

Tinker board

IoT를 위한 최선의 엣지 단말기



MAKER SPACE
G·CAMP

Tinker Board 2S GPIO 활용

라이브러리 추가

깃허브 링크

Mraa library for android

The IO mapping can be found at: <https://github.com/TinkerBoard/TinkerBoard/wiki/User-Guide#gpio-config-table-for-tinker-board-2s>

Android Archive file for the IO interface of 40 pin on ASUS Tinker Board 2: [mraa-2.2.0.zip](#)

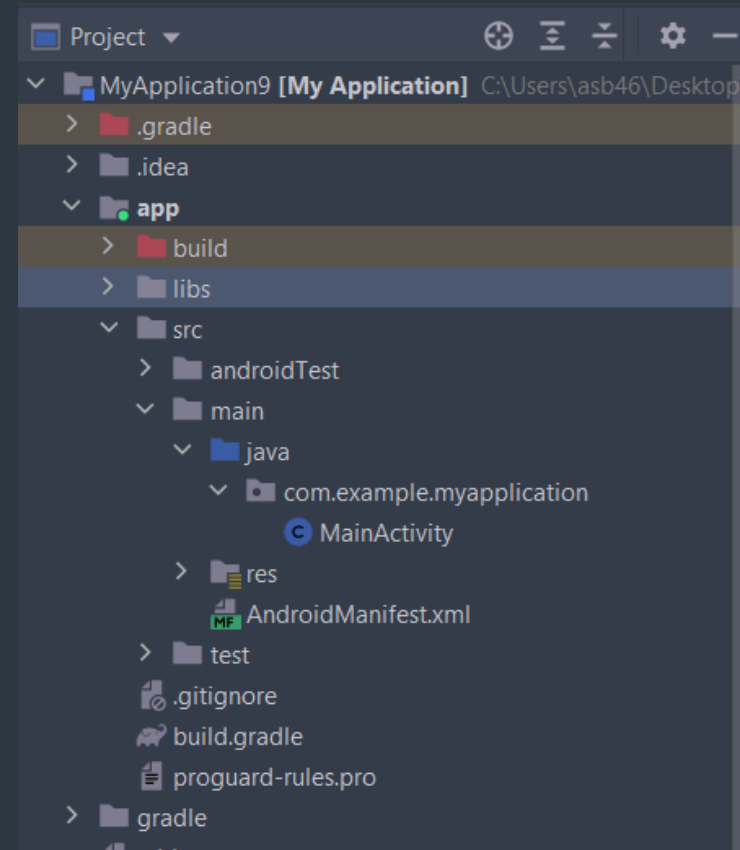
The Sample codes of Mraa API for Tinker Board 2: [Android-MraaDemo_tinkerboard2.zip](#)

The apk of Mraa API for Tinker Board 2: [Android-MraaDemo_tinkerboard2_APK.zip](#)

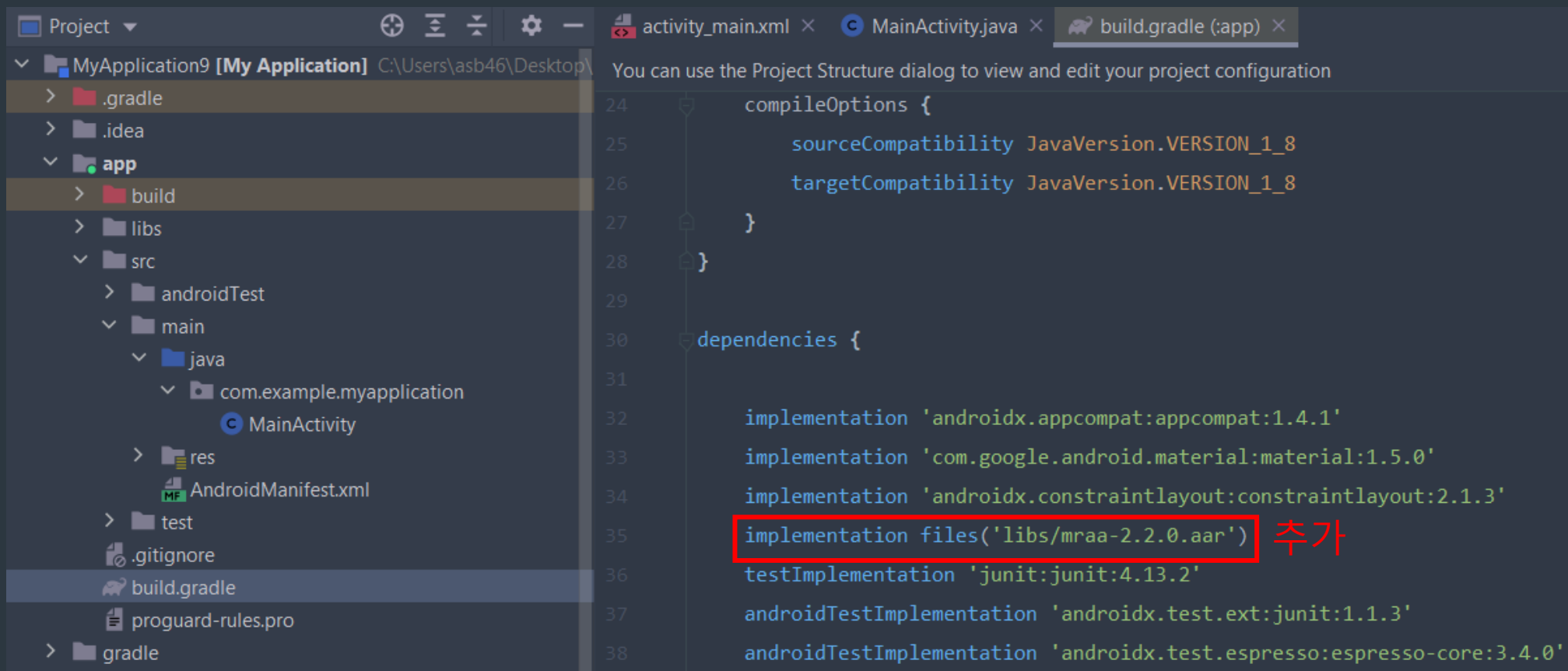
- Class

class	constructor	class	constructor
Gpio	Gpio(int pin_index)	Pwm	Pwm (int pin_index)
I2c	I2c (int i2c_index)	Uart	Uart(int uart_index)
Spi	Spi (int spi_index)	---	---

lib 디렉토리에 aar 라이브러리 파일 추가

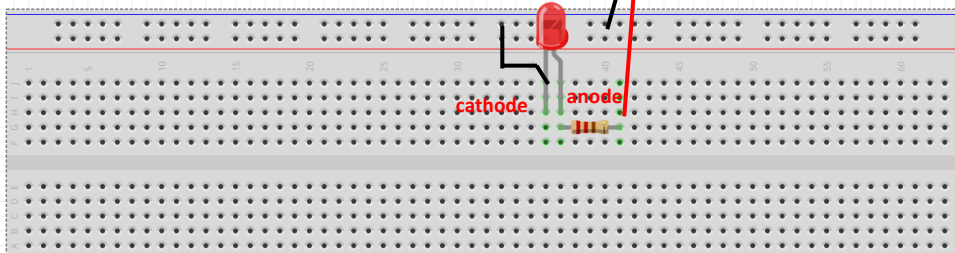
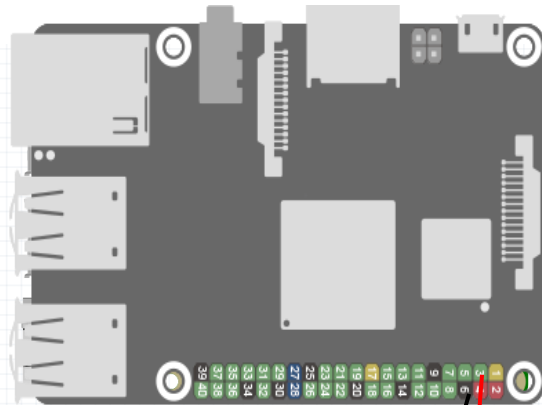


Gradle 파일 수정



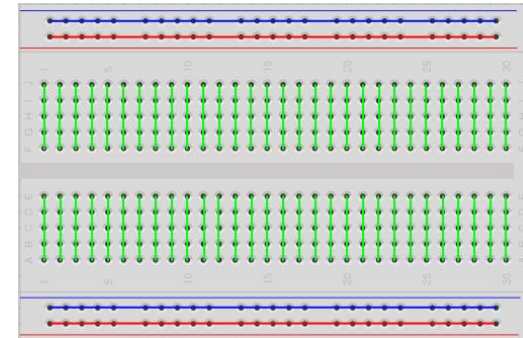
예제 1: LED 점등

- LED anode는 저항과 연결하고, 팅커보드의 3번핀에 연결
- LED cathode는 팅커보드의 6번핀(ground)와 연결



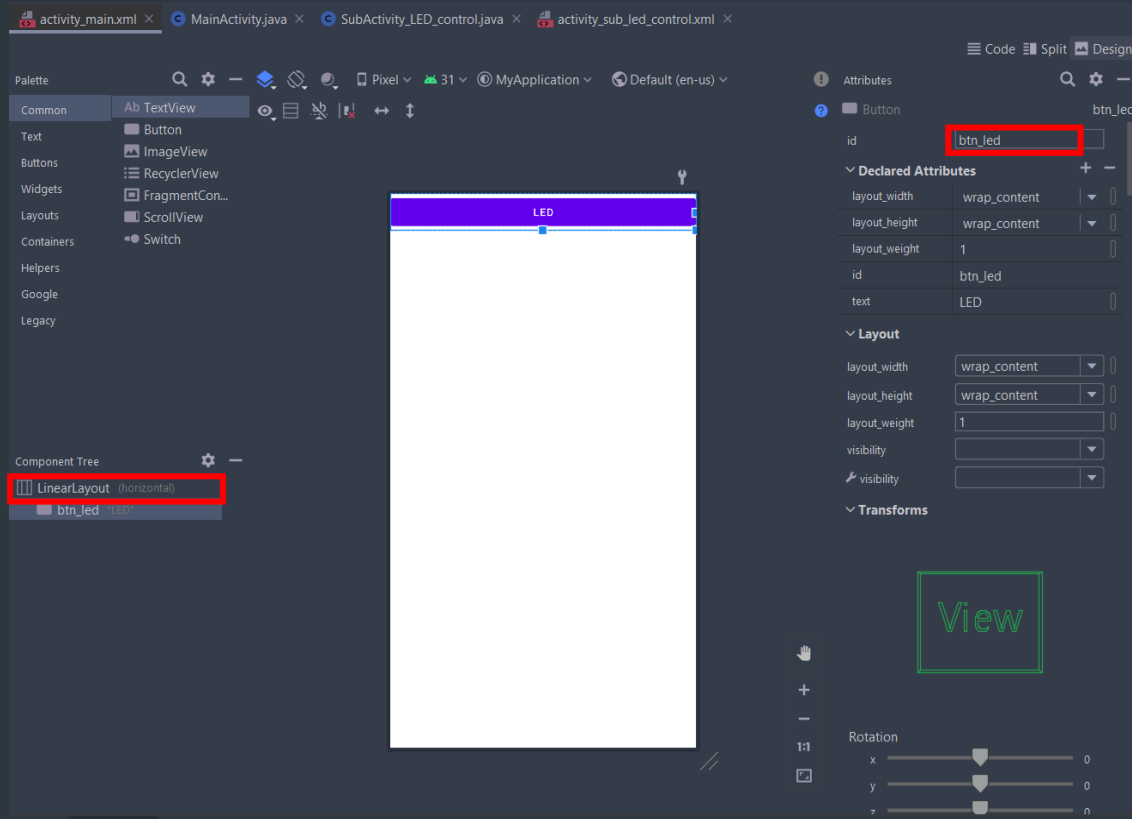
LED는 최대 20mA의 전류까지 버틸 수 있기 때문에 150ohm 이상의 저항을 추가하여 LED에 흐르는 전류를 낮춤

1	VCC3.3V_IO	2	VCC5V_SYS
3	GP8A4_I2C1_SDA	4	VCC5V_SYS
5	GP8A5_I2C1_SCL	6	GND
7	GP0C1_CLKOUT	8	GP5B1_UART1TX
9	GND	10	GP5B0_UART1RX
11	GP5B4_SPI0CLK_UART4CTSN	12	GP6A0_PCM/I2S_CLK
13	GP5B6_SPI0_TXD_UART4TX	14	GND
15	GP5B7_SPI0_RXD_UART4RX	16	GP5B2_UART1CTSN
17	VCC3.3V_IO	18	GP5B3_UART1RTSN
19	GP8B1_SPI2TXD	20	GND
21	GP8B0_SPI2RXD	22	GP5C3
23	GP8A6_SPI2CLK	24	GP8A7_SPI2CSN0
25	GND	26	GP8A3_SPI2CSN1
27	GP7C1_I2C4_SDA	28	GP7C2_I2C4_SCL
29	GP5B5_SPI0CSN0_UART4RTSN	30	GND
31	GP5C0_SPI0CSN1	32	GP7C7_UART2TX_PWM3
33	GP7C6_UART2RX_PWM2	34	GND
35	GP6A1_PCM/I2S_FS	36	GP7A7_UART3RX
37	GP7B0_UART3TX	38	GP6A3_PCM/I2S_SDI
39	GND	40	GP6A4_PCM/I2S_SDO

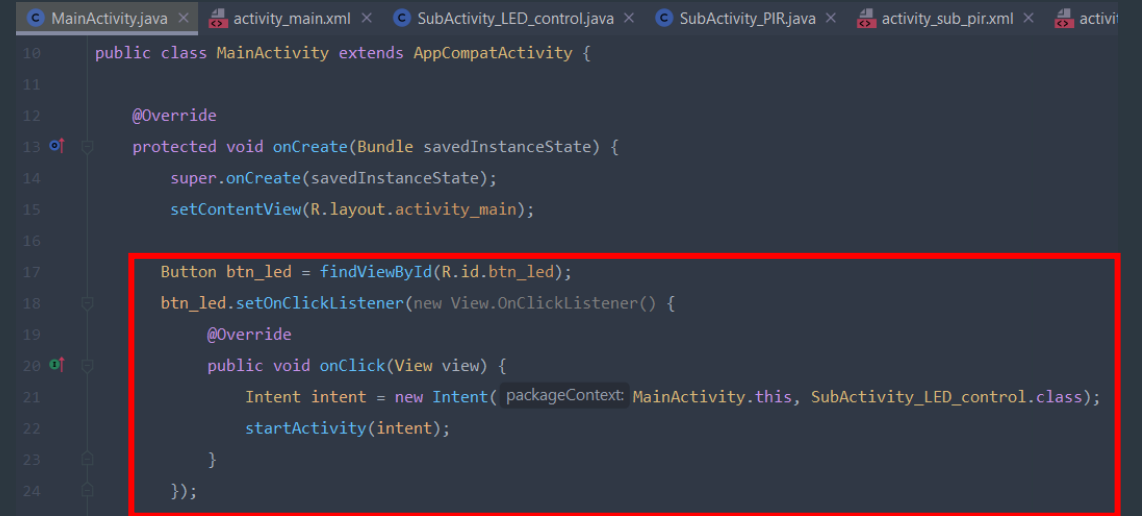


예제 1: LED 버튼 구현 & 함수 연동

1. layout 파일에서 id 값 변경



2. Button 객체 선언후 onClick 메소드 정의

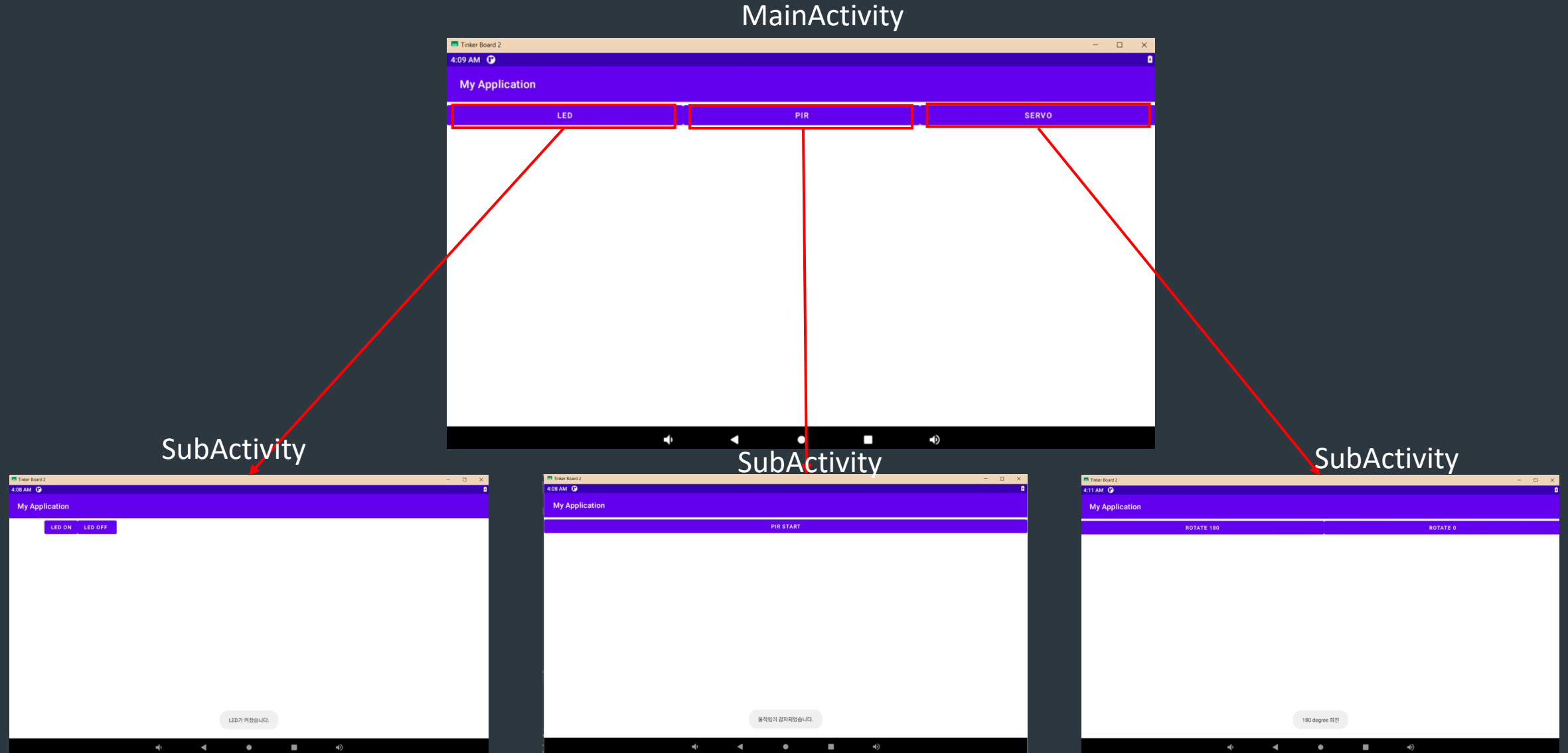


Intent 객체

- 다른 액티비티를 띄우거나 기능을 동작하기 위한 객체
- startActivity 메소드를 호출하여 다른 액티비티로 전환
- 다른 화면으로 이동 하거나 웹페이지를 열거나 다이얼 화면으로 전환하는 데 사용 가능

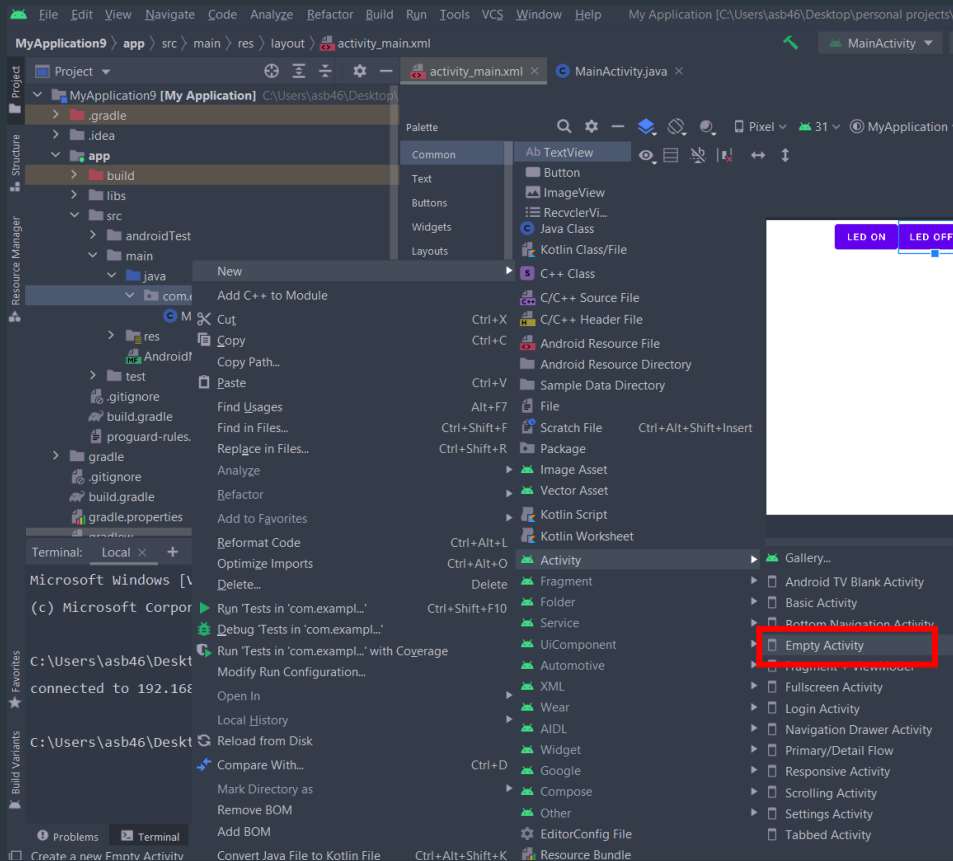
```
Button btn_led = findViewById(R.id.btn_led); layout 파일에서 id가 btn_led라는 이름을 가진 오브젝트를 find
btn_led.setOnClickListener(new View.OnClickListener() { View 클래스의 OnClickListener 인터페이스 (추상 클래스)의 onClick 함수를 재정의
    @Override
    public void onClick(View view) {
        Intent intent = new Intent( packageContext: MainActivity.this, SubActivity_LED_control.class); intent 객체 초기화
        startActivity(intent);
    }
});
```

최종 도안 미리보기

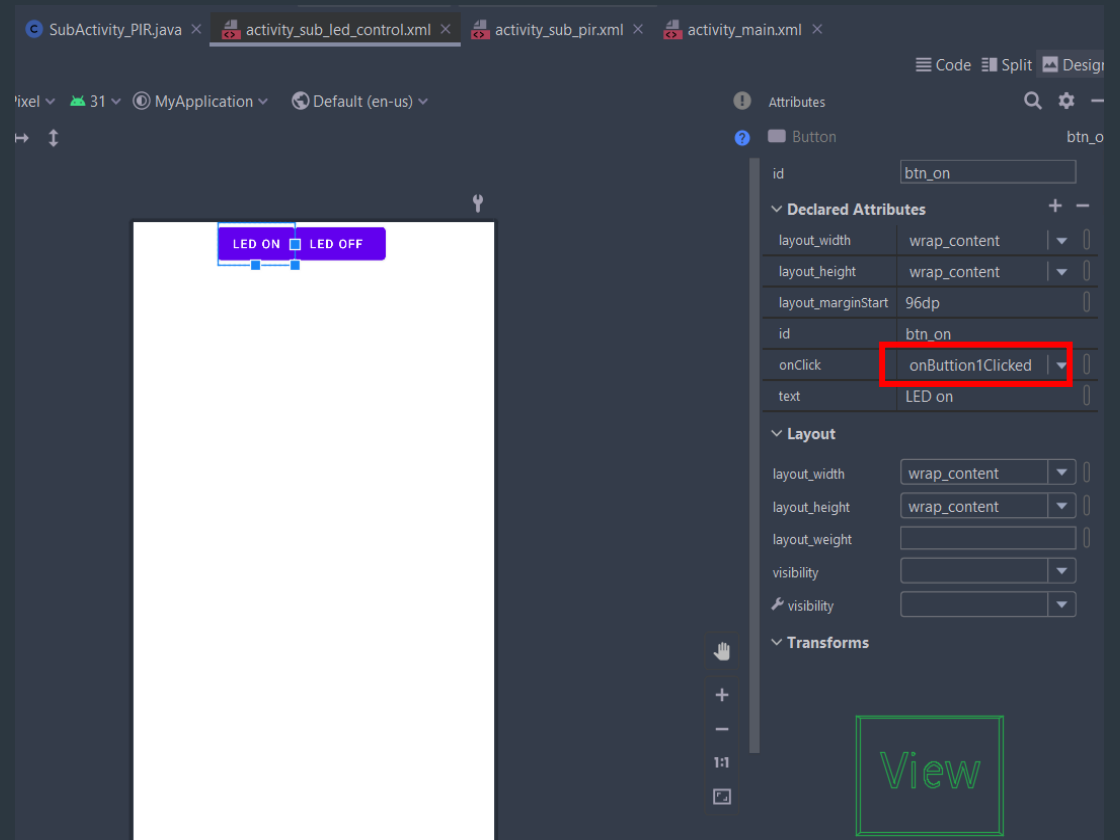


예제 1: SubActivity 생성

3. Empty Activity 생성



4. onClick 함수 사용



예제 1: SubActivity 작성

```
SubActivity_LED_control.java × MainActivity.java × activity_main.xml × activity_sub_led_control.xml ×
10 import mraa.*;
11
12 public class SubActivity_LED_control extends AppCompatActivity {
13
14     @Override
15     protected void onCreate(Bundle savedInstanceState) {
16         super.onCreate(savedInstanceState);
17         setContentView(R.layout.activity_sub_led_control);
18     }
19
20     public void onButton1Clicked(View v) {
21         Gpio gpio3 = new Gpio(TinkerBoard.TINKERBOARD_PIN3.swigValue());
22         gpio3.dir(Dir.DIR_OUT);
23         gpio3.write(1);
24         Toast.makeText(context: this, text: "LED가 켜졌습니다.", Toast.LENGTH_LONG).show();
25     }
26
27     public void onButton2Clicked(View v) {
28         Gpio gpio3 = new Gpio(TinkerBoard.TINKERBOARD_PIN3.swigValue());
29         gpio3.dir(Dir.DIR_OUT);
30         gpio3.write(0);
31         Toast.makeText(context: this, text: "LED가 꺼졌습니다.", Toast.LENGTH_LONG).show();
32     }
33 }
```

onClick 함수를 구현하는 2가지 방법

MainActivity에서 setOnClickListener를 통해 onClick 재정의

```
Button btn_led = findViewById(R.id.btn_led);
btn_led.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent( packageContext MainActivity.this, SubActivity_LED_control.class);
        startActivity(intent);
    }
});
```

MainActivity.java의 onCreate 안에서 정의

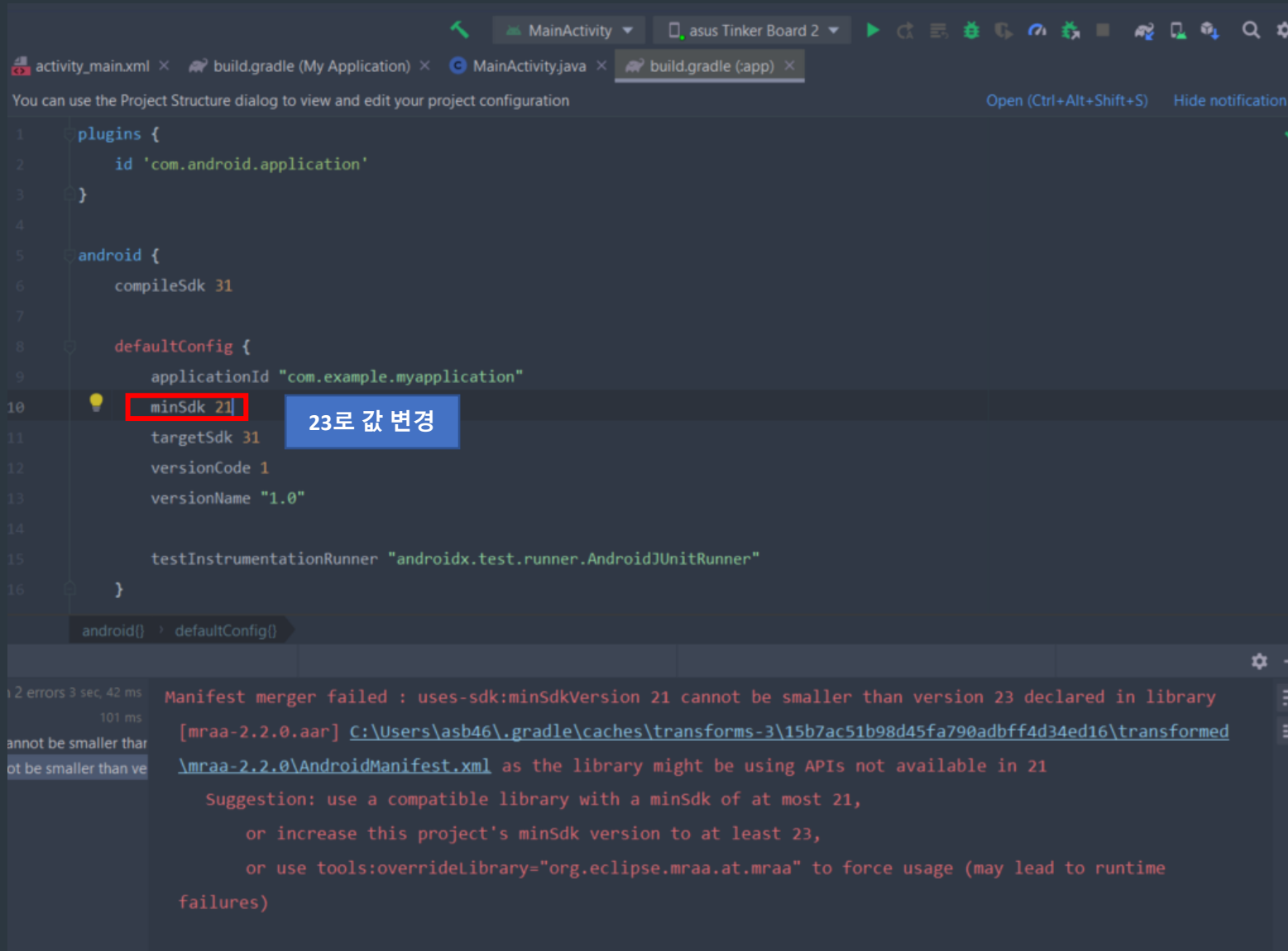
XML에서 Button의 onClick 속성에 함수 선언

```
<Button
    android:id="@+id/btn_on"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="96dp"
    android:onClick="onButton1Clicked"
    android:text="LED on" />

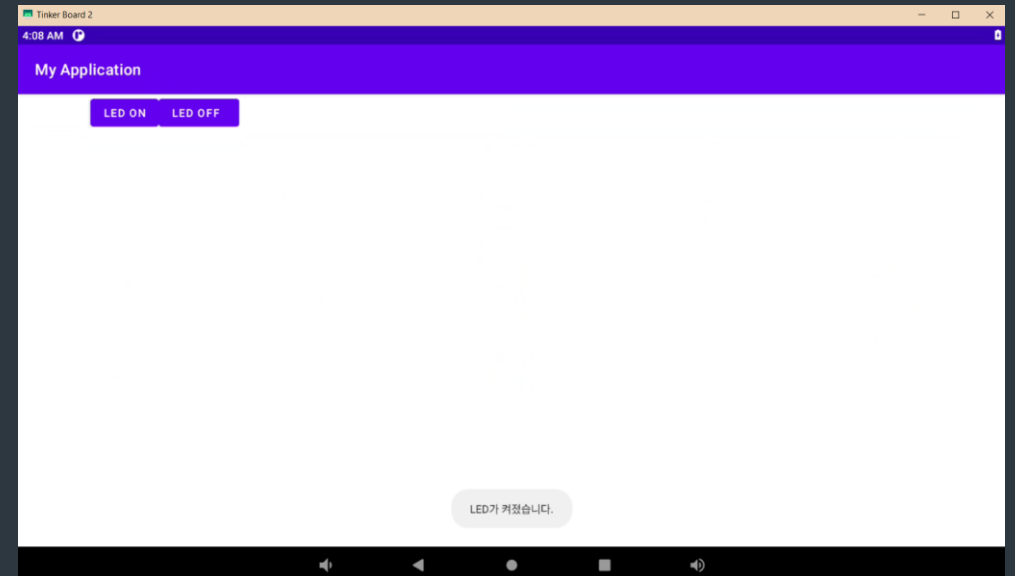
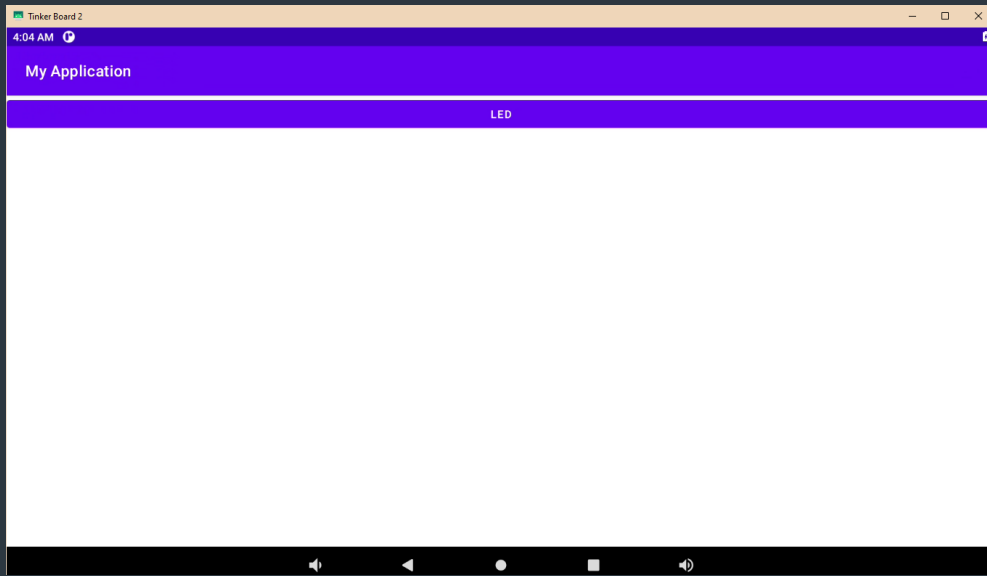
public void onButton1Clicked(View v) {
    Gpio gpio3 = new Gpio(TinkerBoard.TINKERBOARD_PIN3.swigValue());
    gpio3.dir(Dir.DIR_OUT);
    gpio3.write( 1);
    Toast.makeText( context this, text "LED가 켜졌습니다.", Toast.LENGTH_LONG).show();
}
```

MainActivity.java의 onCreate 밖에서 정의

예제 1: minSDK 버전 변경

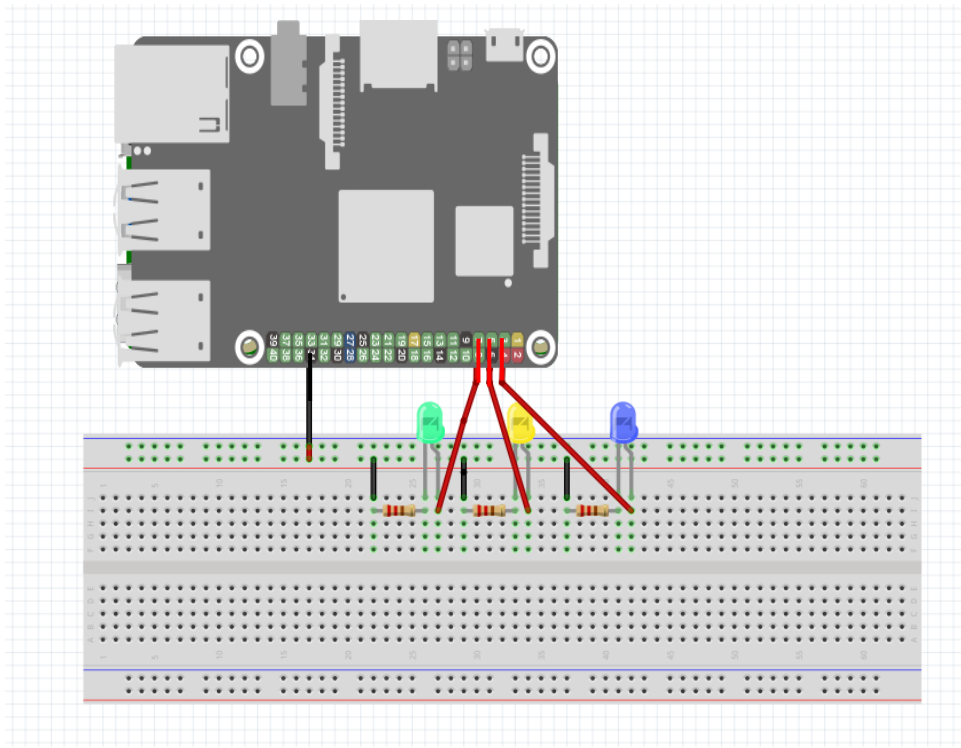


예제 1: 실행화면



예제2: LED 3개를 점등하는 예제

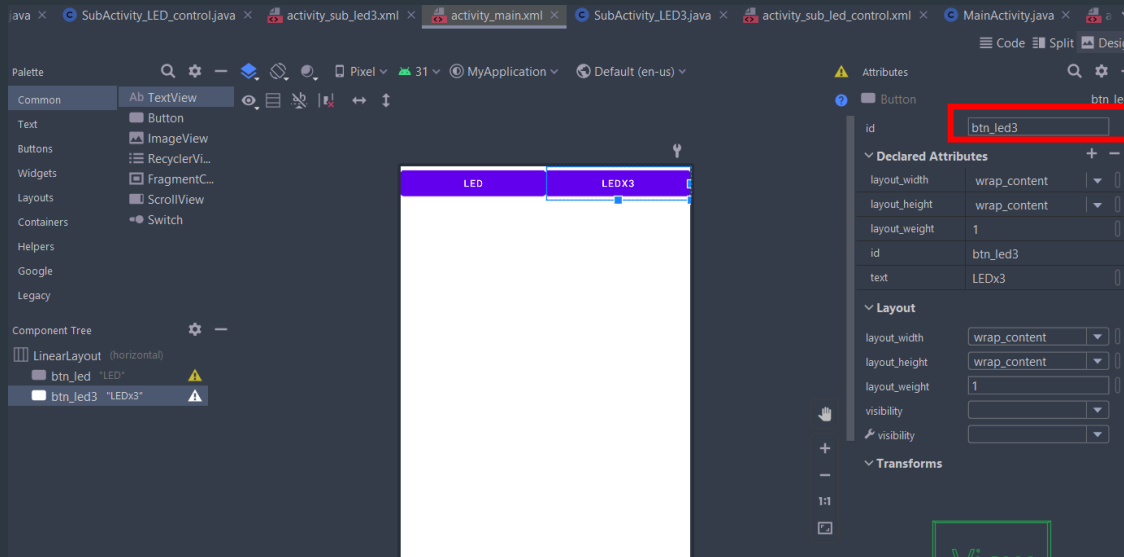
- Blue LED 3핀 연결
- Yellow LED 5핀 연결
- Green LED 7핀 연결
- GND 34핀 연결



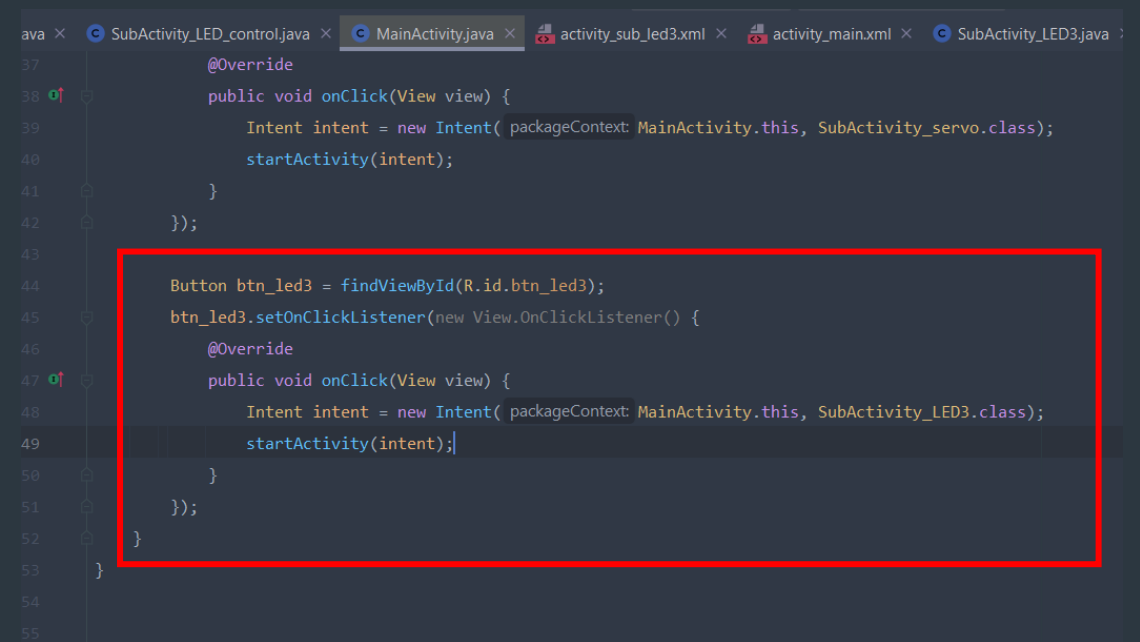
1	VCC3.3V_IO	2	VCC5V_SYS
3	GP8A4_I2C1_SDA	4	VCC5V_SYS
5	GP8A5_I2C1_SCL	6	GND
7	GP0C1_CLKOUT	8	GP5B1_UART1TX
9	GND	10	GP5B0_UART1RX
11	GP5B4_SPI0CLK_UART4CTSN	12	GP6A0_PCM/I2S_CLK
13	GP5B6_SPI0_TXD_UART4TX	14	GND
15	GP5B7_SPI0_RXD_UART4RX	16	GP5B2_UART1CTSN
17	VCC33_IO	18	GP5B3_UART1RTSN
19	GP8B1_SPI2TXD	20	GND
21	GP8B0_SPI2RXD	22	GP5C3
23	GP8A6_SPI2CLK	24	GP8A7_SPI2CSN0
25	GND	26	GP8A3_SPI2CSN1
27	GP7C1_I2C4_SDA	28	GP7C2_I2C4_SCL
29	GP5B5_SPI0CSN0_UART4RTSN	30	GND
31	GP5C0_SPI0CSN1	32	GP7C7_UART2TX_PWM3
33	GP7C6_UART2RX_PWM2	34	GND
35	GP6A1_PCM/I2S_FS	36	GP7A7_UART3RX
37	GP7B0_UART3TX	38	GP6A3_PCM/I2S_SDI
39	GND	40	GP6A4_PCM/I2S_SDO

예제2: LED 3개를 점등하는 예제

1. subActivity 생성 후 MainActivity에 LED X 3 버튼 생성

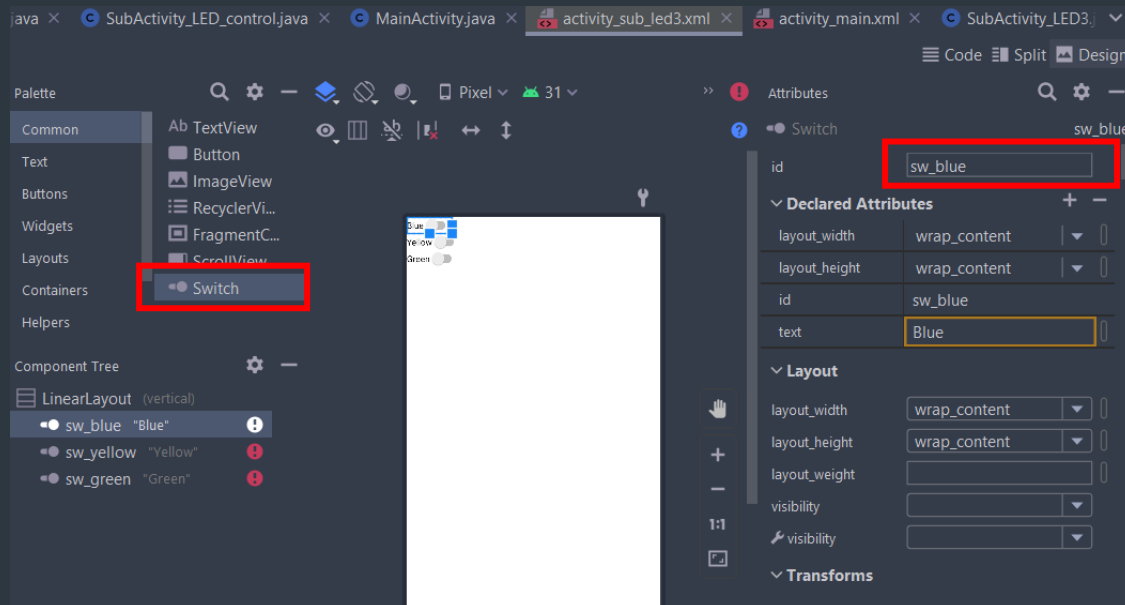


2. MainActivity에서 subActivity로 이동하는 onClick함수 정의

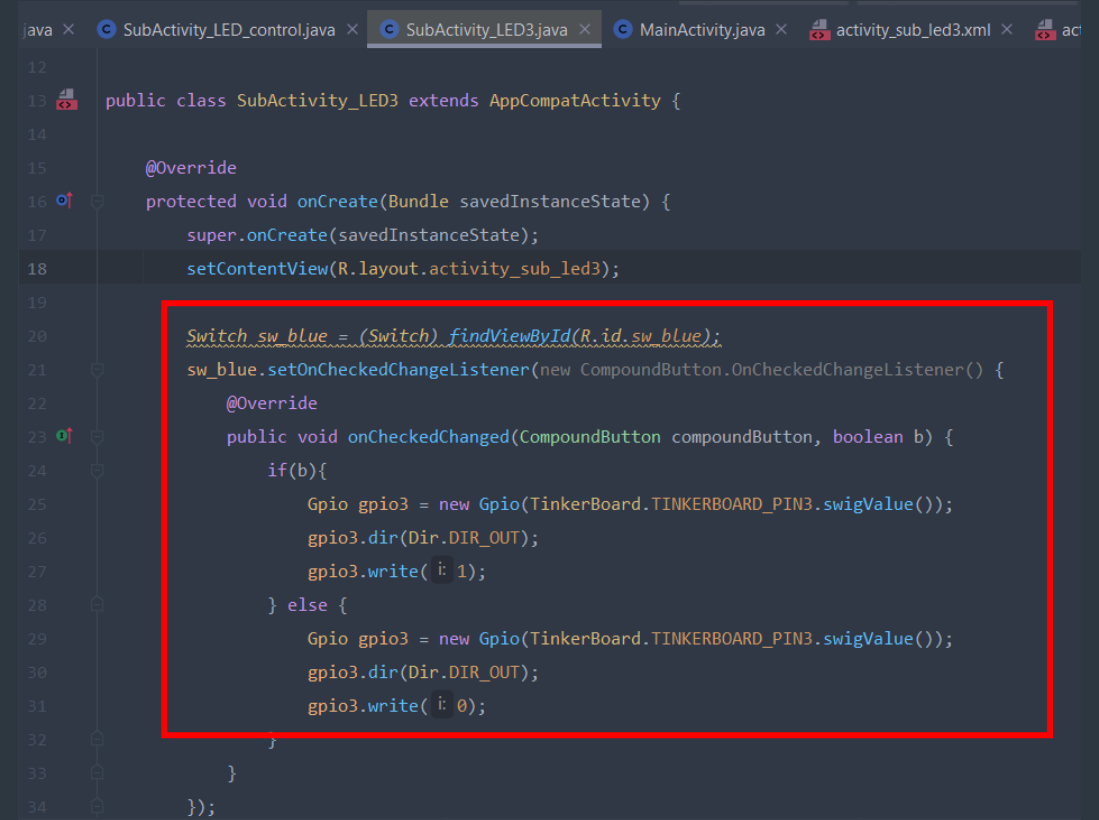


예제 2: SubActivity layout과 소스코드 작성

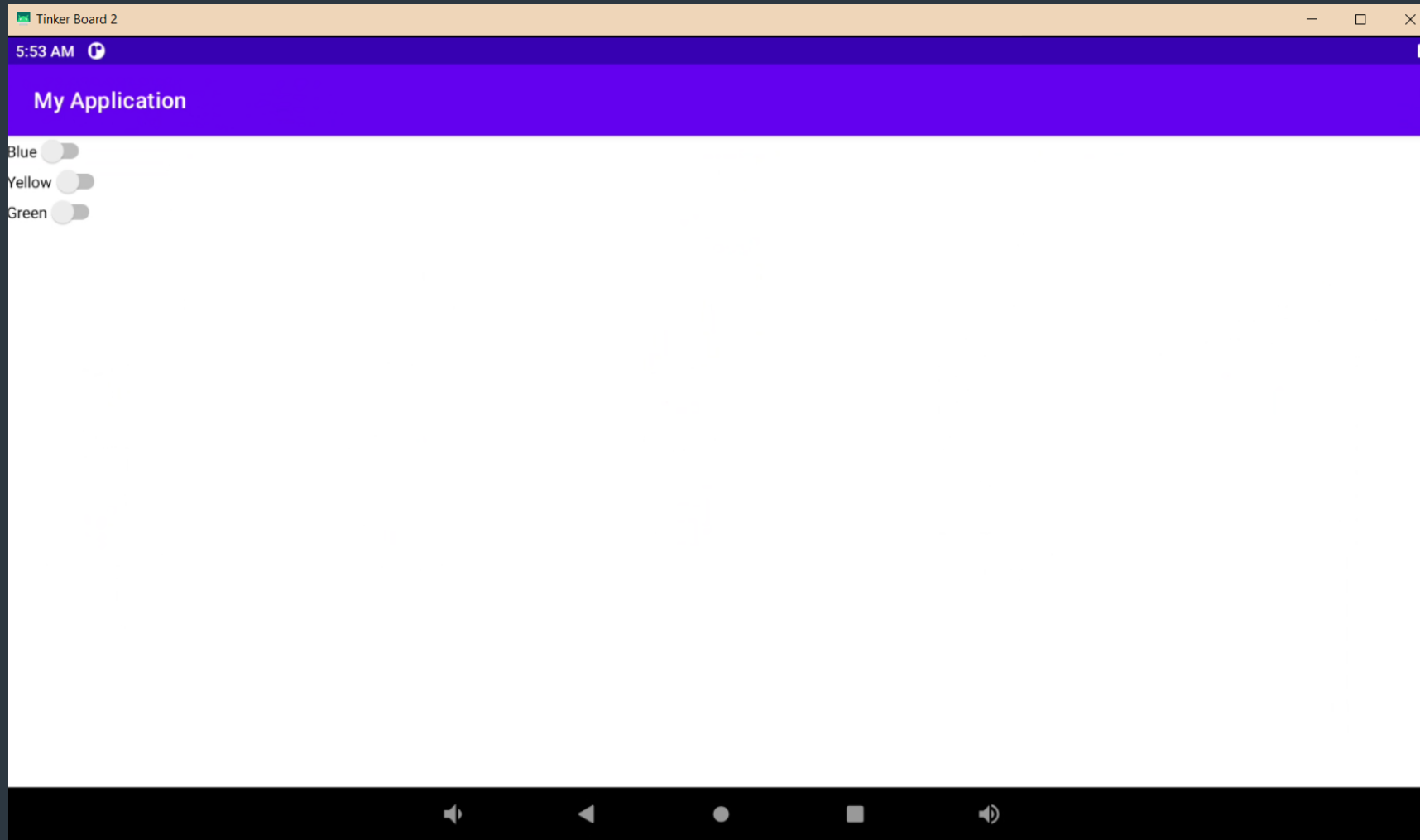
3. subActivity의 layout에서 Switch 배치 후 id 값 설정



4. SubActivity에서 Switch on off 함수 LED 별로 구현

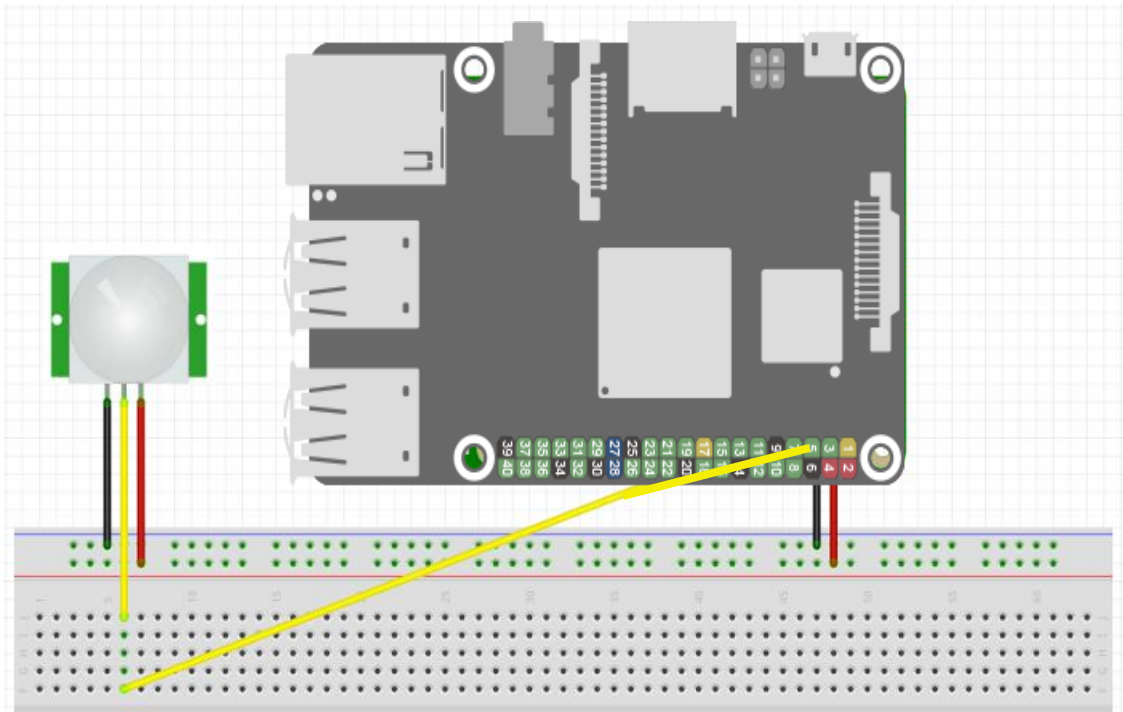


예제 2: 실행화면



예제3: PIR 모션 센서

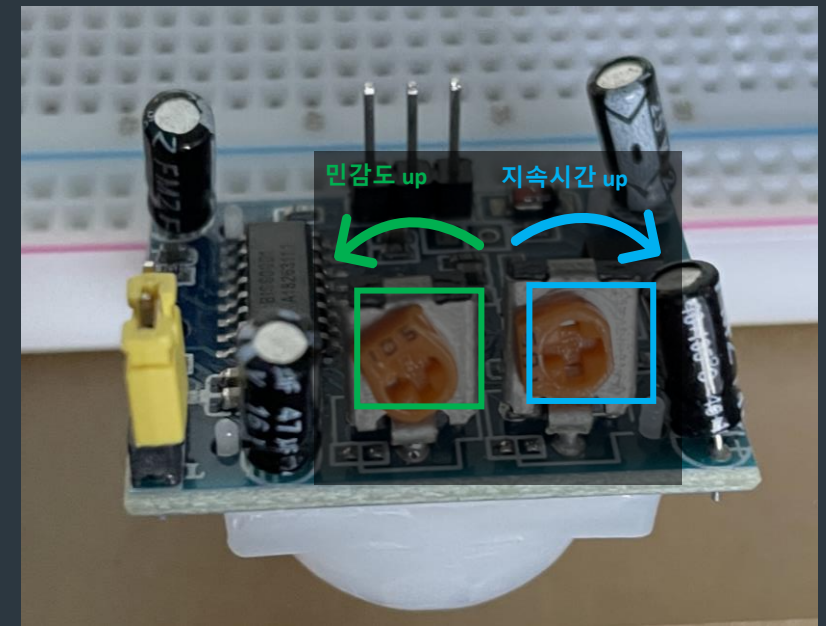
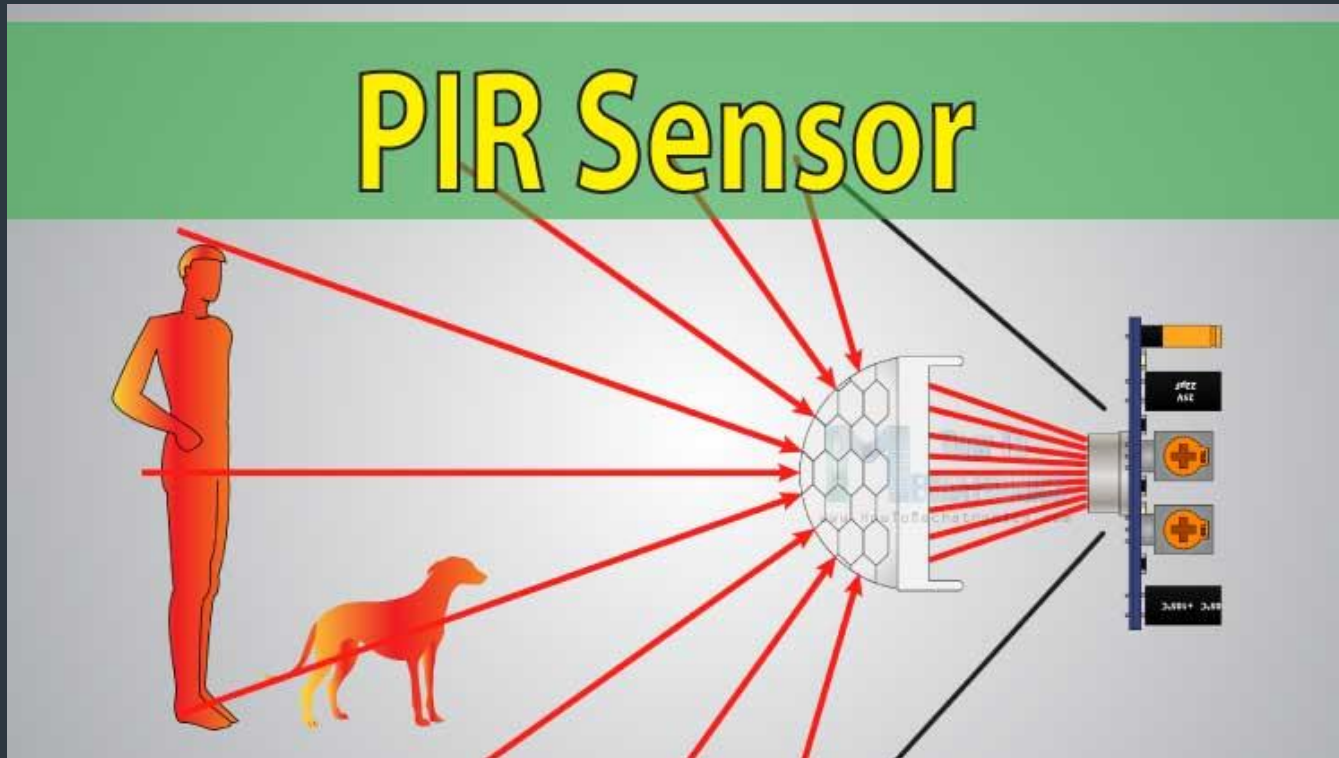
- PIR의 VCC핀(red)은 4번핀과 연결
- PIR의 신호핀(yellow)은 5번핀과 연결
- PIR의 ground (black)은 6번핀과 연결



1	VCC3.3V_IO	2	VCC5V_SYS
3	GP8A4_I2C1_SDA	4	VCC5V_SYS
5	GP8A5_I2C1_SCL	6	GND
7	GP0C1_CLKOUT	8	GP5B1_UART1TX
9	GND	10	GP5B0_UART1RX
11	GP5B4_SPI0CLK_UART4CTSN	12	GP6A0_PCM/I2S_CLK
13	GP5B6_SPI0_TXD_UART4TX	14	GND
15	GP5B7_SPI0_RXD_UART4RX	16	GP5B2_UART1CTSN
17	VCC3.3V_IO	18	GP5B3_UART1RTSN
19	GP8B1_SPI2TXD	20	GND
21	GP8B0_SPI2RXD	22	GP5C3
23	GP8A6_SPI2CLK	24	GP8A7_SPI2CSN0
25	GND	26	GP8A3_SPI2CSN1
27	GP7C1_I2C4_SDA	28	GP7C2_I2C4_SCL
29	GP5B5_SPI0CSN0_UART4RTSN	30	GND
31	GP5C0_SPI0CSN1	32	GP7C7_UART2TX_PWM3
33	GP7C6_UART2RX_PWM2	34	GND
35	GP6A1_PCM/I2S_FS	36	GP7A7_UART3RX
37	GP7B0_UART3TX	38	GP6A3_PCM/I2S_SDI
39	GND	40	GP6A4_PCM/I2S_SDO

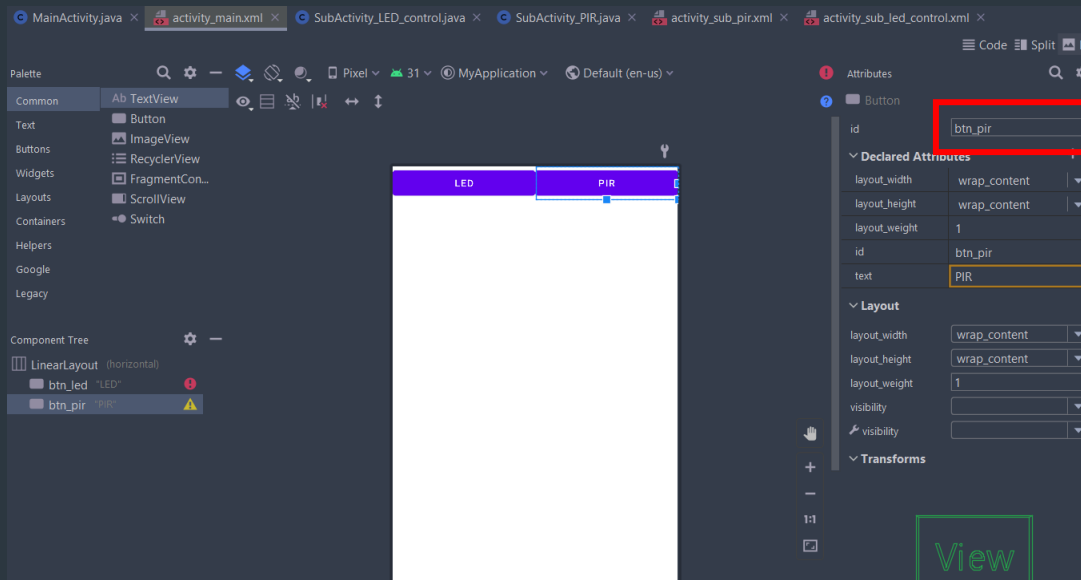
예제3: PIR 센서의 원리

- 적외선 센서가 렌즈를 통해 들어오는 인체의 적외선을 감지
- 신호가 감지되면 HIGH, 감지되지 않으면 LOW 신호를 반환



예제3: SubActivity 생성 후 Onclick 함수 정의

1. subActivity 생성 후 MainActivity에 PIR 버튼 생성

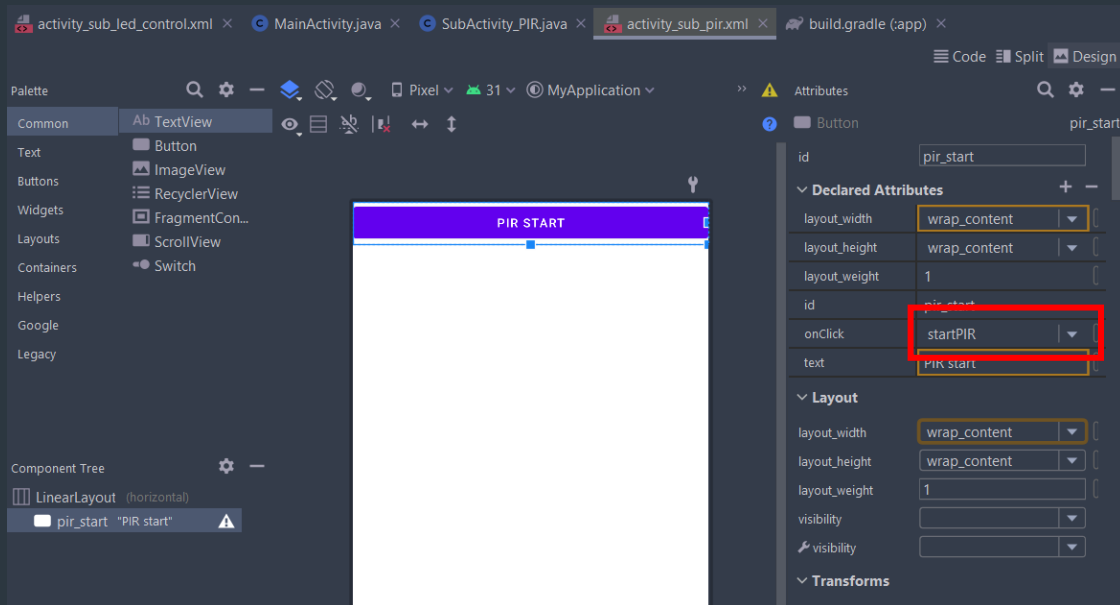


2. MainActivity에서 subActivity로 이동하는 onClick함수 정의

```
16  
17 Button btn_led = findViewById(R.id.btn_led);  
18 btn_led.setOnClickListener(new View.OnClickListener() {  
19     @Override  
20     public void onClick(View view) {  
21         Intent intent = new Intent( packageContext: MainActivity.this, SubActivity_LED_control.class);  
22         startActivity(intent);  
23     }  
24 });  
25  
26 Button btn_pir = findViewById(R.id.btn_pir);  
27 btn_pir.setOnClickListener(new View.OnClickListener() {  
28     @Override  
29     public void onClick(View view) {  
30         Intent intent = new Intent( packageContext: MainActivity.this, SubActivity_PIR.class);  
31         startActivity(intent);  
32     }  
33 });
```

예제3: SubAcitivity 생성 후 Onclick 함수 정의

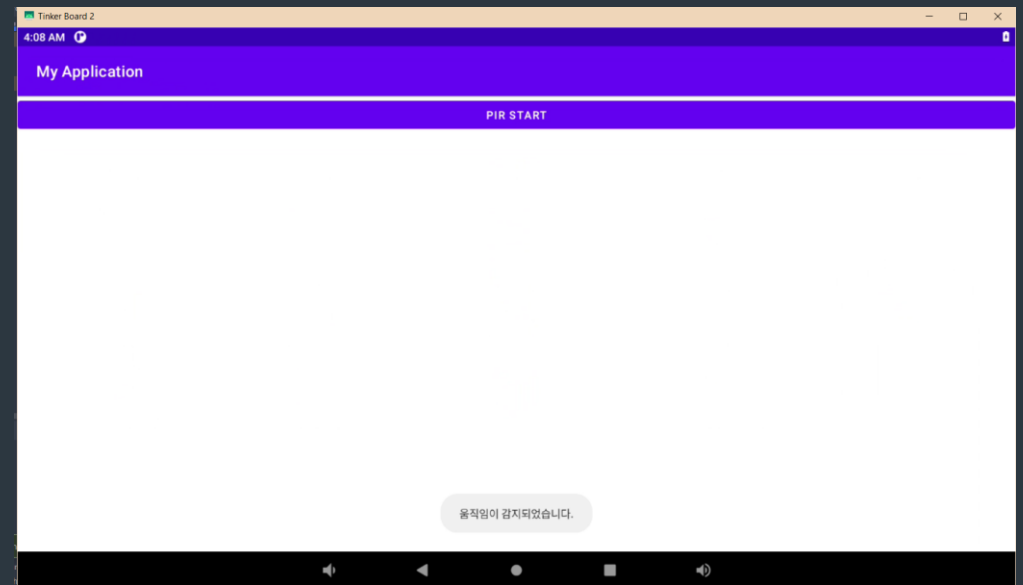
3. subActivity 레이아웃에서 PIR 버튼 생성 및 onClick



4. subActivity 소스코드에서 onClick 함수 정의

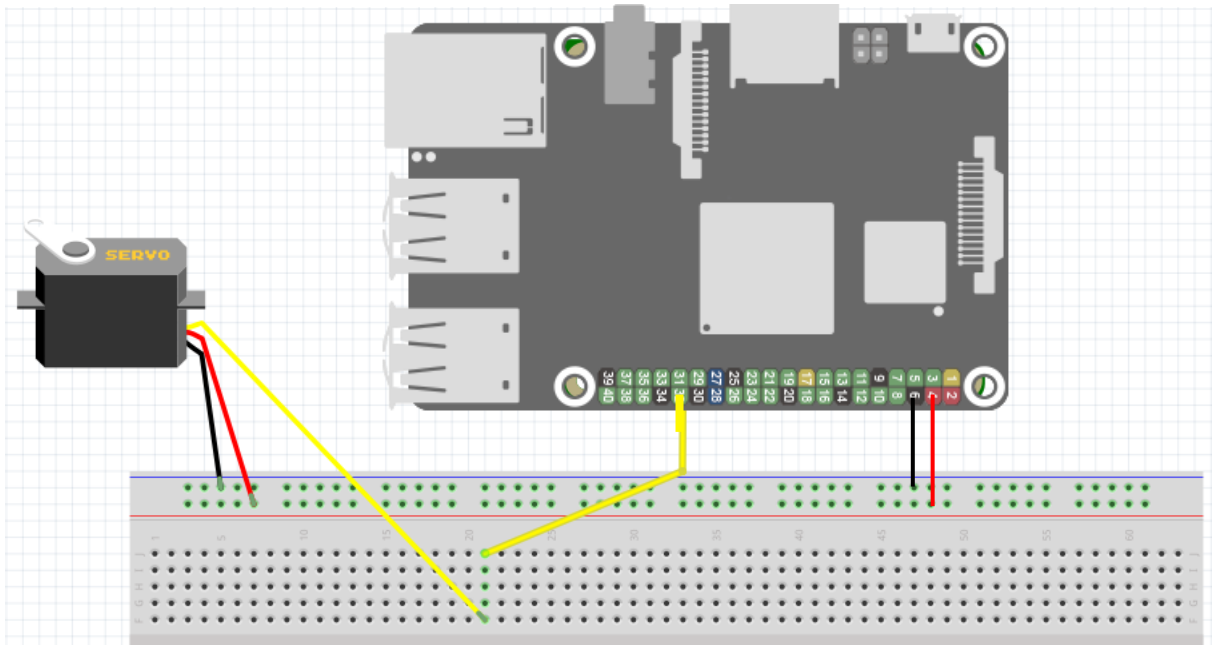
```
public void startPIR(View v) {  
    Gpio gpio5 = new Gpio(TinkerBoard.TINKERBOARD_PINS.swigValue());  
    gpio5.dir(Dir.DIR_IN);  
    int sensor_val = gpio5.read();  
    if( sensor_val == 1) {  
        Toast.makeText( context: this, text: "움직임이 감지되었습니다.", Toast.LENGTH_SHORT).show();  
    } else {  
        Toast.makeText( context: this, text: "움직임이 감지되지 않았습니다.", Toast.LENGTH_SHORT).show();  
    }  
}
```

예제3: 실행화면



예제4: 서보 모터 (PWM 제어)

- Servo의 VCC핀(red)은 4번핀과 연결
- Servo의 신호핀(yellow)은 32번핀과 연결
- Servo의 ground (black)은 6번핀과 연결

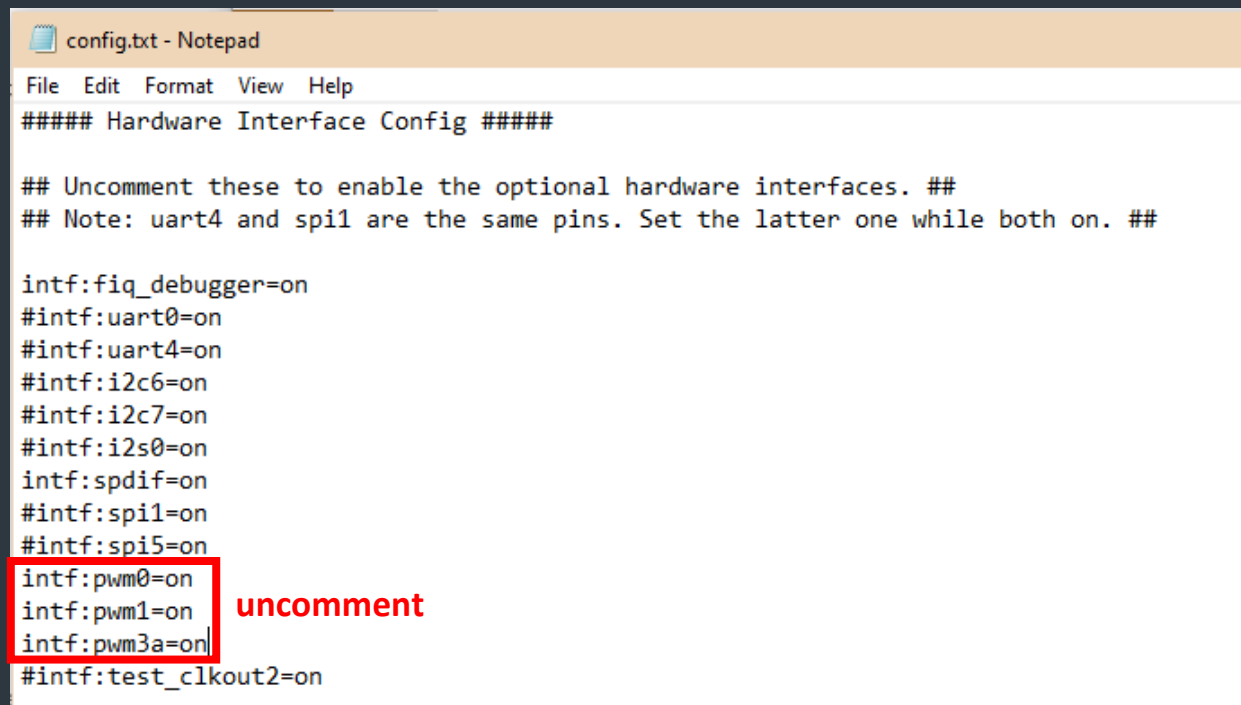


1	VCC3.3V_IO	2	VCC5V_SYS
3	GP8A4_I2C1_SDA	4	VCC5V_SYS
5	GP8A5_I2C1_SCL	6	GND
7	GP0C1_CLKOUT	8	GP5B1_UART1TX
9	GND	10	GP5B0_UART1RX
11	GP5B4_SPI0CLK_UART4CTSN	12	GP6A0_PCM/I2S_CLK
13	GP5B6_SPI0_TXD_UART4TX	14	GND
15	GP5B7_SPI0_RXD_UART4RX	16	GP5B2_UART1CTSN
17	VCC3.3V_IO	18	GP5B3_UART1RTSN
19	GP8B1_SPI2TXD	20	GND
21	GP8B0_SPI2RXD	22	GP5C3
23	GP8A6_SPI2CLK	24	GP8A7_SPI2CSN0
25	GND	26	GP8A3_SPI2CSN1
27	GP7C1_I2C4_SDA	28	GP7C2_I2C4_SCL
29	GP5B5_SPI0CSN0_UART4RTSN	30	GND
31	GP5C0_SPI0CSN1	32	GP7C7_UART2TX_PWM3
33	GP7C6_UART2RX_PWM2	34	GND
35	GP6A1_PCM/I2S_FS	36	GP7A7_UART3RX
37	GP7B0_UART3TX	38	GP6A3_PCM/I2S_SDI
39	GND	40	GP6A4_PCM/I2S_SDO

예제 4: Tinker Board OS의 config.txt 파일 수정

Window cmd에서 아래 순서대로 진행

1. adb devices
2. adb root
3. adb remount
4. adb pull /dtoverlay/config.txt
5. config.txt 파일 내용 수정
6. adb push config.txt /dtoverlay/
7. adb reboot



```
config.txt - Notepad
File Edit Format View Help
##### Hardware Interface Config #####

## Uncomment these to enable the optional hardware interfaces. ##
## Note: uart4 and spi1 are the same pins. Set the latter one while both on. ##

intf:fiq_debugger=on
#intf:uart0=on
#intf:uart4=on
#intf:i2c6=on
#intf:i2c7=on
#intf:i2s0=on
intf:spdif=on
#intf:spi1=on
#intf:spi5=on
intf:pwm0=on
intf:pwm1=on
intf:pwm3a=on
#intf:test_clkout2=on
```

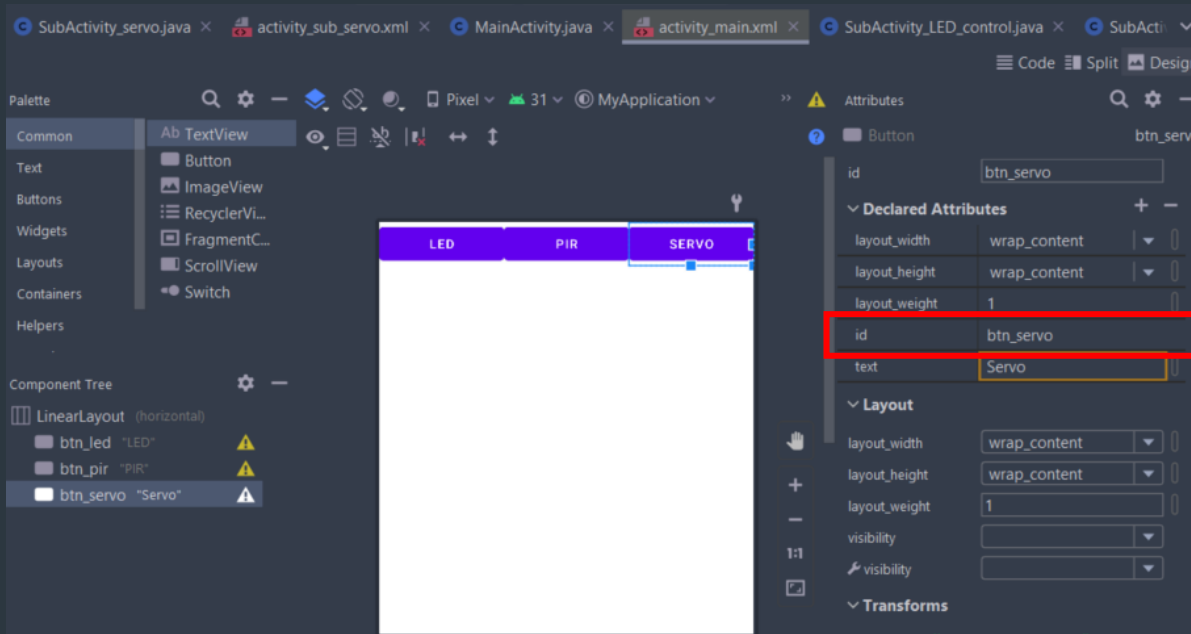
uncomment

예제 4: Tinker Board OS의 config.txt 파일 수정

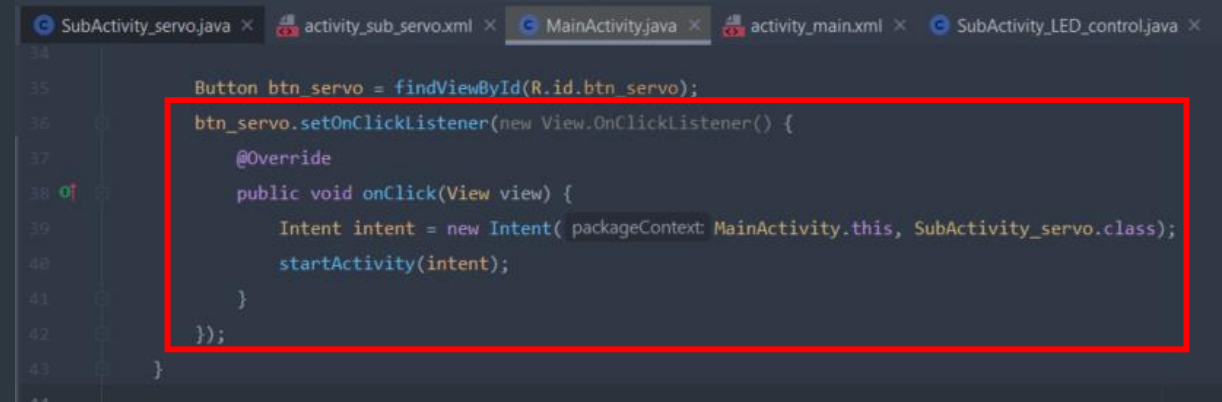
```
Command Prompt
C:\Users\wasb46>cd C:\Users\wasb46\Downloads\Android-MraaDemo_tinkerboard2\Android-MraaDemo_tinkerboard2\MraaDemo\app\build\outputs\apk\debug
C:\Users\wasb46\Downloads\Android-MraaDemo_tinkerboard2\Android-MraaDemo_tinkerboard2\MraaDemo\app\build\outputs\apk\debug>adb devices
List of devices attached
210686628300573 device
C:\Users\wasb46\Downloads\Android-MraaDemo_tinkerboard2\Android-MraaDemo_tinkerboard2\MraaDemo\app\build\outputs\apk\debug>adb root
restarting adbd as root
C:\Users\wasb46\Downloads\Android-MraaDemo_tinkerboard2\Android-MraaDemo_tinkerboard2\MraaDemo\app\build\outputs\apk\debug>adb remount
remount succeeded
C:\Users\wasb46\Downloads\Android-MraaDemo_tinkerboard2\Android-MraaDemo_tinkerboard2\MraaDemo\app\build\outputs\apk\debug>adb pull /dtoverlay/config.txt
/dtoverlay/config.txt: 1 file pulled, 0 skipped. 1.1 MB/s (1732 bytes in 0.002s)
C:\Users\wasb46\Downloads\Android-MraaDemo_tinkerboard2\Android-MraaDemo_tinkerboard2\MraaDemo\app\build\outputs\apk\debug>adb push config.txt /dtoverlay/
config.txt: 1 file pushed, 0 skipped. 2.5 MB/s (1732 bytes in 0.001s)
C:\Users\wasb46\Downloads\Android-MraaDemo_tinkerboard2\Android-MraaDemo_tinkerboard2\MraaDemo\app\build\outputs\apk\debug>adb reboot
```

예제 4: SubActivity 생성 후 Onclick 함수 정의

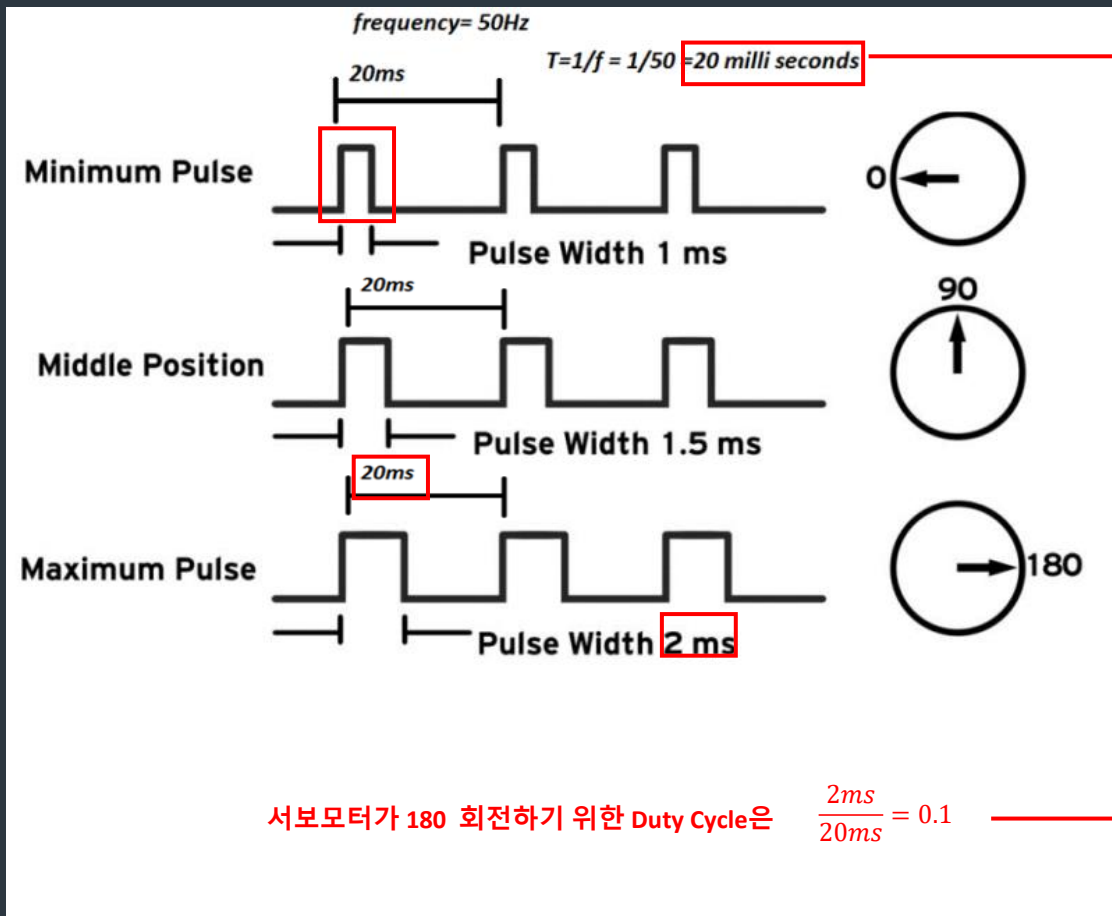
1. subActivity 생성 후 MainActivity에 SERVO 버튼 생성



2. MainActivity에서 subActivity로 이동하는 onClick함수 정의



예제3: 서보모터의 원리



```
public void onbtn_servo180(View v) {  
    Pwm pwm = new Pwm(TinkerBoard.TINKERBOARD_PIN32.swigValue());  
    pwm.enable( b: true);  
    pwm.period_ms( i: 20);  
    pwm.write((float) 0.1);  
    Toast.makeText( context: this, text: "180 degree 회전", Toast.LENGTH_SHORT).show();  
}  
  
public void onbtn_servo0(View v) {  
    Pwm pwm = new Pwm(TinkerBoard.TINKERBOARD_PIN32.swigValue());  
    pwm.enable( b: true);  
    pwm.period_ms( i: 20);  
    pwm.write((float) 0.05);  
    Toast.makeText( context: this, text: "180 degree 회전", Toast.LENGTH_SHORT).show();  
}
```

예제3: 서보모터의 원리

실제 180도와 0도로 동작되지 않고 더 적은 범위로 동작된다면?

- 180도 일 때, 0.12, 0도 일때 0.03으로 수정하여 테스트
- SG90모터는 작동범위에 오차가 많기 때문에 교정이 필요함

최소 범위: 0.03

최대 범위: 0.12

```
public void onbtn_servo180(View v) {
    Pwm pwm = new Pwm(TinkerBoard.TINKERBOARD_PIN32.swigValue());
    pwm.enable( b: true);
    pwm.period_ms( i: 20);
    pwm.write((float) 0.1);
    Toast.makeText( context: this, text: "180 degree 회전", Toast.LENGTH_SHORT).show();
}

public void onbtn_servo0(View v) {
    Pwm pwm = new Pwm(TinkerBoard.TINKERBOARD_PIN32.swigValue());
    pwm.enable( b: true);
    pwm.period_ms( i: 20);
    pwm.write((float) 0.05);
    Toast.makeText( context: this, text: "180 degree 회전", Toast.LENGTH_SHORT).show();
}
```

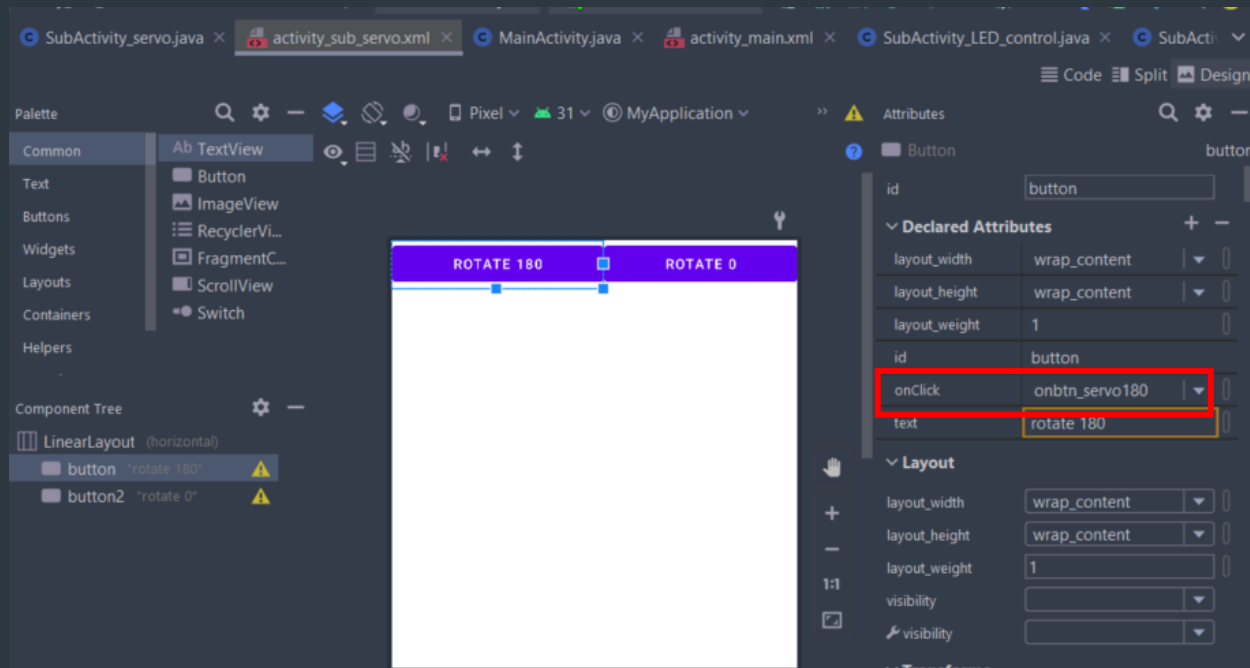


```
public void onbtn_servo180(View v) {
    Pwm pwm = new Pwm(TinkerBoard.TINKERBOARD_PIN32.swigValue());
    pwm.enable( b: true);
    pwm.period_ms( i: 20);
    pwm.write((float) 0.12);
    Toast.makeText( context: this, text: "180 degree 회전", Toast.LENGTH_SHORT).show();
}

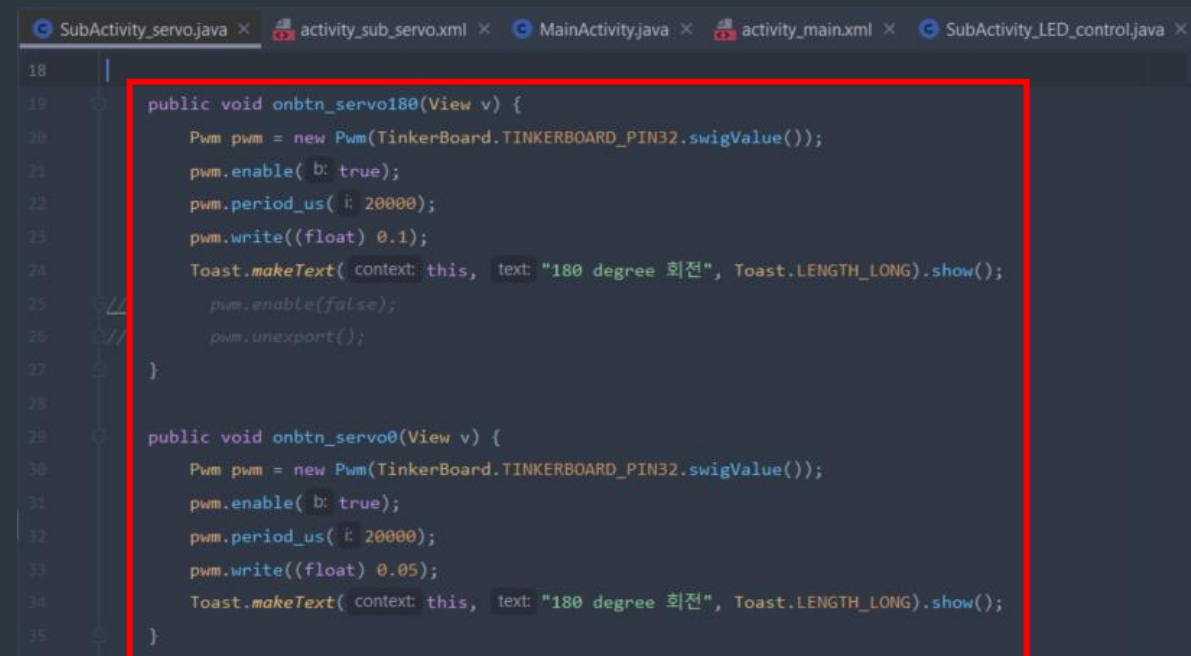
public void onbtn_servo0(View v) {
    Pwm pwm = new Pwm(TinkerBoard.TINKERBOARD_PIN32.swigValue());
    pwm.enable( b: true);
    pwm.period_ms( i: 20);
    pwm.write((float) 0.03);
    Toast.makeText( context: this, text: "180 degree 회전", Toast.LENGTH_SHORT).show();
}
```

예제 4: SubActivity의 layout과 소스코드 작성

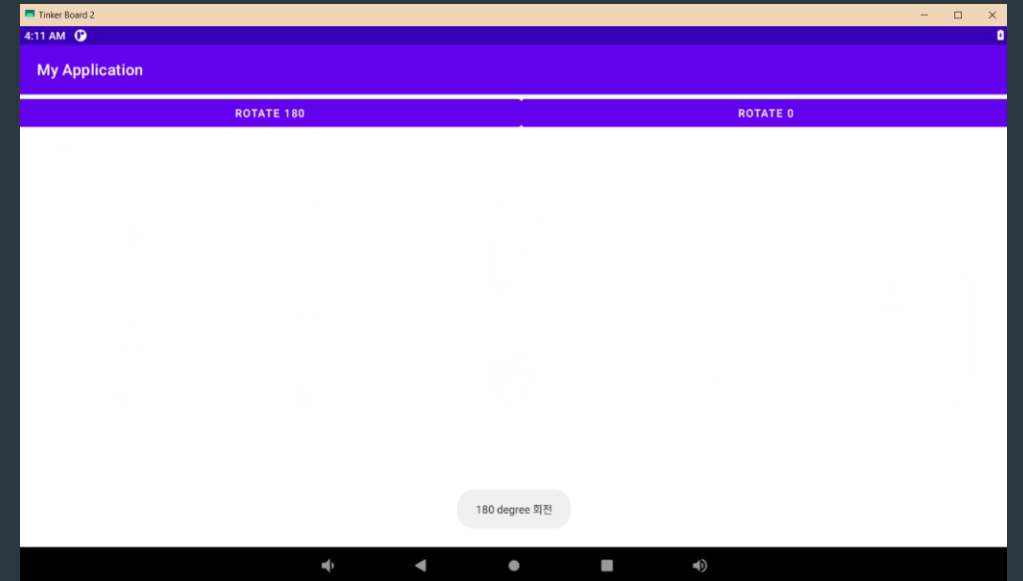
3. subActivity의 layout에서 onClick함수 정의



4. SubActivity에서 PWM 메소드 구현



예제 4: 실행화면



샘플 코드: APK of Mraa API for Tinker Board 2

[다운로드 링크](#)

아래 명령어를 통해서 apk 파일을 디바이스에 install

- **adb install Android-MraaDemo_tinkerboard2.apk**

9:00 AM

MraaDemo for Tinker Board 2

	GPIO number	Function2	Function1	GPIO	Pin#	Pin#	GPIO	Function1	Function2	GPIO number
1	Io Map		VCC3.3V_IO		1	2		VCC5V_SYS		
2	Gpio	73	I2C6_SDA	GPIO2_B1	3	4		VCC5V_SYS		
		74	I2C6_SCL	GPIO2_B2	5	6		GND		
3	I2c	8	TEST_CLKOUT2	GPIO0_B0	7	8	GPIO2_C1	UART0_TXD		81
			GND		9	10	GPIO2_C0	UART0_RXD		80
4	Pwm	83	UART0_RTSN	GPIO2_C3	11	12	GPIO3_D0	I2S0_SCLK		120
		85	SPI5_TX	GPIO2_C5	13	14		GND		
5	Spi	84	SPI5_RX	GPIO2_C4	15	16	GPIO2_C6	SPI5_CLK		86
			VCC3.3V_IO		17	18	GPIO2_C7	SPI5_CSN		87
6	Uart	40	UART4_TXD	SPI1_TXD	GPIO1_B0	19	20		GND	
		39	UART4_RXD	SPI1_RXD	GPIO1_A7	21	22	GPIO3_D4	I2S0_SDO3	124
		41		SPI1_CLK	GPIO1_B1	23	24	GPIO1_B2	SPI1_CSN	42
			GND			25	26	GPIO0_A6	PWM3A_IR	6
		71	I2C7_SDA	GPIO2_A7	27	28	GPIO2_B0	I2C7_SCL		72
		126	I2S0_SDO1	GPIO3_D6	29	30		GND		
		125	I2S0_SDO2	GPIO3_D5	31	32	GPIO4_C2	PWM0		146
		150	PWM1	GPIO4_C6	33	34		GND		
		121	I2S0_FS	GPIO3_D1	35	36	GPIO2_C2	UART0_CTSN		82
		149	SPDIF_TX	GPIO4_C5	37	38	GPIO3_D3	I2S1_SDI0		123
			GND		39	40	GPIO3_D7	I2S1_SDO0		127