"/>
    <TextView
        android:id="@+id/text1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignRight="@+id/image1"
        android:layout_alignWithParentIfMissing="true"
        android:layout_below="@+id/image1"
        android:text="그림 제목"
        android:textColor="#ff0000"
        android:textSize="18sp"/>
```

```
<TextView
    android:id="@+id/text2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/text1"
    android:layout_marginLeft="2dp"
    android:layout_marginRight="2dp"
    android:layout_toLeftOf="@+id/text1"
    android:layout_toRightOf="@id/button1"
    android:text="여기에 그림에 대한\n설명이 들어간다."
    android:textColor="#0000ff"
    android:textSize="14sp"/>
</RelativeLayout>
```

```
package org.lklab.p99_relativelayout2;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;
import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {

    private ImageView mImage1;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mImage1 = (ImageView) findViewById(R.id.image1);
    }

    public void mOnClick(View v) {
        if (mImage1.getVisibility() == View.VISIBLE)
            mImage1.setVisibility(View.GONE);
        else
            mImage1.setVisibility(View.VISIBLE);
    }
}
```

❖ 테이블 레이아웃(TableLayout)

- 테이블table (표)의 각 셀에 자식 뷰를 차례로 배치하는 뷰 그룹
- 한 개 이상의 행(row)으로 구성
- 주로 TableRow 객체가 사용되며 가끔은 일반 View 객체가 사용
- TableRow 객체에는 각 셀에 자식 뷰가 한 개씩 들어갈 수 있으며 전체 열column의 개수는 셀 개수가 가장 많은 TableRow 객체로 결정
- TableRow 객체 대신 일반 View 객체가 사용되면 해당 View 객체가 한 행 전체를 무조건 차지
- 각각의 자식 뷰는 폭을 지정할 수 없고(폭은 항상 MATCH_PARENT) 높이만 지정할 수 있다.

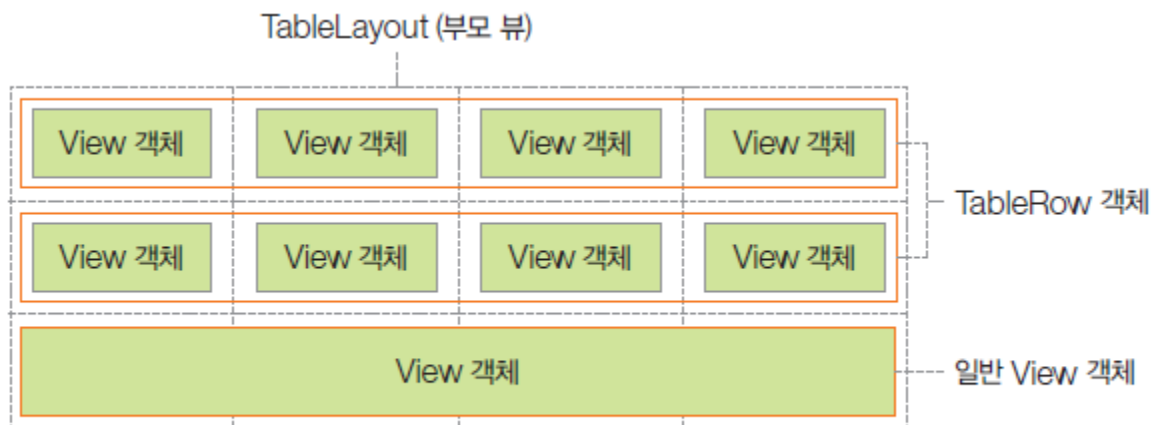
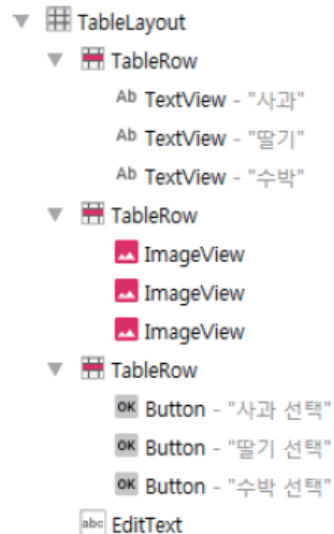


그림 2-19 테이블 레이아웃과 TableRow 객체

❖ 네 개의 행을 가진 테이블 레이아웃



(a) 컴포넌트 트리



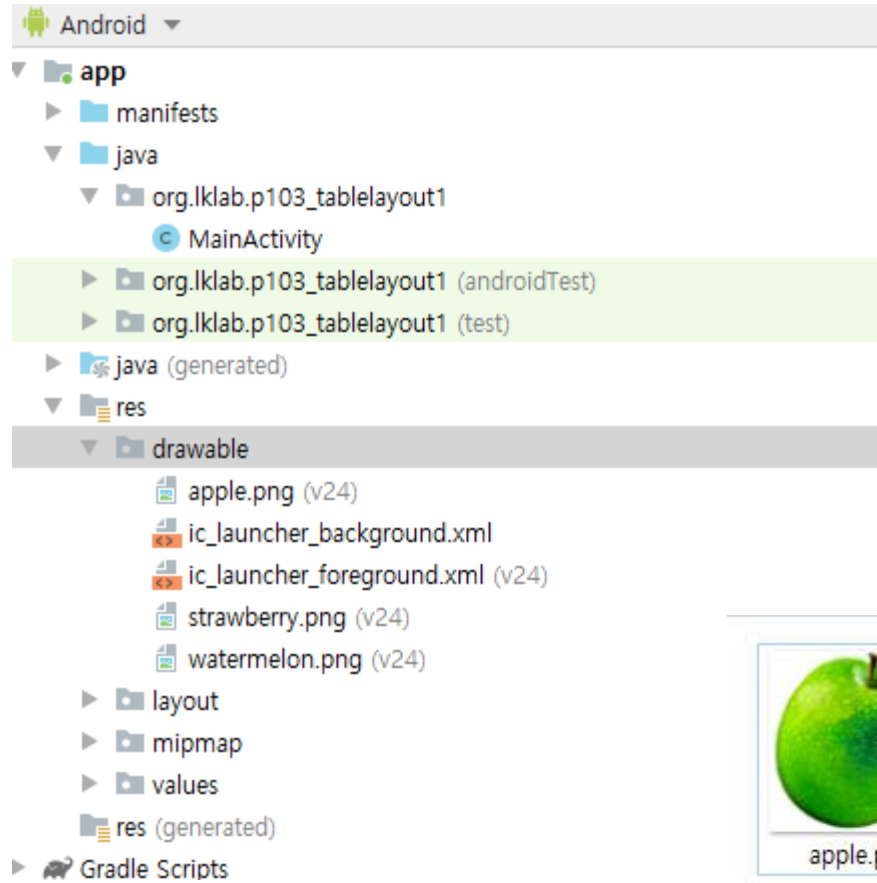
(b) 실행 화면 (1) - 세로 방향

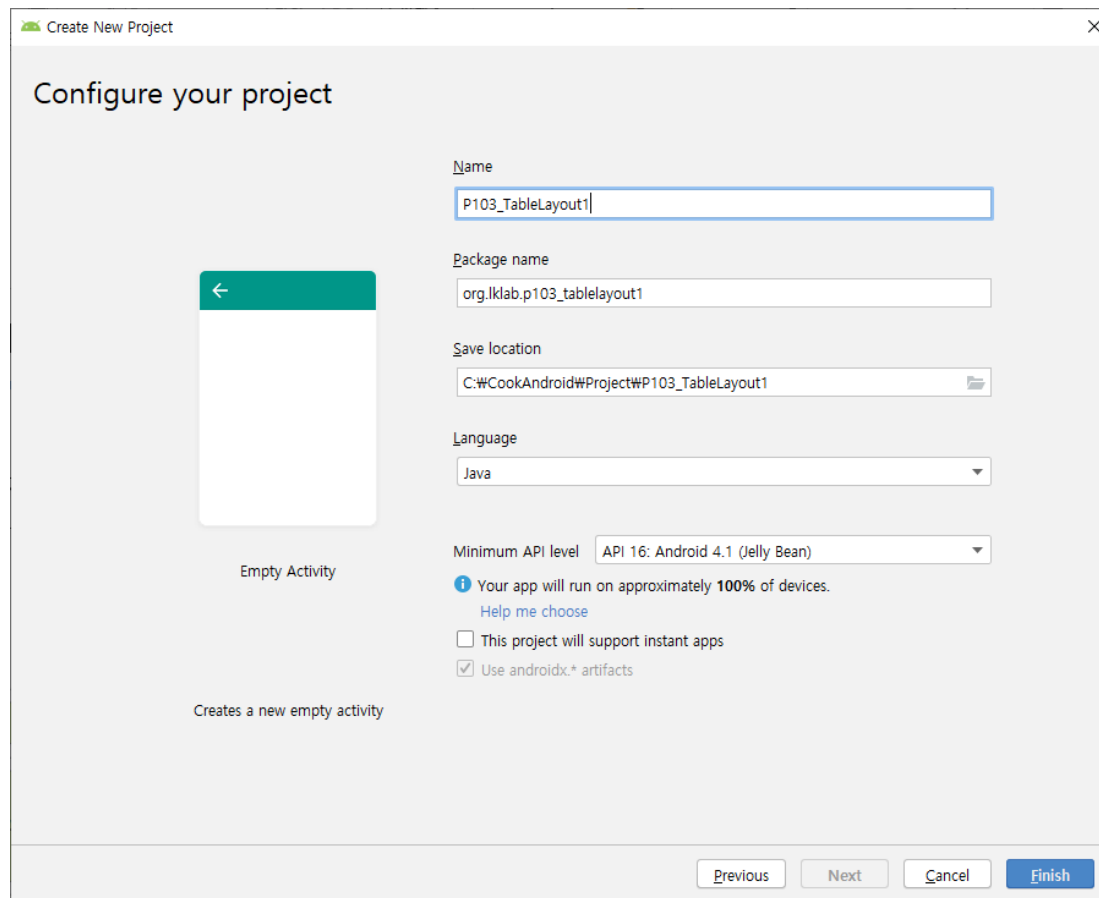


(c) 실행 화면 (2) - 가로 방향

그림 2-20 컴포넌트 트리와 실행 화면

❖ 이미지 배치





```
<?xml version="1.0" encoding="utf-8"?>
<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:shrinkColumns="2"
    android:stretchColumns="2">
    <TableRow
        android:layout_width="wrap_content"
        android:layout_height="wrap_content">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:text="사과"/>
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:text="딸기"/>
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:text="수박"/>
    </TableRow>
```

```
<TableRow
    android:layout_width="wrap_content"
    android:layout_height="wrap_content">
    <ImageView
        android:layout_width="80dp"
        android:layout_height="80dp"
        android:src="@drawable/apple"/>
    <ImageView
        android:layout_width="80dp"
        android:layout_height="80dp"
        android:src="@drawable/strawberry"/>
    <ImageView
        android:layout_width="80dp"
        android:layout_height="80dp"
        android:src="@drawable/watermelon"/>
</TableRow>
<TableRow
    android:layout_width="wrap_content"
    android:layout_height="wrap_content">
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="사과 선택"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="딸기 선택"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="수박 선택"/>
</TableRow>
```

```
<EditText  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:hint="일반 뷰는 한 행을 차지"/>  
</TableLayout>
```

```
package org.lklab.p103_tablelayout1;

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);
    }
}
```

❖ 그리드 레이아웃(GridLayout)

- 격자(grid) 내부의 셀에 자식 뷰를 배치하는 뷰 그룹
- 테이블 레이아웃과 비슷하지만, 각각의 자식 뷰가 자신의 위치와 차지하는 셀의 개수를 독립적으로 지정 가능
- 테이블 레이아웃과 달리 안드로이드 4.0부터 지원된다는 제약

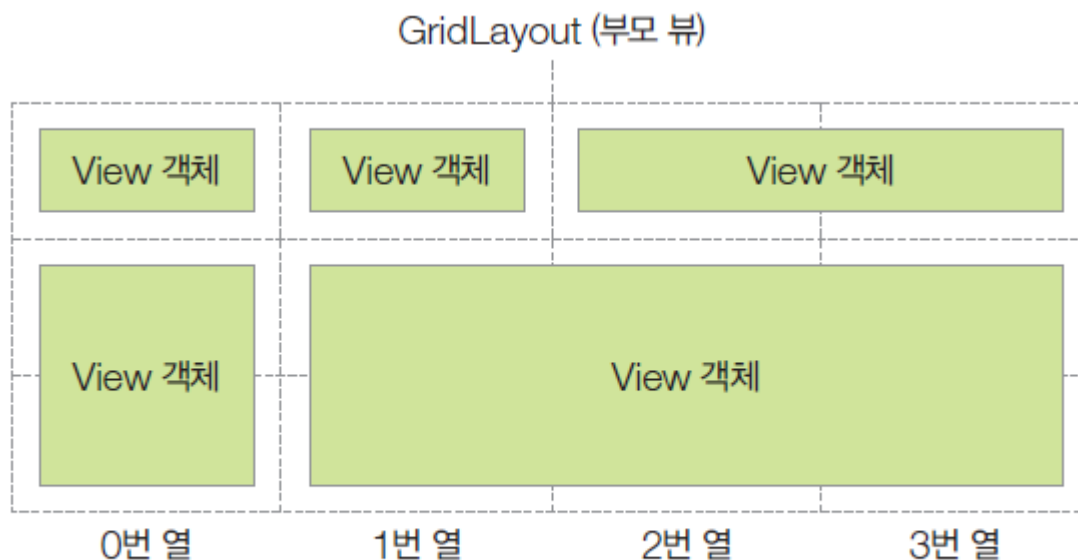
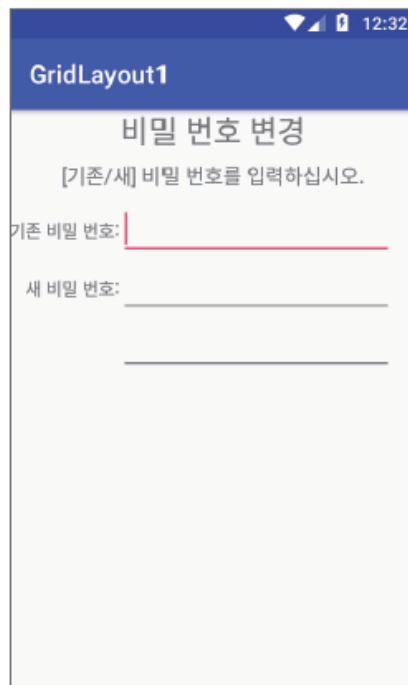


그림 2-21 그리드 레이아웃

❖ 그리드 레이아웃을 활용하여 비밀번호 변경 화면 구성하기



(a) 컴포넌트 트리



(b) 실행 화면

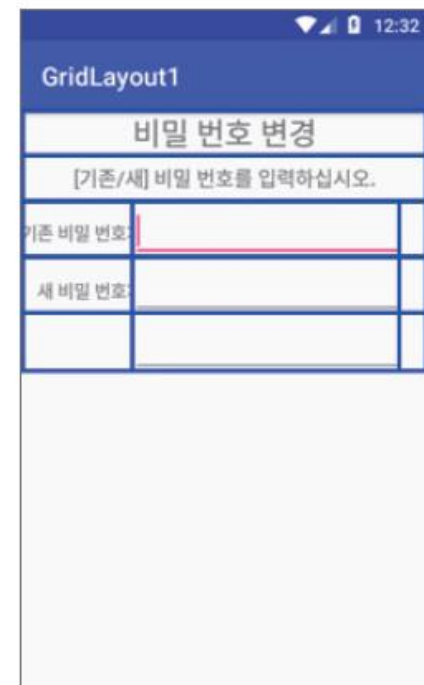
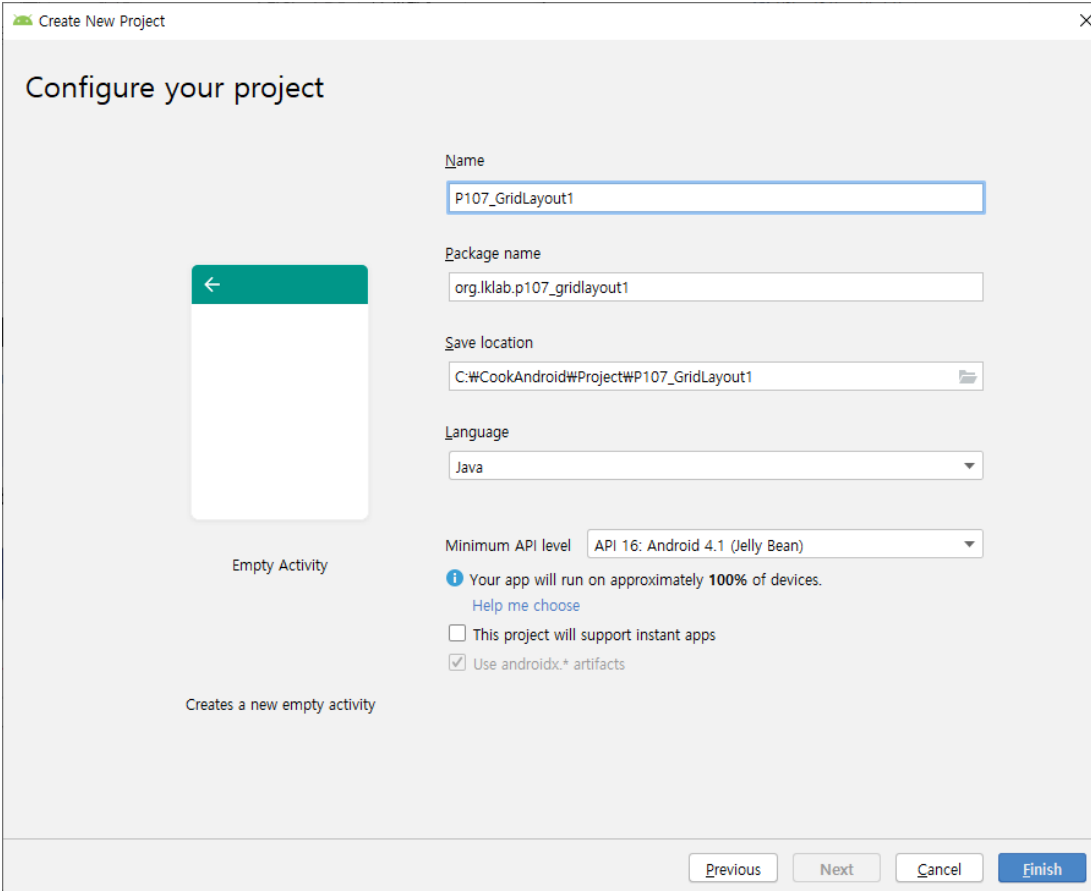


그림 2-23 격자 구조

그림 2-22 컴포넌트 트리와 실행 화면



Create New Project

Configure your project

Name
P107_GridLayout1

Package name
org.lklab.p107_gridlayout1

Save location
C:\\CookAndroid\\Project\\WP107_GridLayout1

Language
Java

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

Empty Activity

Creates a new empty activity

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

☐ This project will support instant apps

☒ Use androidx.* artifacts

Previous Next Cancel Finish

```
<?xml version="1.0" encoding="utf-8"?>
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:columnCount="3">
    <TextView
        android:layout_column="0"
        android:layout_columnSpan="3"
        android:layout_gravity="center_horizontal"
        android:layout_row="0"
        android:text="비밀 번호 변경"
        android:textSize="24sp"/>
    <TextView
        android:layout_column="0"
        android:layout_columnSpan="3"
        android:layout_gravity="center_horizontal"
        android:layout_margin="8dp"
        android:layout_row="1"
        android:text="[기존/새] 비밀번호를 입력하십시오."
        android:textSize="16sp"/>
    <TextView
        android:layout_gravity="right"
        android:text="기존 비밀번호:"/>
    <EditText
        android:ems="10"
        android:inputType="textPassword"/>
```

```
<TextView
    android:layout_column="0"
    android:layout_gravity="right"
    android:text="새 비밀번호:"/>
<EditText
    android:ems="10"
    android:inputType="textPassword"/>
<EditText
    android:layout_column="1"
    android:ems="10"
    android:inputType="textPassword"/>
</GridLayout>
```

```
package org.lklab.p107_gridlayout1;

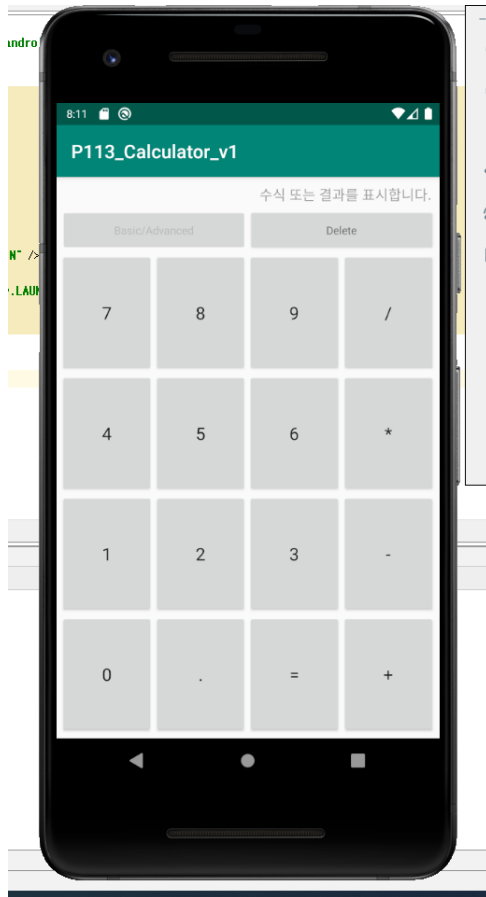
import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

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


❖ 계산기



Create New Project

Configure your project


Name
P113_Calculator_v1

Package name
org.lklab.p113_calculator_v1

Save location
C:\CookAndroid\Project\WP113_Calculator_v1

Language
Java

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

 Empty Activity

Creates a new empty activity

i 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"
    android:padding="4dp">
    <TextView
        android:id="@+id/textDisplay"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="right|center_vertical"
        android:hint="수식 또는 결과를 표시합니다."
        android:padding="4dp"
        android:textAppearance="?android:attr/textAppearanceMedium"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <Button
            android:id="@+id/btnMode"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="2"
            android:enabled="false"
            android:onClick="mOnClick"
            android:text="Basic/Advanced"
            android:textAppearance="?android:attr/textAppearanceSmall"/>
        <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>
```



```
<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"/>
</LinearLayout>
```

```
package org.lklab.p113_calculator_v1;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

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

public class MainActivity extends AppCompatActivity {

    private TextView mTextDisplay;

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

        mTextDisplay = (TextView) findViewById(R.id.textDisplay);
    }

    public void mOnClick(View v) {
        switch (v.getId()) {
            case R.id.btnMode:
                break;
            case R.id.btn0:
                break;
            case R.id.btn1:
                break;
            case R.id.btn2:
                break;
        }
    }
}
```

```
case R.id.btn3:
    break;
case R.id.btn4:
    break;
case R.id.btn5:
    break;
case R.id.btn6:
    break;
case R.id.btn7:
    break;
case R.id.btn8:
    break;
case R.id.btn9:
    break;
case R.id.btnDot:
    break;
case R.id.btnAdd:
    break;
case R.id.btnSub:
    break;
case R.id.btnMul:
    break;
case R.id.btnDiv:
    break;
case R.id.btnDel:
    break;
case R.id.btnEq:
    break;
}
Toast.makeText(this, ((Button) v).getText(), Toast.LENGTH_SHORT).show();
}
```

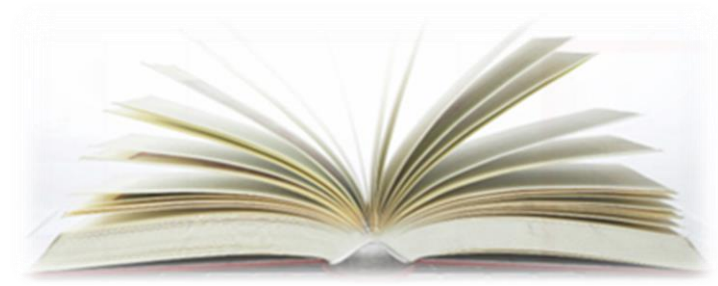
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="org.lklab.p113_calculator_v1">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

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