

CDA0125: Mobile Programming

Dong Hyun Kang (donghyun@changwon.ac.kr)

NOSlab (<https://noslab.github.io>)

Changwon National University

Create the second android program

- Android Project 사용법과 자동 완성 기능
- 화면 구성을 위한 XML 파일의 문법
- Java 코드 코딩 방법
 - activity_main.xml 파일에 포함된 위젯의 접근 방법
 - 위젯의 이벤트 발생 시 작동하는 코드 작성법
- R.java의 내용



그림 2-40 안드로이드 프로젝트 표준 틀 실행 결과

Create the second android program

- 1 안드로이드 프로젝트 생성
 - (1) [File] – [New Project] 선택 or [초기화면]의 [New Project] 클릭

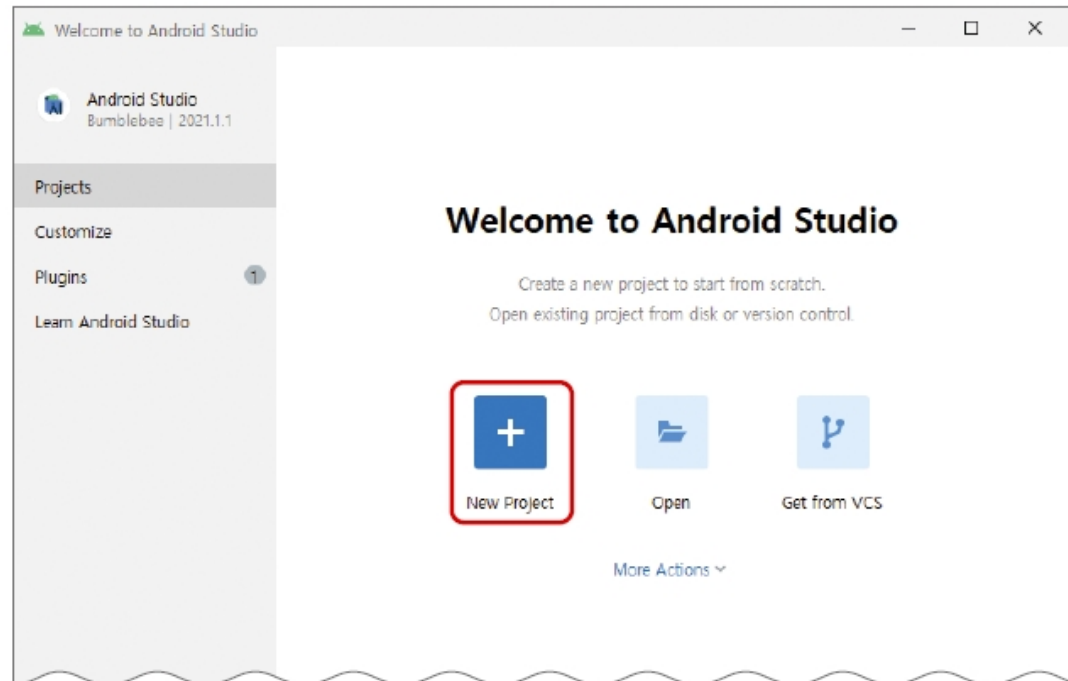


그림 2-41 새 프로젝트 생성

Create the second android program

- (2) [Template] – [Phone and Tablet] – [Empty Activity] 선택 – [Next] 클릭

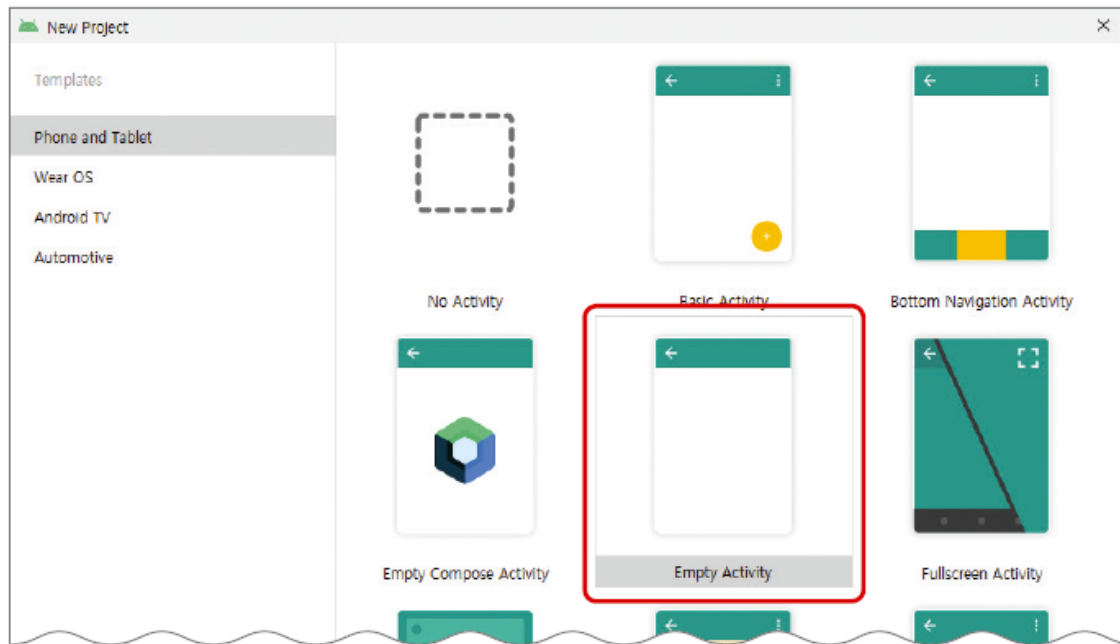


그림 2-42 프로젝트 형태 선택

Create the second android program

- (3) [Empty Activity]에서 정보 입력
 - Name만 입력해도 나머지 자동 완성

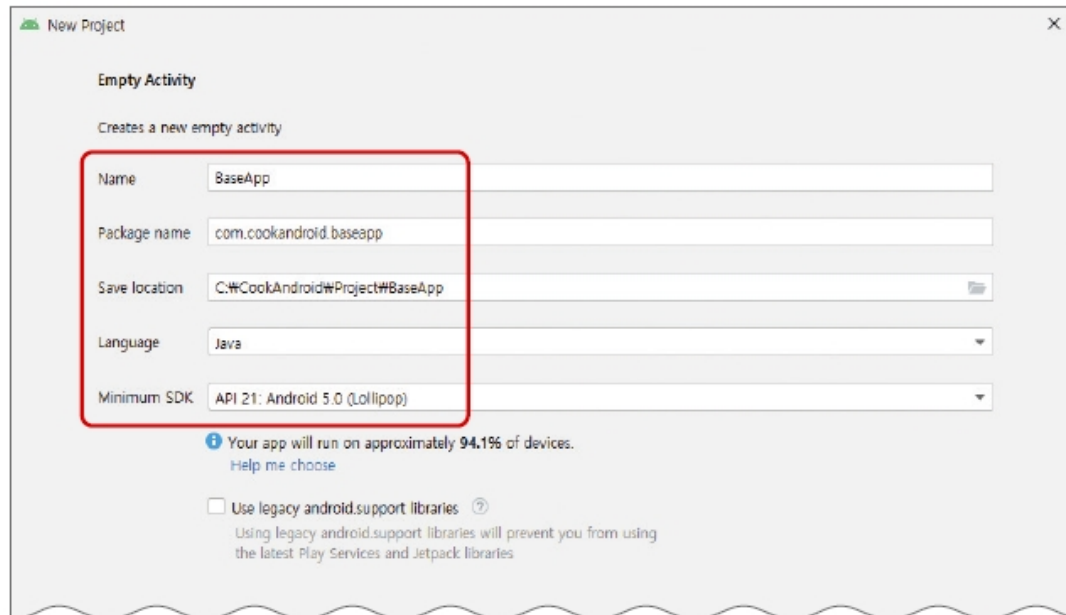


그림 2-43 프로젝트 환경 설정

Create the second android program

- 2 화면 디자인 및 편집
 - (1) Project Tree에서 [java]-[com.cookandroid.baseapp]-[MainActivity]가 기본적으로 열린 상태
 - (2) 상단의 [activity_main.xml] 클릭 – 오른쪽 위의 [Design] 탭 클릭

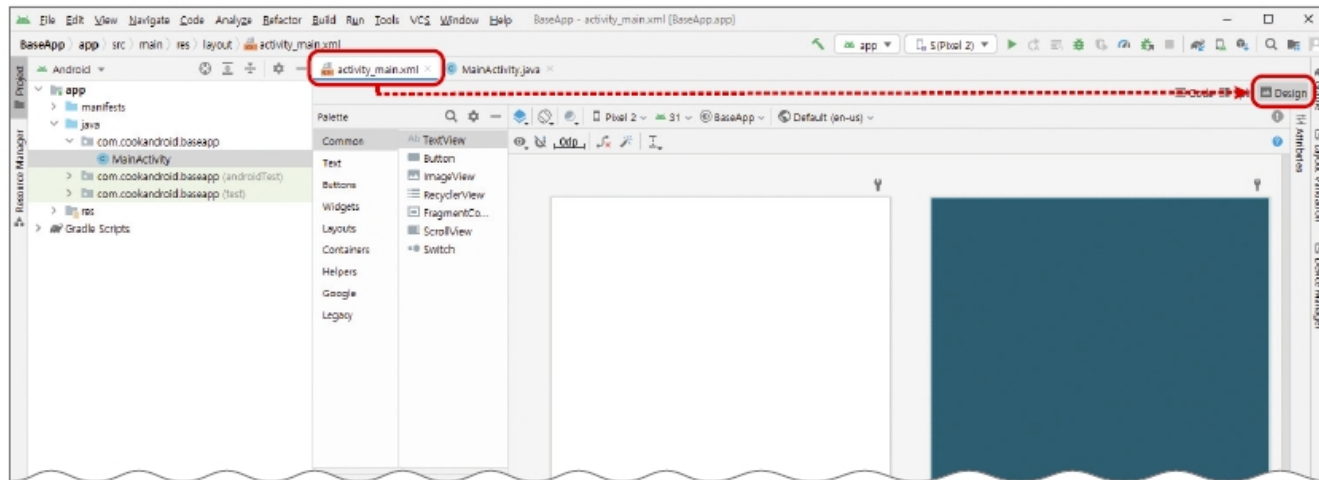


그림 2-44 activity_main.xml 확인

Create the second android program

- (3) 화면 오른쪽 위의 [Code] 탭 클릭 : XML 코드 확인
 - 주로 XML 코드로 편집
 - 프로젝트의 기본 레이아웃 : <LinearLayout>

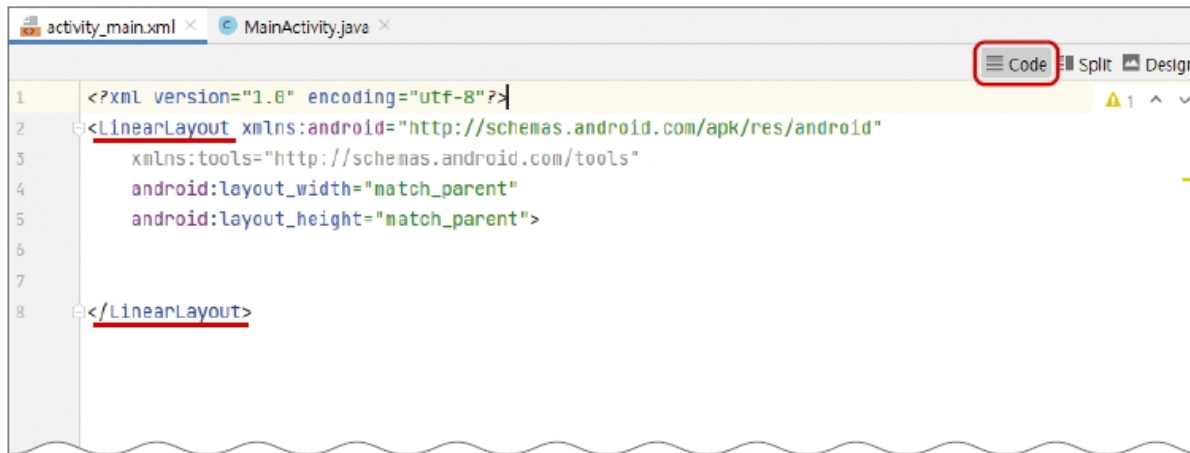
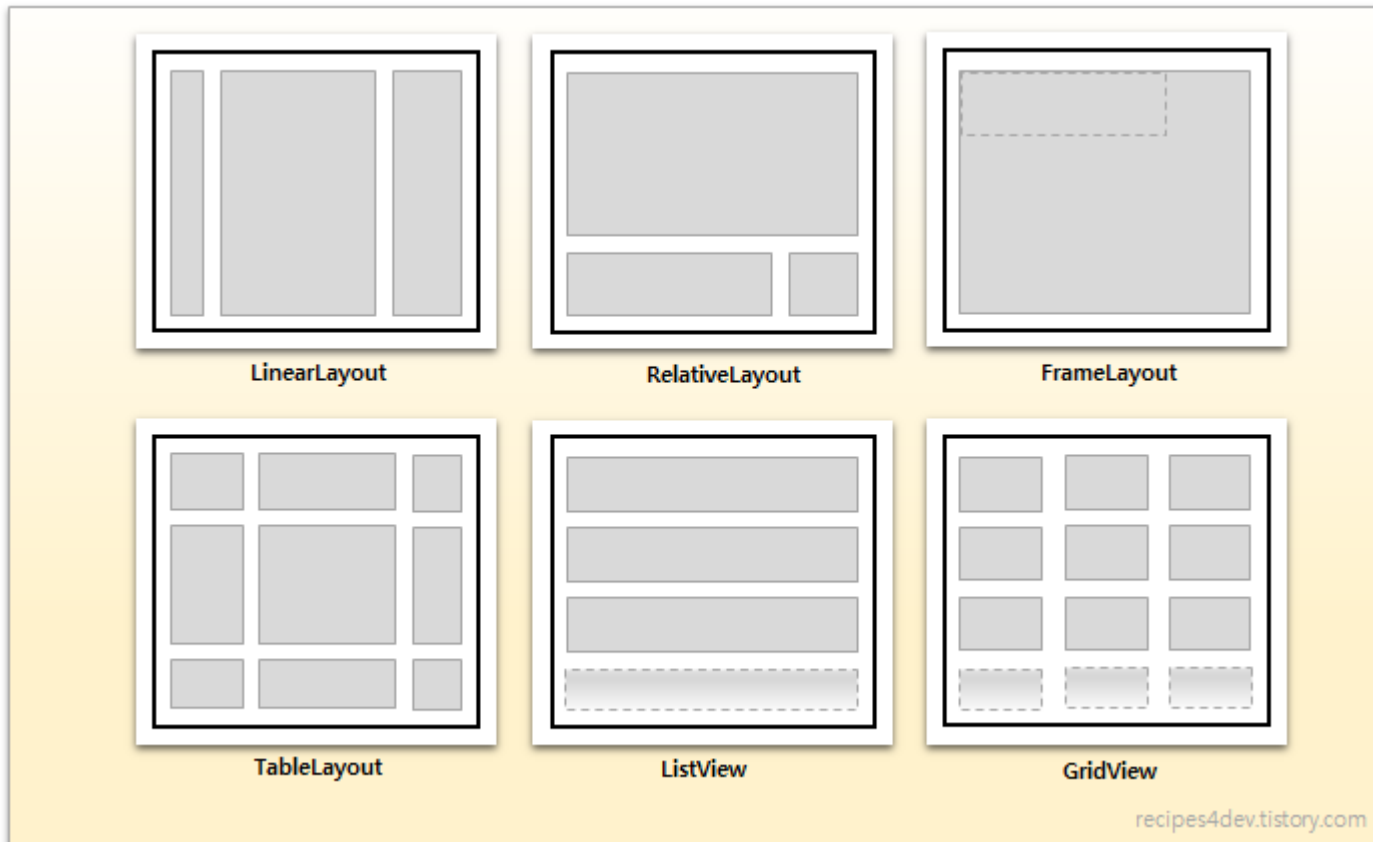


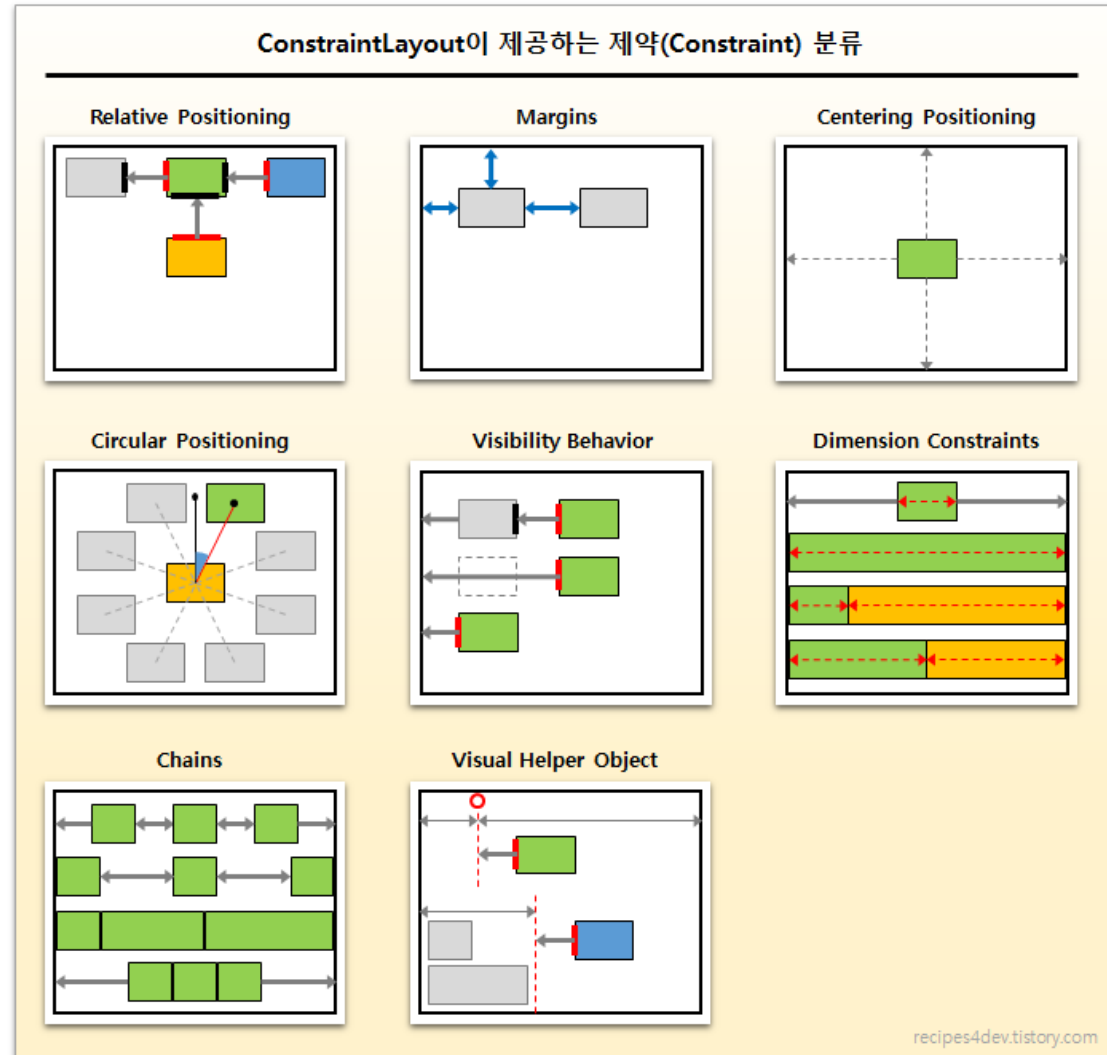
그림 2-45 XML 코드 변경

Layout의 종류



<https://recipies4dev.tistory.com/66>

Layout의 종류



Create the second android program

- (4) <LinearLayout> ... </LinearLayout> 사이에 버튼 추가
 - <Button> 입력

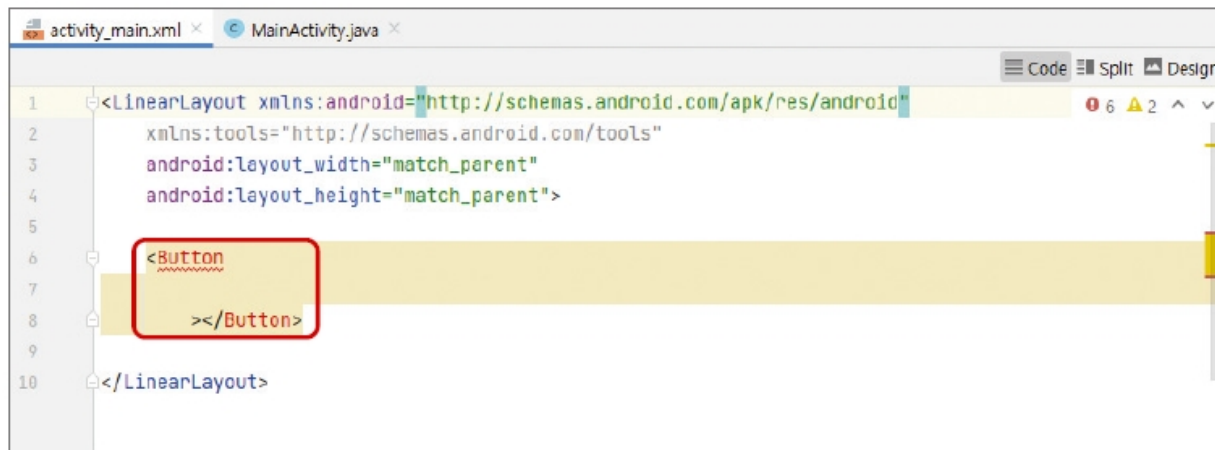


그림 2-46 버튼 추가

Create the second android program

- (5) “android:” 입력 – 목록에서 ‘layout_height’ 더블클릭

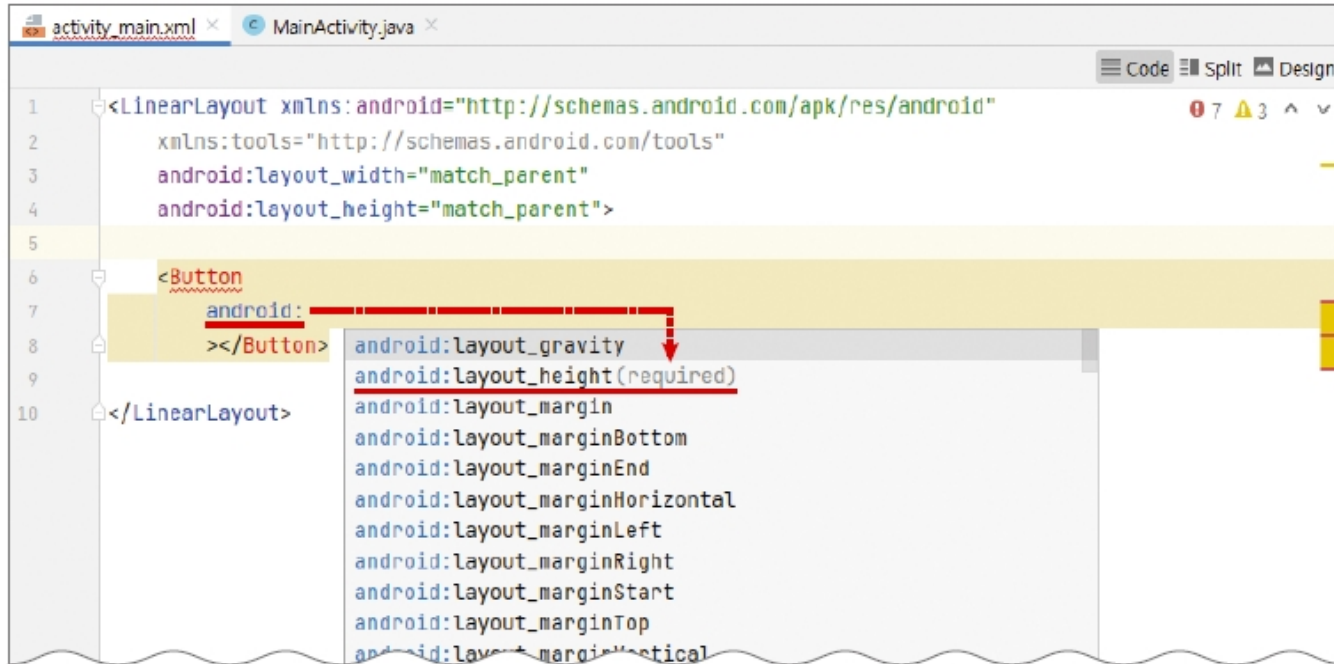


그림 2-47 버튼 속성 편집

Create the second android program

- (6) 자동 팝업 창 - 'wrap_content' 선택 - [Enter]
 - 자동 팝업 창이 뜨지 않으면, [Ctrl]-[Space Bar]

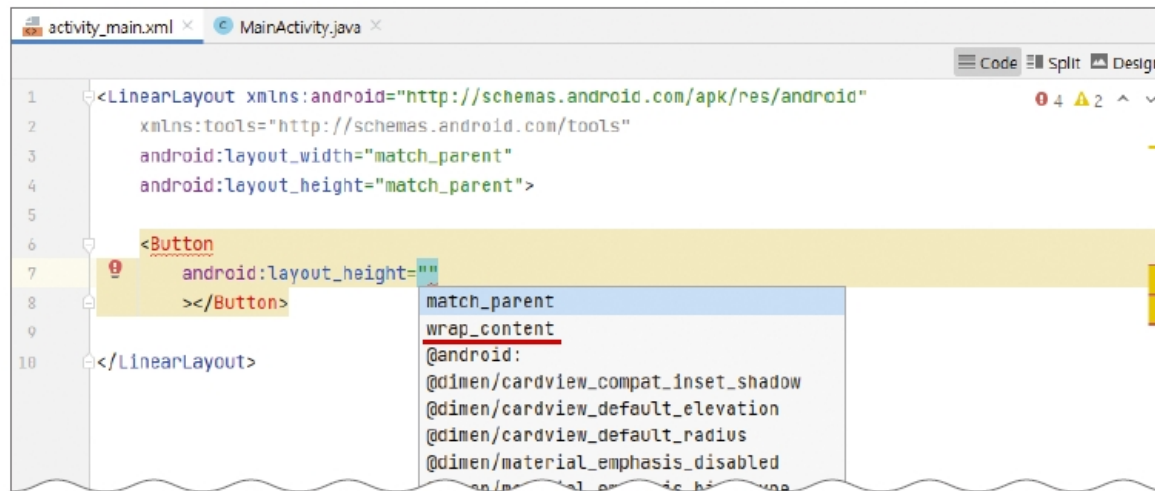


그림 2-48 버튼 속성 중 layout_height 편집

Create the second android program

- (7) 나머지 코드 입력하고 저장
 - 최종 코드 : [예제 2-1]

예제 2-1 activity_main.xml

```
1 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
2     xmlns:tools="http://schemas.android.com/tools"
3     android:layout_width="match_parent"
4     android:layout_height="match_parent" >
5
6
7     <Button
8         android:layout_width="match_parent"
9         android:layout_height="wrap_content"
10        android:id="@+id/button1"
11        android:text="@string/strBtn1"
12    >/Button>
13
14 </LinearLayout>
```

Create the second android program

- (8) activity_main.xml의 11행에 오류 발생
 - Project Tree의 [app]-[res]-[values]-[strings.xml] 더블클릭: xml코드 확인

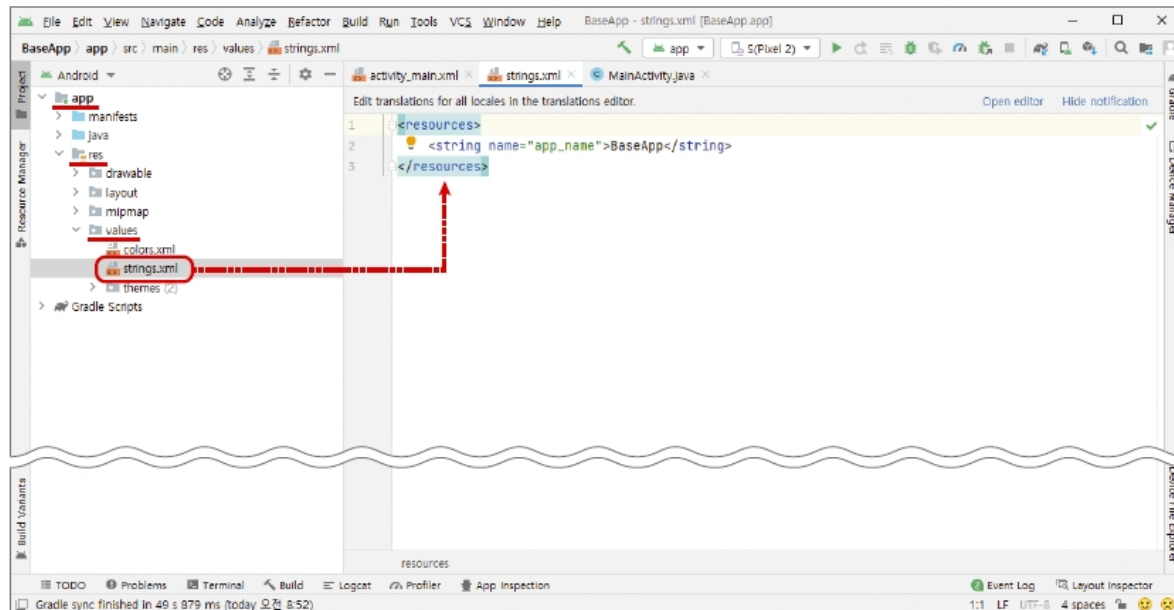


그림 2-49 strings.xml 파일

Create the second android program

- (9) strBtn1문자열 추가하고 저장

예제 2-2 strings.xml

```
1 <resources>
2     <string name="app_name">BaseApp</string>
3     <string name="strBtn1">버튼입니다</string>
4 </resources>
```

Create the second android program

- (10) activity_main.xml에서 오른쪽 위의 [Design] 탭 클릭 : 추가된 버튼 확인

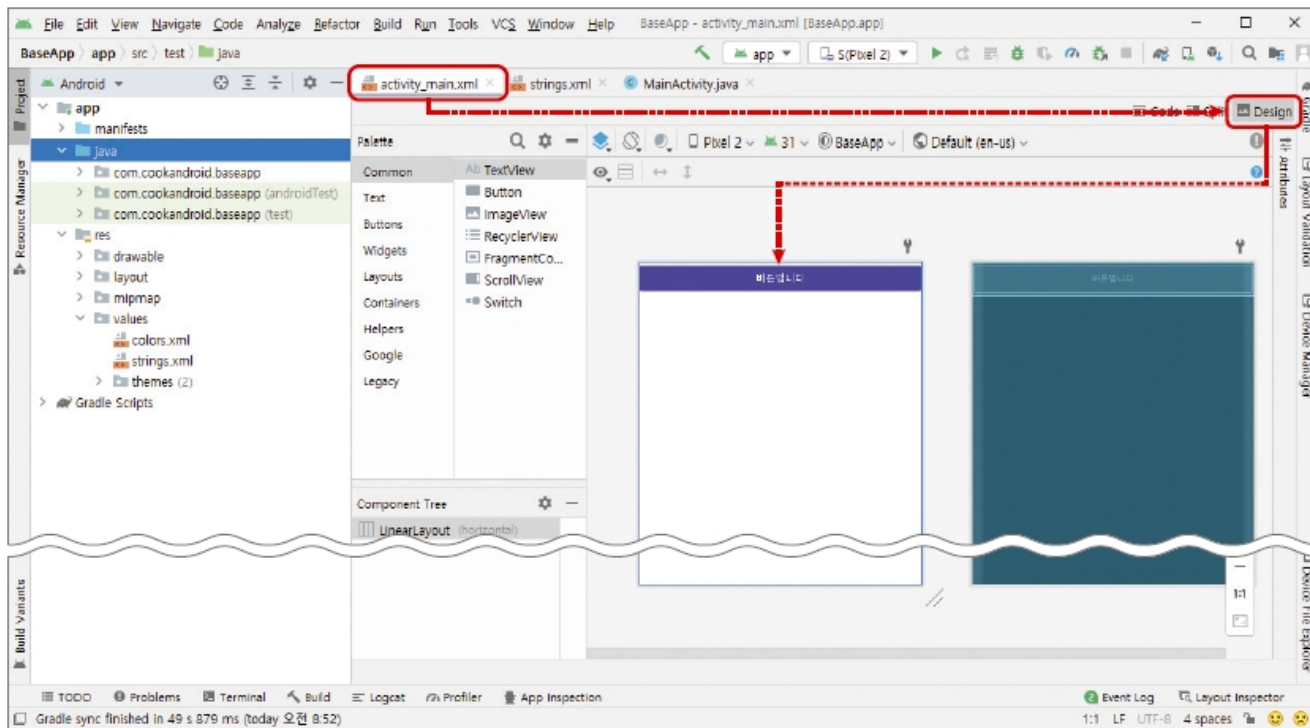


그림 2-50 activity_main.xml 코드의 그래픽 화면

Create the second android program

- (11) [Run] – [Run 'app'] 선택 혹은 [Run 'app'] 아이콘 클릭

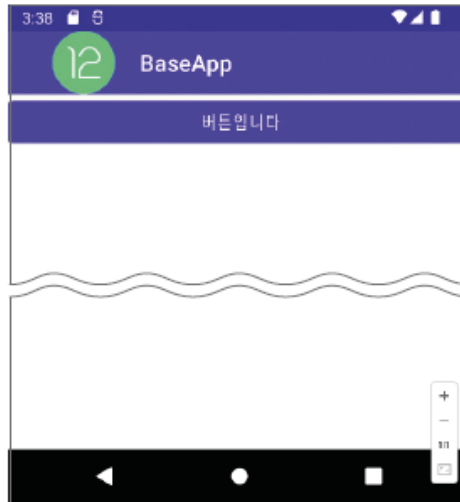


그림 2-52 프로젝트 실행

Create the second android program

- 3 Java 코드 작성 및 수정
 - (1) 버튼을 눌렀을 때 작동이 일어나도록 코드 작성
 - (2) Project Tree의 [java]-[패키지 이름] - [MainActivity] 더블클릭

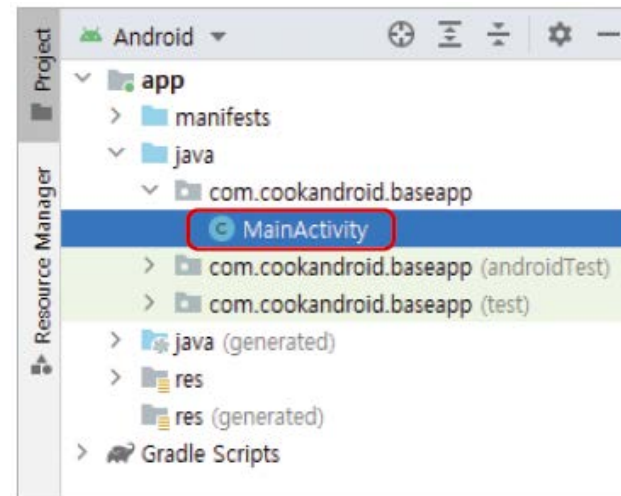


그림 2-53 Java 파일 선택

Create the second android program

- (3) 기본적인 코드 구성



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

그림 2-54 MainActivity.java의 기본 코드

Create the second android program

- (4) 3행의 import 앞에 작은 '+' 아이콘 클릭 : 행 확장
- (5) [ctrl]+[alt]+[o](알파벳 'o') : 불필요하게 импорт된 문장 제거



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

그림 2-55 필요 없는 import문 삭제

Create the second android program

- (6) Button 변수 추가
 - 멤버변수 위치인 @Override 바로 위에 다음을 입력

Button button1;



그림 2-56 Button 변수 추가

Create the second android program

- (7) **[Alt]+[Enter]** : Button과 관련된 클래스가 자동으로 import문에 추가



```
1 package com.cookandroid.baseapp;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.os.Bundle;
6 import android.widget.Button;
7
8 public class MainActivity extends AppCompatActivity {
9     Button button1;
10    @Override
11    protected void onCreate(Bundle savedInstanceState) {
12        super.onCreate(savedInstanceState);
13        setContentView(R.layout.activity_main);
14    }
15 }
```

그림 2-57 자동 import 확인

Create the second android program

- (8) activity_main.xml 파일에서 만든 객체에 접근 : findViewById() 메소드 사용
 - setContentView() 메소드 바로 아래에 다음 코드 추가

```
button1 = (Button) findViewById(R.id.button1);
```

- (9) 이벤트 메소드 정의
 - 'button1.setoncl'까지만 입력 – 팝업 창 맨 뒤의 'setOnClickListener()' 선택



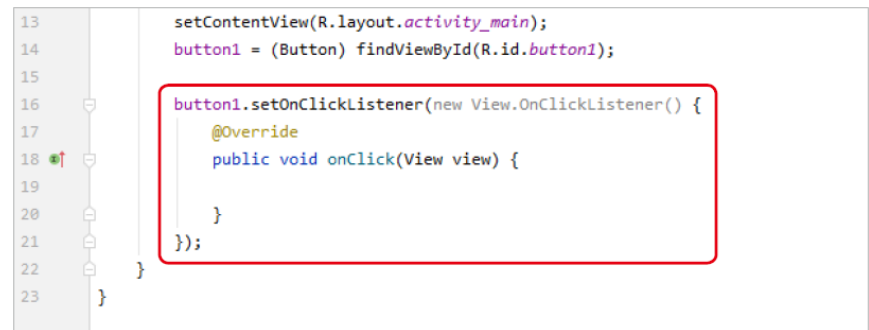
Create the second android program

- (10) `setOnClickListener()` 괄호 안에 "new View" 입력 - `OnClickListener()` 선택



그림 2-59 Java 코드 자동 완성 2

- (11) `onClick()` 메소드 자동 완성




Create the second android program

- (12) 버튼 클릭 시 작동하길 원하는 이벤트를 onClick() 메소드 안에 입력

예제 2-3 MainActivity.java

```
1 package com.cookandroid.baseapp;
2
3 import android.support.v7.app.AppCompatActivity;
4 import android.os.Bundle;
5 import android.view.View;
6 import android.widget.Button;
7 import android.widget.Toast;
8
9 public class MainActivity extends AppCompatActivity {
10     Button button1;
11
12     @Override
13     public void onCreate(Bundle savedInstanceState) {
14         super.onCreate(savedInstanceState);
15         setContentView(R.layout.activity_main);
16
17         button1 = (Button) findViewById(R.id.button1);
18
```

Anonymous inner class

```
18         button1.setOnClickListener( new View.OnClickListener() {
19             @Override
20             public void onClick(View v) {
21
22                 Toast.makeText(getApplicationContext(), "버튼을 눌렀어요",
23                     Toast.LENGTH_SHORT).show();
24             }
25         });
26 }
```

Create the second android program

- 4 프로젝트 실행 및 결과 확인
 - (1) [File]-[Save All] 선택 : 현재까지의 내용 모두 저장
 - (2) [Run]-[Run 'app']을 선택 혹은 [Run 'app'] 아이콘을 클릭 - 버튼 클릭

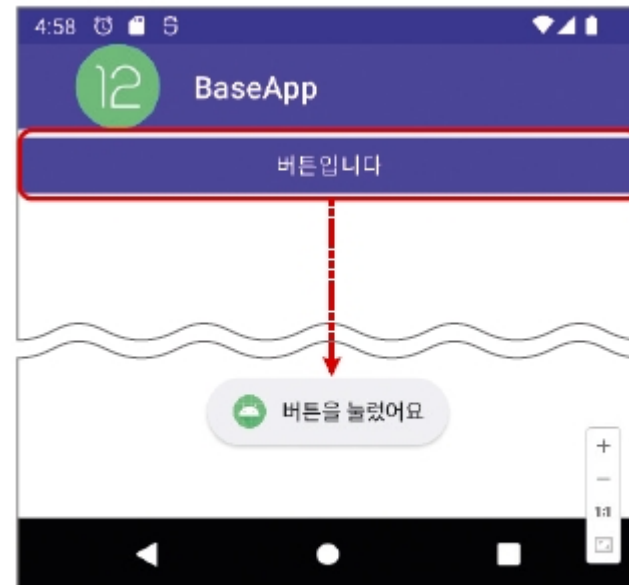


그림 2-61 실행 결과

Android Event

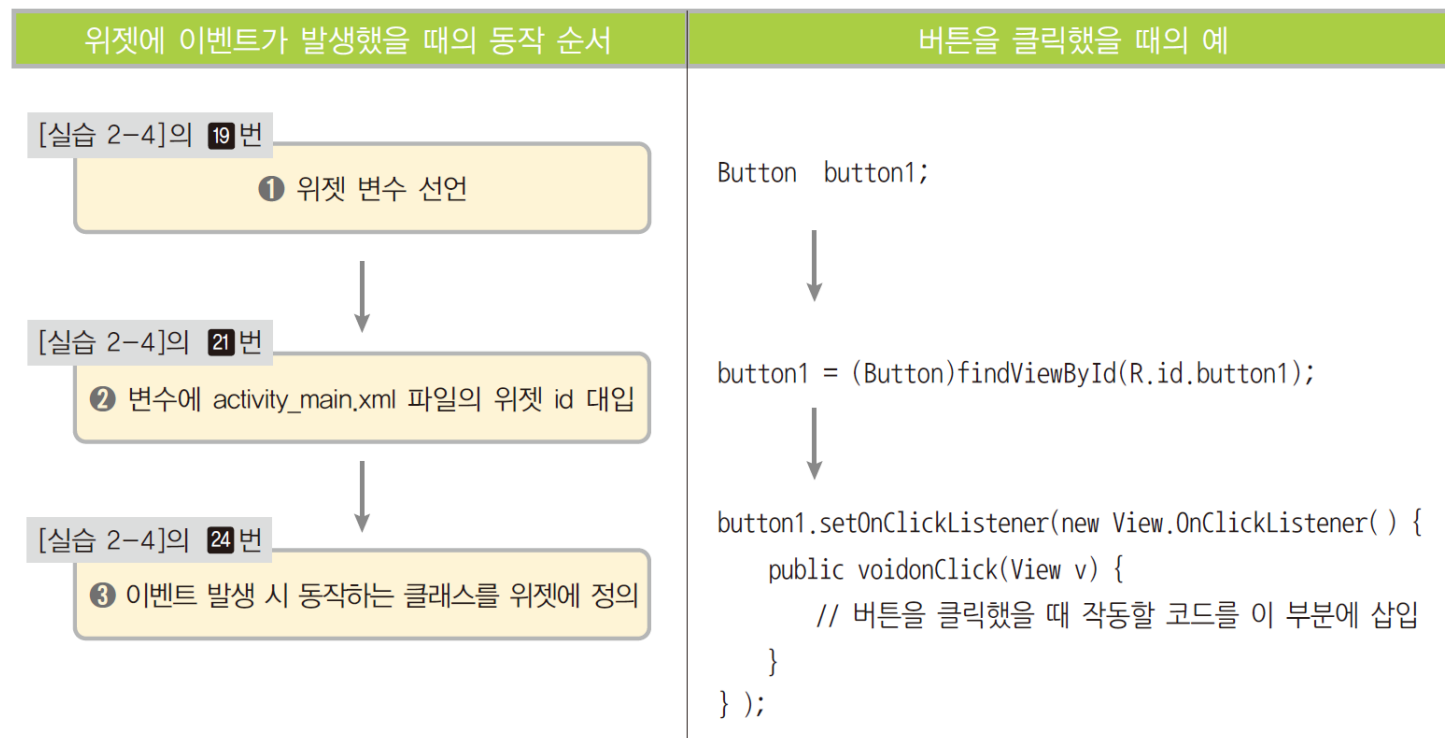


그림 2-63 위젯에 이벤트를 작동하기 위한 코딩 요약

Android Event

- 코드 응용
- activity_main.xml 파일에 체크박스를 추가하고 체크박스를 클릭할 때마다 작동하기

```
// ① 체크박스 변수 선언
CheckBox check1;

// ② 변수에 activity_main.xml 파일의 체크박스 대입
check1 = (CheckBox) findViewById(R.id.check1);

// ③ 체크박스를 클릭했을 때 작동하는 클래스 정의
check1.setOnClickListener( new View.OnClickListener() {
    public void onClick(View v) {
        // 체크박스를 클릭했을 때 작동할 코드를 이 부분에 삽입
    }
} );
```

Q & A