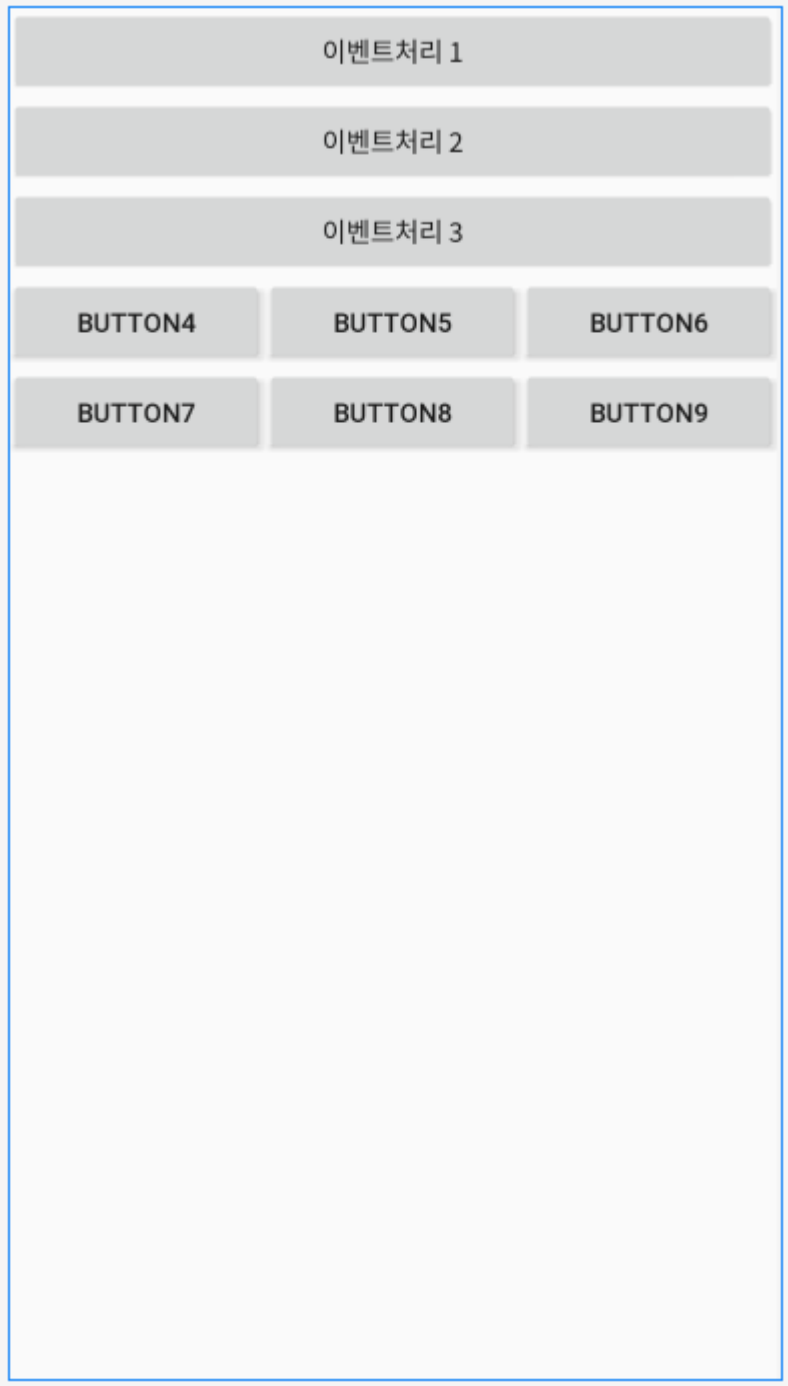
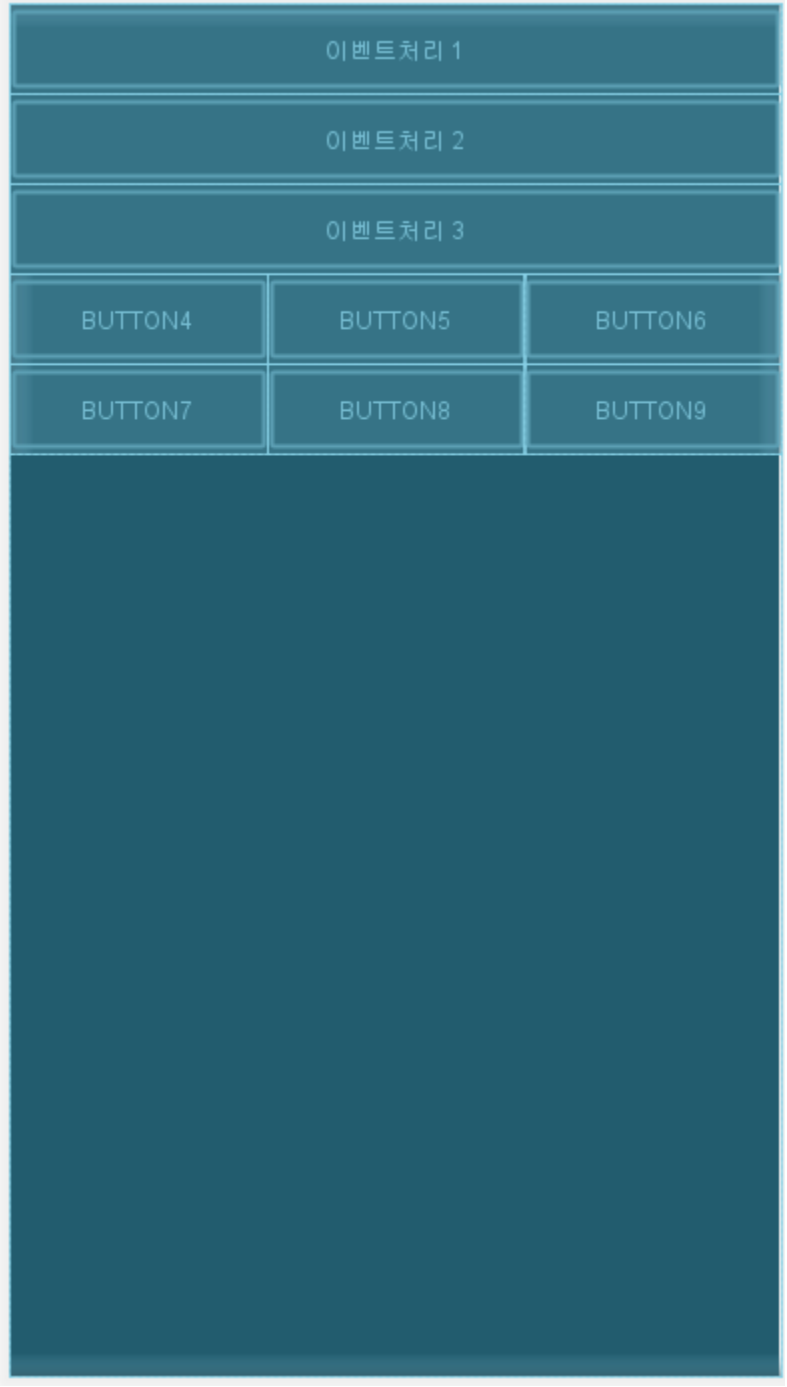
03

이벤트 처리 – Event Handling

이벤트 핸들링 방법


6:09
EventHandling
이벤트처리 1
이벤트처리 2
이벤트처리 3
BUTTON4
BUTTON5
BUTTON6
BUTTON7
BUTTON8
BUTTON9


이벤트처리 1
이벤트처리 2
이벤트처리 3
BUTTON4
BUTTON5
BUTTON6
BUTTON7
BUTTON8
BUTTON9


이벤트처리 1
이벤트처리 2
이벤트처리 3
BUTTON4
BUTTON5
BUTTON6
BUTTON7
BUTTON8
BUTTON9

메서드	설명
onClick(View v)	사용자가 항목을 터치하면 호출됩니다.
onLongClick(View v)	사용자가 항목을 길게 터치하면 호출됩니다.
onFocusChange(View v)	사용자가 다른 항목으로 포커스를 하면 호출됩니다.
onKey(View v)	사용자가 기기에 있는 키를 누르거나 손을 떼면 호출됩니다.
onTouch(View v)	사용자가 터치 이벤트로서의 자격을 만족하는 작업을 수행하는 경우에 호출되며, 여기에는 누르기, 손 떼기와 화면에서 이루어지는 모든 움직임 동작(항목의 경계 내에서)이 포함됩니다.
onCreateContextMenu(View v)	메뉴가 구축되는 중일 때 호출됩니다.

id값을 이용한 이벤트 처리

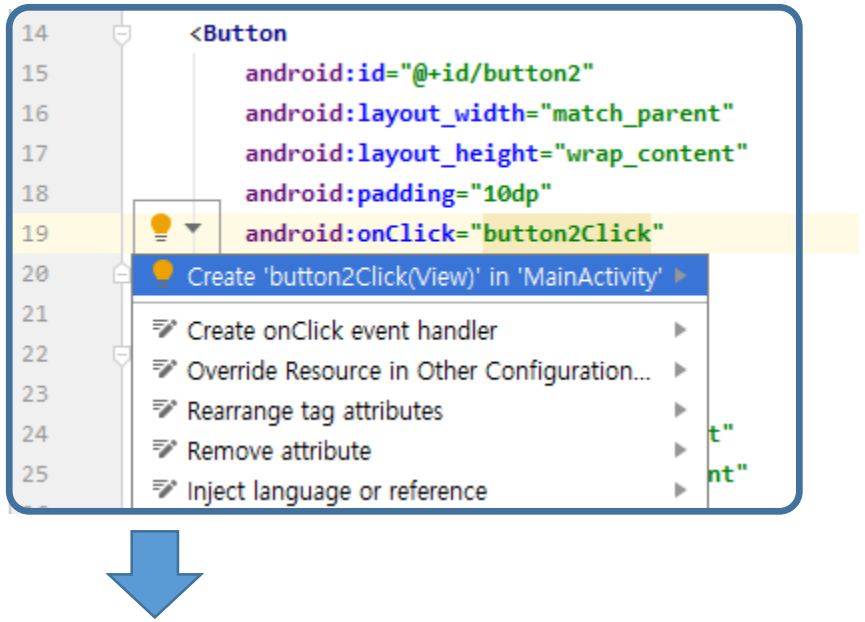
id값을 이용하여 이벤트 처리를 하기 위해서는 아래와 같은 메서드를 이용합니다.

메서드	설명
findViewById(int id)	id값을 찾아주는 역할을 하는 메서드입니다. 매개변수로 R.id.아이디명을 넣어주면 됩니다.
setOnclickListener(OnclickListener listener)	해당 요소에 이벤트 리스너를 설정해주는 메서드입니다.

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    android:orientation="vertical" >

    <Button
        android:id="@+id/button"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="10dp"
        android:text="이벤트처리 1" />
</LinearLayout>
```

```java
public class MainActivity extends AppCompatActivity {
    Button button;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);
        button = findViewById(R.id.button);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Toast.makeText(getApplicationContext(), "버튼1이 눌려졌습니다", Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

```xml
<?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:id="@+id/button2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="10dp"
        android:onClick="button2Click"
        android:text="이벤트처리 2" />
</LinearLayout>
```




```java
public class MainActivity extends AppCompatActivity {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    public void button2Click(View view) {
        Toast.makeText(getApplicationContext(), "버튼2가 눌려졌습니다", Toast.LENGTH_SHORT).show();
    }
}
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    android:orientation="vertical" >

    <Button
        android:id="@+id/button3"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="10dp"
        android:text="이벤트처리 3" />
</LinearLayout>
```

```java
public class MainActivity extends AppCompatActivity {
    Button button3;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        button3 = findViewById(R.id.button3);
        MyButtonClass buttonListener = new MyButtonClass();
        button3.setOnClickListener(buttonListener);
    }

    class MyButtonClass implements View.OnClickListener{
        @Override
        public void onClick(View view) {
            Toast.makeText(getApplicationContext(),"버튼3이 눌려졌습니다",Toast.LENGTH_SHORT).show();
        }
    }
}
```

1. MyButtonClass 생성(이름은 각자가 생성)

```
public class MainActivity extends AppCompatActivity {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
    }
    class MyButtonClass implements View.OnClickListener{
        @Override
        public void onClick(View view) {
            Toast.makeText(getApplicationContext(),"버튼3이 눌려졌습니다",Toast.LENGTH_SHORT).show();
        }
    }
}
```

2. MyButtonClass 클래스의 객체를 생성하고 button3의 setOnClickListener 와 연동

```
public class MainActivity extends AppCompatActivity {
    Button button3;
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        button3 = findViewById(R.id.button3);
        MyButtonClass buttonListener = new MyButtonClass();
        button3.setOnClickListener(buttonListener);
    }
}
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <Button
        android:id="@+id/button4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Button4" />

    <Button
        android:id="@+id/button5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Button5" />

    <Button
        android:id="@+id/button6"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Button6" />
</LinearLayout>
```

```
public class MainActivity extends AppCompatActivity {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        Button.OnClickListener onClickListener = new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                switch (view.getId()) {
                    case R.id.button4:
                        Toast.makeText(getApplicationContext(), "버튼4가 눌려졌습니다", Toast.LENGTH_SHORT).show();
                        break;
                    case R.id.button5:
                        Toast.makeText(getApplicationContext(), "버튼5이 눌려졌습니다", Toast.LENGTH_SHORT).show();
                        break;
                    case R.id.button6:
                        Toast.makeText(getApplicationContext(), "버튼6이 눌려졌습니다", Toast.LENGTH_SHORT).show();
                        break;
                }
            }
        };
        //버튼의 위치은 onClickListener아래에
        Button button4 = findViewById(R.id.button4);
        button4.setOnClickListener(onClickListener);
        Button button5 = findViewById(R.id.button5);
        button5.setOnClickListener(onClickListener);
        Button button6 = findViewById(R.id.button6);
        button6.setOnClickListener(onClickListener);
    }
}
```

```java
public class MainActivity extends AppCompatActivity {
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        BtnOnClickListener onClickListener1 = new BtnOnClickListener();
        Button button7 = findViewById(R.id.button7);
        button7.setOnClickListener(onClickListener1);
        Button button8 = findViewById(R.id.button8);
        button8.setOnClickListener(onClickListener1);
        Button button9 = findViewById(R.id.button9);
        button9.setOnClickListener(onClickListener1);
    }

    class BtnOnClickListener implements View.OnClickListener{
        @Override
        public void onClick(View view) {
            switch (view.getId()){
                case R.id.button7 :
                    Toast.makeText(getApplicationContext(), "버튼7이 눌려졌습니다", Toast.LENGTH_SHORT).show();
                    break;
                case R.id.button8:
                    Toast.makeText(getApplicationContext(), "버튼8이 눌려졌습니다", Toast.LENGTH_SHORT).show();
                    break;
                case R.id.button9:
                    Toast.makeText(getApplicationContext(), "버튼9가 눌려졌습니다", Toast.LENGTH_SHORT).show();
                    break;
            }
        }
    }
}
```

```xml
<?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:id="@+id/button"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="10dp"
        android:text="이벤트처리 1" />

    <Button
        android:id="@+id/button2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="10dp"
        android:onClick="button2Click"
        android:text="이벤트처리 2" />

    <Button
        android:id="@+id/button3"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="10dp"
        android:text="이벤트처리 3" />
```

```xml
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <Button
        android:id="@+id/button4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Button4" />

    <Button
        android:id="@+id/button5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Button5" />

    <Button
        android:id="@+id/button6"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Button6" />
</LinearLayout>
```

```
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <Button
            android:id="@+id/button7"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Button7" />

        <Button
            android:id="@+id/button8"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Button8" />

        <Button
            android:id="@+id/button9"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Button9" />
    </LinearLayout>

</LinearLayout>
```

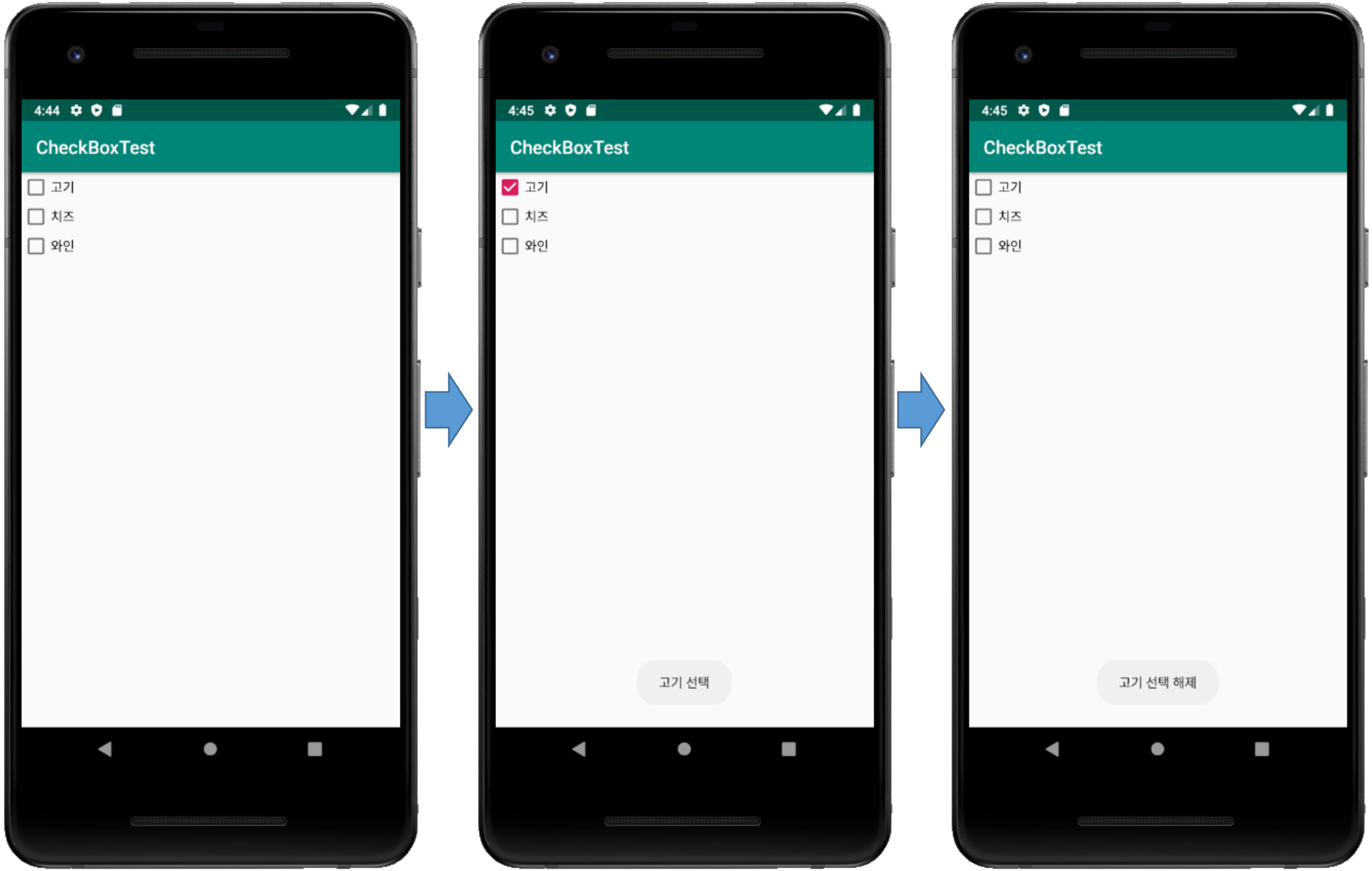
```java
public class MainActivity extends AppCompatActivity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);
        Button button = (Button) findViewById(R.id.button);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Toast.makeText(getApplicationContext(), "버튼1이 눌려졌습니다", Toast.LENGTH_SHORT).show();
            }
        });

        Button button3 = findViewById(R.id.button3);
        MyButtonClass buttonListener = new MyButtonClass();
        button3.setOnClickListener(buttonListener);
```

```
Button.OnClickListener onClickListener = new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        switch (view.getId()) {
            case R.id.button4:
                Toast.makeText(getApplicationContext(), "버튼4가 눌려졌습니다", Toast.LENGTH_SHORT).show();
                break;
            case R.id.button5:
                Toast.makeText(getApplicationContext(), "버튼5이 눌려졌습니다", Toast.LENGTH_SHORT).show();
                break;
            case R.id.button6:
                Toast.makeText(getApplicationContext(), "버튼6이 눌려졌습니다", Toast.LENGTH_SHORT).show();
                break;
        }
    }
};
//버튼의 위치은 onClickListener아래에
 Button button4 = findViewById(R.id.button4);
button4.setOnClickListener(onClickListener);
Button button5 = findViewById(R.id.button5);
button5.setOnClickListener(onClickListener);
Button button6 = findViewById(R.id.button6);
button6.setOnClickListener(onClickListener);
```

```
        BtnOnClickListener onClickListener1 = new BtnOnClickListener();

        Button button7 = findViewById(R.id.button7);
        button7.setOnClickListener(onClickListener1);
        Button button8 = findViewById(R.id.button8);
        button8.setOnClickListener(onClickListener1);
        Button button9 = findViewById(R.id.button9);
        button9.setOnClickListener(onClickListener1);

    }

    public void button2Click(View view) {
        Toast.makeText(getApplicationContext(), "버튼2가 눌려졌습니다", Toast.LENGTH_SHORT).show();
    }

    class MyButtonClass implements View.OnClickListener{
        @Override
        public void onClick(View view) {
            Toast.makeText(getApplicationContext(),"버튼3이 눌려졌습니다",Toast.LENGTH_SHORT).show();
        }
    }
```

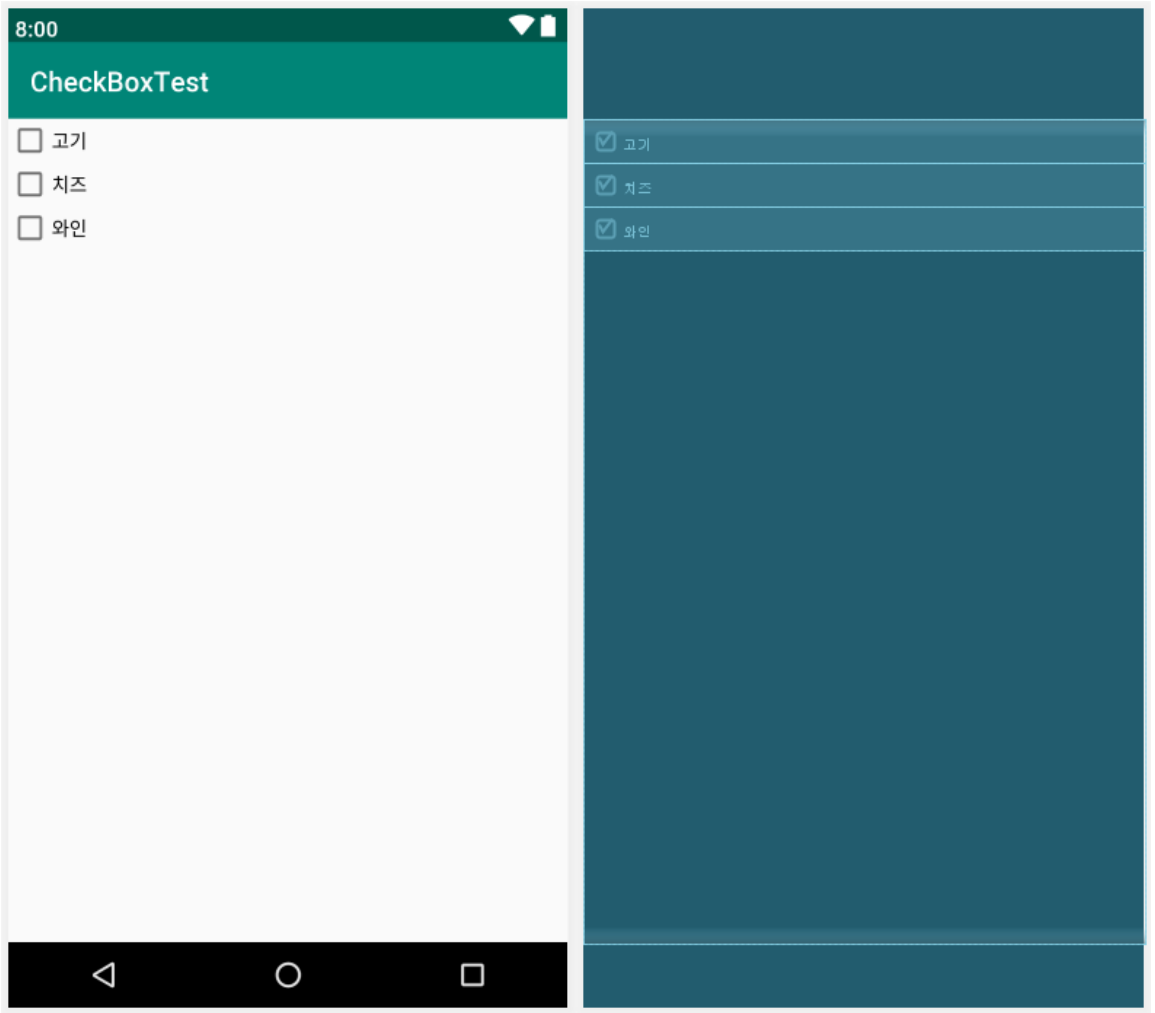
```
    class BtnOnClickListener implements View.OnClickListener{
        @Override
        public void onClick(View view) {
            switch (view.getId()){
                case R.id.button7 :
                    Toast.makeText(getApplicationContext(), "버튼7이 눌려졌습니다", Toast.LENGTH_SHORT).show();
                    break;
                case R.id.button8:
                    Toast.makeText(getApplicationContext(), "버튼8이 눌려졌습니다", Toast.LENGTH_SHORT).show();
                    break;
                case R.id.button9:
                    Toast.makeText(getApplicationContext(), "버튼9가 눌려졌습니다", Toast.LENGTH_SHORT).show();
                    break;
            }
        }
    }

}
```

CHECKBOX


4:44
CheckBoxTest
고기
치즈
와인
4:45
CheckBoxTest
고기
치즈
와인
고기 선택
4:45
CheckBoxTest
고기
치즈
와인
고기 선택 해제

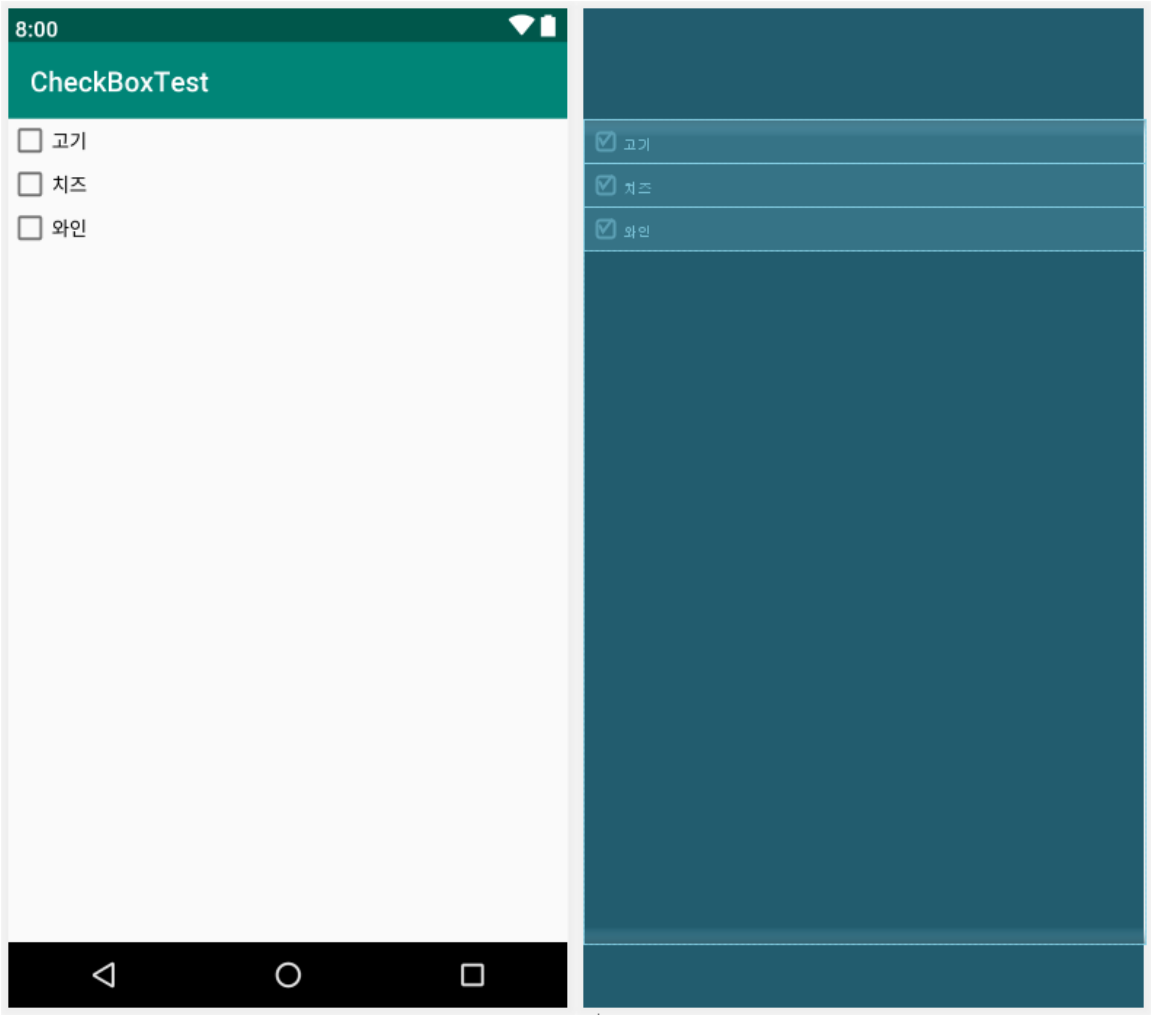
activity_main.xml


```
<?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/checkbox_meat"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="onCheckboxClicked"
        android:text="고기" />
    <CheckBox
        android:id="@+id/checkbox_cheese"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="onCheckboxClicked"
        android:text="치즈" />
    <CheckBox
        android:id="@+id/checkbox_wine"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="onCheckboxClicked"
        android:text="와인" />
 </LinearLayout>
```



activity_main.xml


```
<?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/checkbox_meat"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="onCheckboxClicked"
        android:text="고기" />
    <CheckBox
        android:id="@+id/checkbox_cheese"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="onCheckboxClicked"
        android:text="치즈" />
    <CheckBox
        android:id="@+id/checkbox_wine"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="onCheckboxClicked"
        android:text="와인" />
 </LinearLayout>
```



MainActivty.java


```
7     <CheckBox
8         android:id="@+id/checkbox_meat"
9         android:layout_width="match_parent"
10        android:layout_height="wrap_content"
11        android:onClick="onCheckboxClicked"
12
13
14
15
16
```

Create onClick event handler
Override Resource in Other Configuration...
Rearrange tag attributes
Remove attribute
Inject language or reference

```java
public class MainActivity extends AppCompatActivity{
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    public void onCheckboxClicked(View view) {
    }
}
```

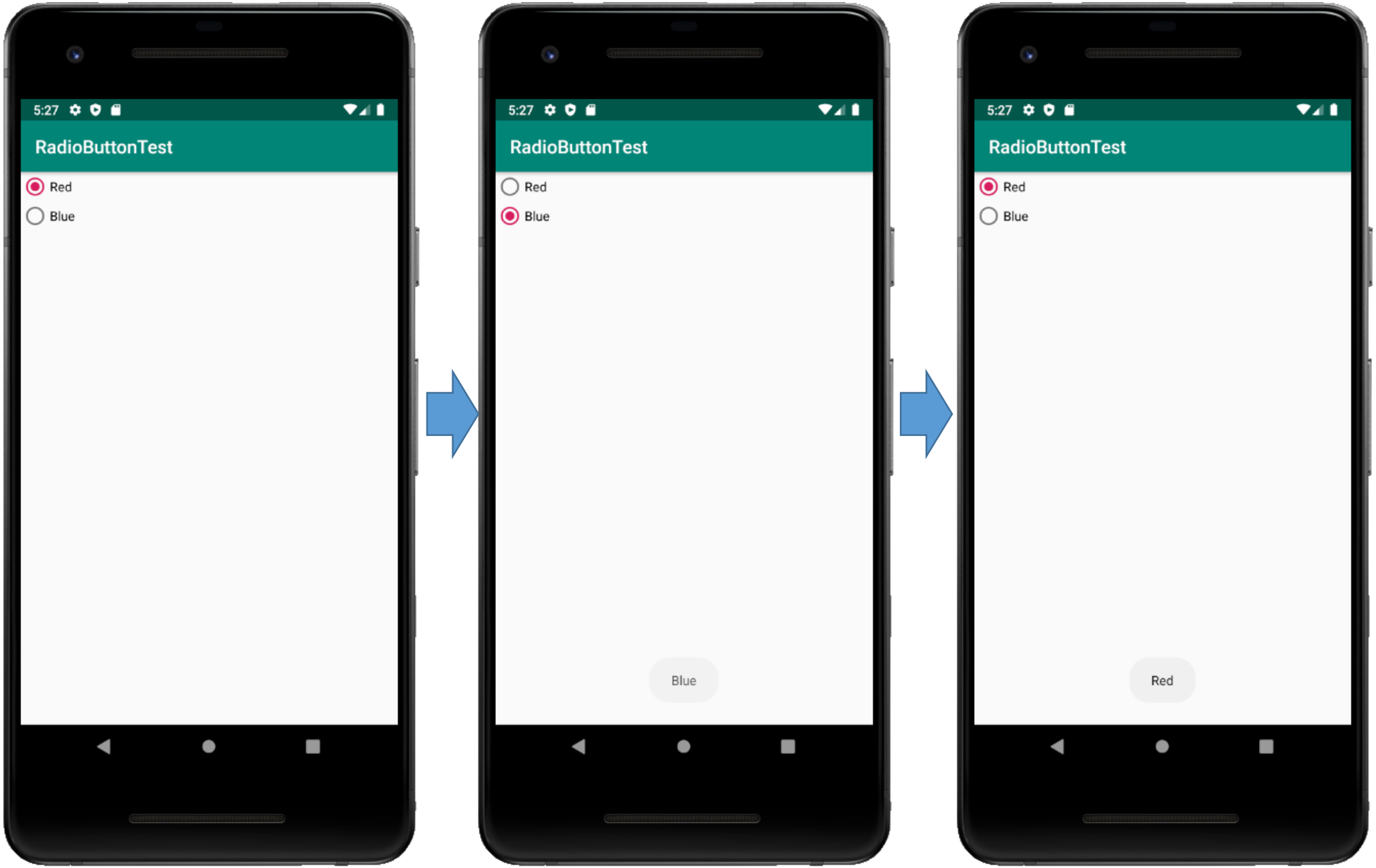
MainActivty.java


```
public class MainActivity extends AppCompatActivity{
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    public void onCheckboxClicked(View view) {
        boolean checked = ((CheckBox) view).isChecked();

        switch(view.getId()) {
            case R.id.checkbox_meat:
                if (checked)
                    Toast.makeText(getApplicationContext(), "고기 선택", Toast.LENGTH_SHORT).show();
                else
                    Toast.makeText(getApplicationContext(), "고기 선택 해제", Toast.LENGTH_SHORT).show();
                break;
        }
    }
}
```

MainActivty.java

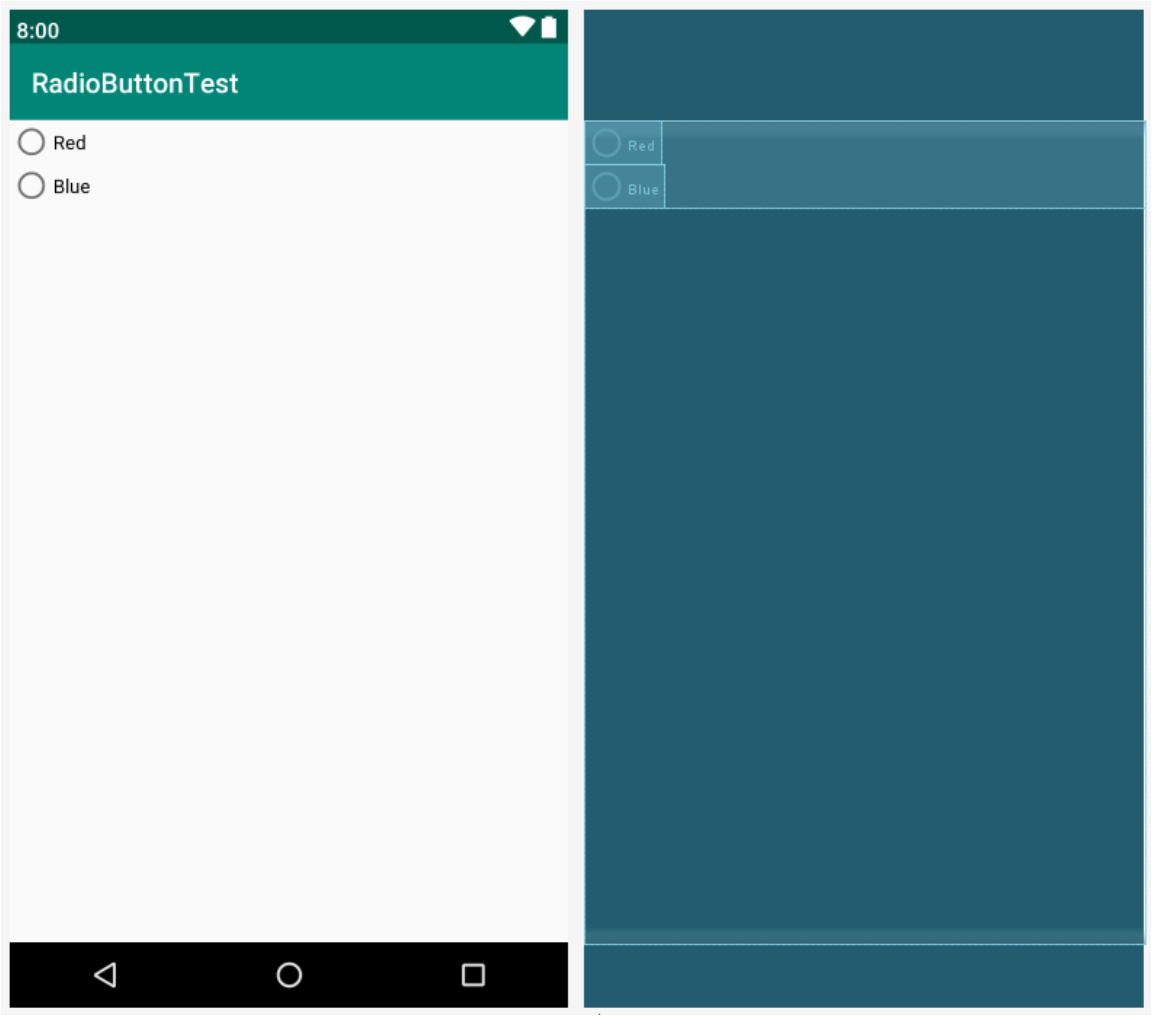

```
public class MainActivity extends AppCompatActivity{
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    public void onCheckboxClicked(View view) {
        boolean checked = ((CheckBox) view).isChecked();

        switch(view.getId()) {
            case R.id.checkbox_meat:
                if (checked)
                    Toast.makeText(getApplicationContext(), "고기 선택", Toast.LENGTH_SHORT).show();
                else
                    Toast.makeText(getApplicationContext(), "고기 선택 해제", Toast.LENGTH_SHORT).show();
                break;
            case R.id.checkbox_cheese:
                if (checked)
                    Toast.makeText(getApplicationContext(), "치즈 선택", Toast.LENGTH_SHORT).show();
                else
                    Toast.makeText(getApplicationContext(), "치즈 선택 해제", Toast.LENGTH_SHORT).show();
                break;
            case R.id.checkbox_wine:
                if (checked)
                    Toast.makeText(getApplicationContext(), "와인 선택", Toast.LENGTH_SHORT).show();
                else
                    Toast.makeText(getApplicationContext(), "와인 선택 해제", Toast.LENGTH_SHORT).show();
                break;
        }
    }
}
```

RADIO BUTTON


5:27
RadioButtonTest
Red
Blue
5:27
RadioButtonTest
Red
Blue
Blue
5:27
RadioButtonTest
Red
Blue
Red

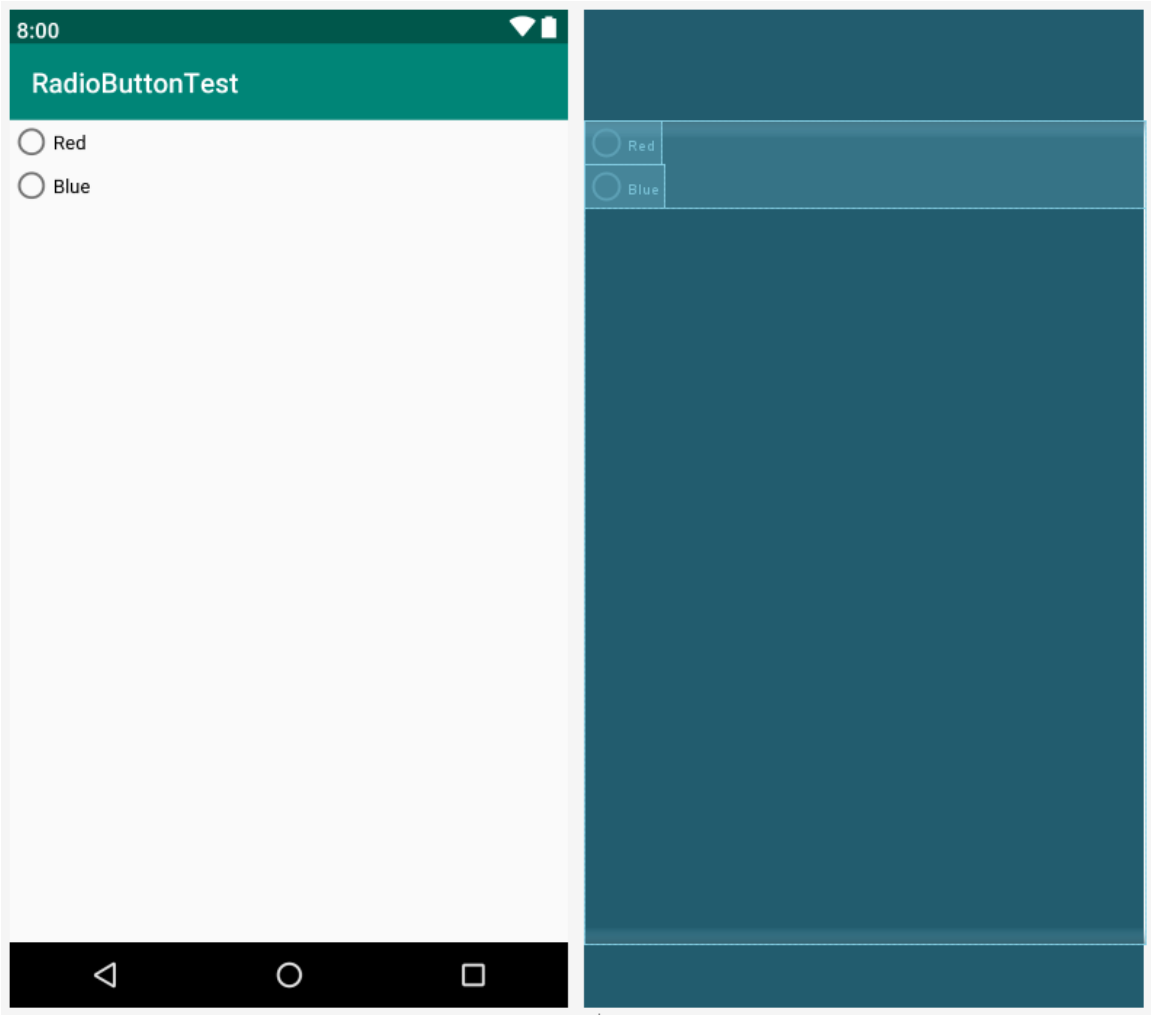
activity_main.xml


```
<?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" >

    <RadioGroup
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical" >

        <RadioButton
            android:id="@+id/radio_red"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:onClick="onRadioButtonClicked"
            android:text="Red"/>
        <RadioButton
            android:id="@+id/radio_blue"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:onClick="onRadioButtonClicked"
            android:text="Blue"/>
    </RadioGroup>
</LinearLayout>
```



activity_main.xml


```
<?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" >

    <RadioGroup
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical" >

        <RadioButton
            android:id="@+id/radio_red"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:onClick="onRadioButtonClicked"
            android:text="Red"/>
        <RadioButton
            android:id="@+id/radio_blue"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:onClick="onRadioButtonClicked"
            android:text="Blue"/>
    </RadioGroup>
</LinearLayout>
```



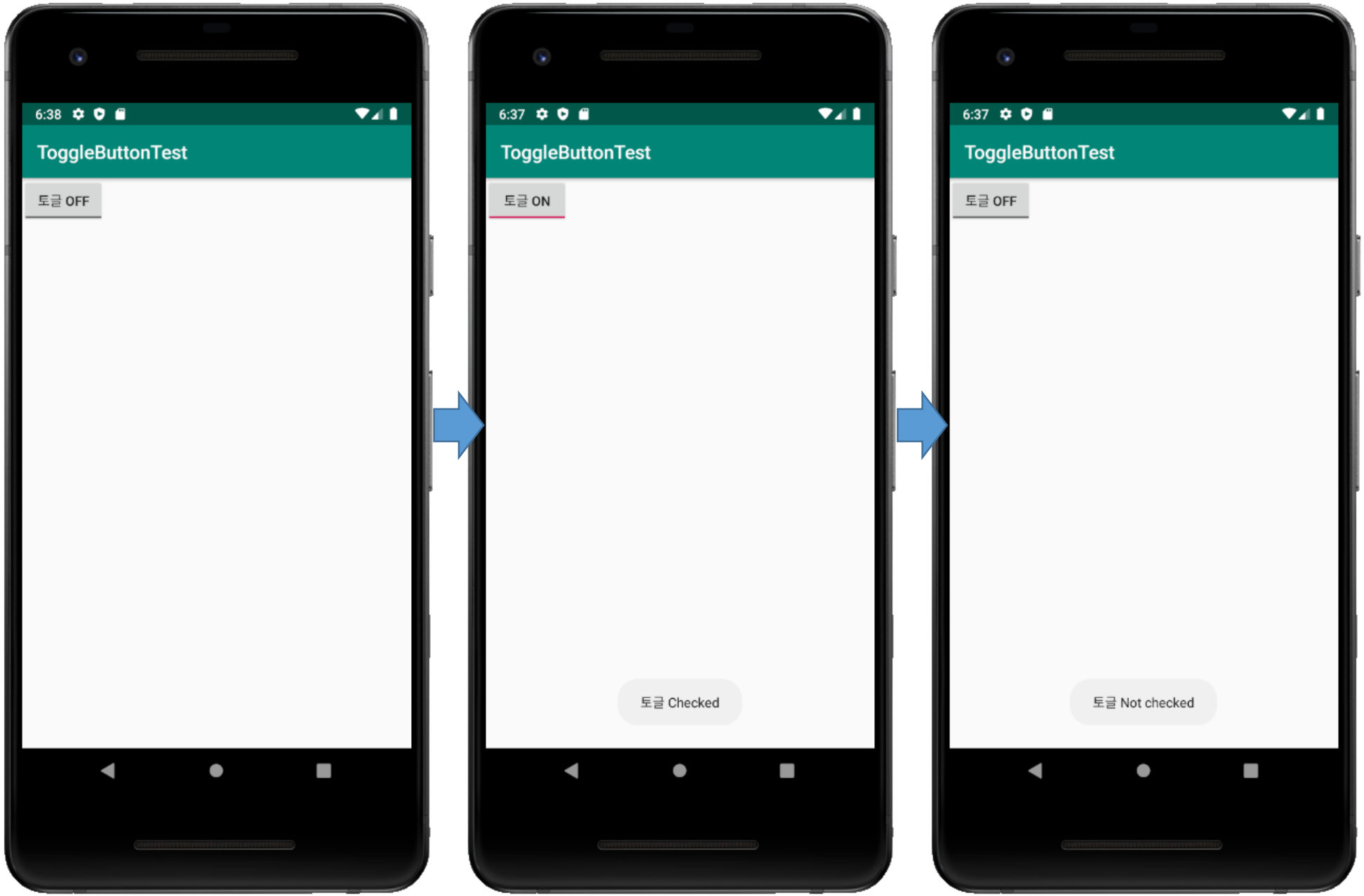
MainActivty.java


```
public class MainActivity extends AppCompatActivity {

    RadioButton radio_red, radio_blue;

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

        radio_red = findViewById(R.id.radio_red);
        radio_blue = findViewById(R.id.radio_blue);
    }

    public void onRadioButtonClicked(View view) {
        boolean checked = ((RadioButton) view).isChecked();
        switch(view.getId()) {
            case R.id.radio_red:
                if (checked)
                    Toast.makeText(getApplicationContext(), radio_red.getText(), Toast.LENGTH_SHORT).show();
                break;
            case R.id.radio_blue:
                if (checked)
                    Toast.makeText(getApplicationContext(), radio_blue.getText(), Toast.LENGTH_SHORT).show();
                break;
        }
    }
}
```

MainActivty.java

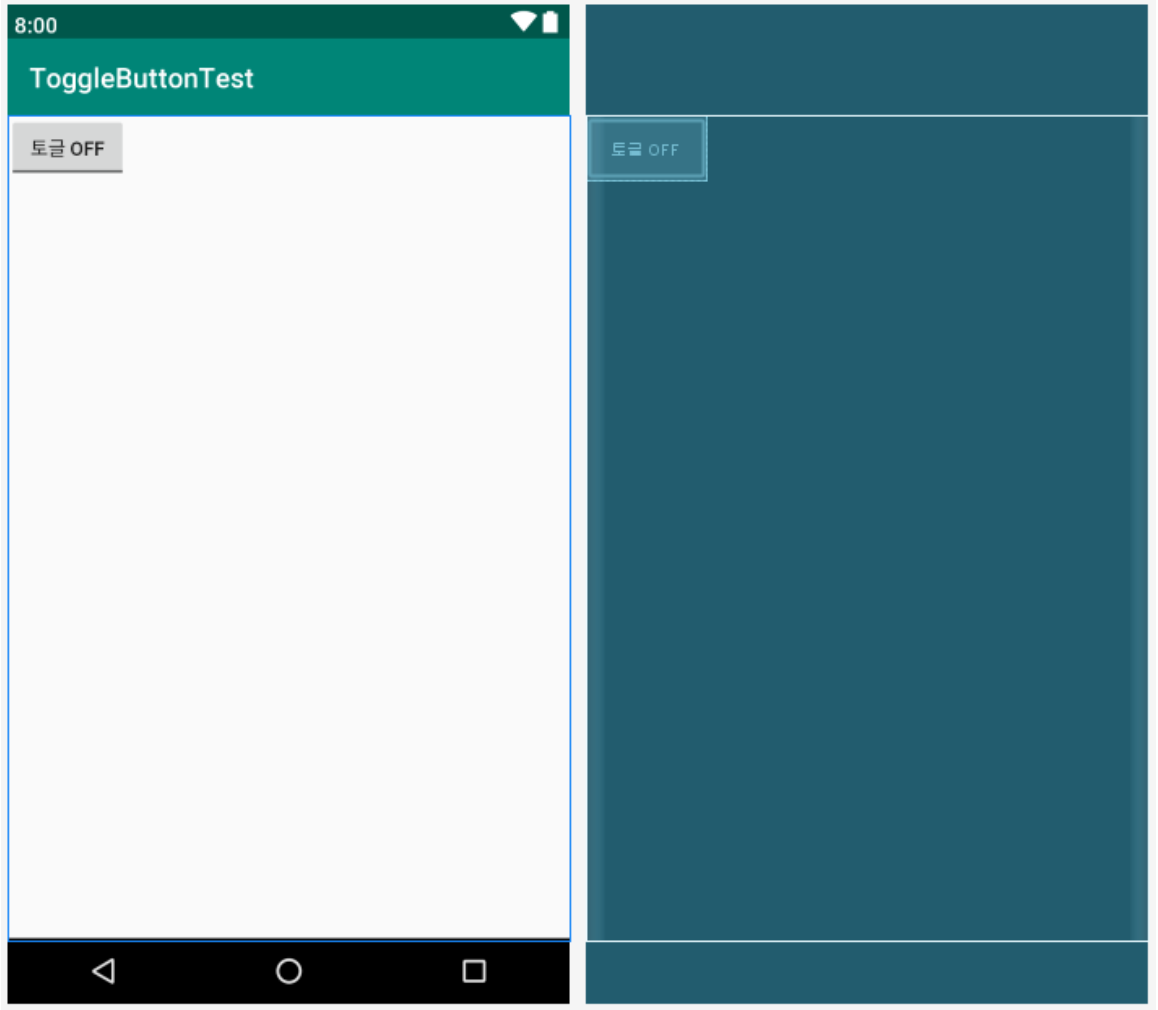

```
public class MainActivity extends AppCompatActivity {

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

    public void onRadioButtonClicked(View view) {
        boolean checked = ((RadioButton) view).isChecked();
        switch(view.getId()) {
            case R.id.radio_red:
                if (checked)
                    Toast.makeText(getApplicationContext(), ((RadioButton) view).getText(), Toast.LENGTH_SHORT).show();
                break;
            case R.id.radio_blue:
                if (checked)
                    Toast.makeText(getApplicationContext(), ((RadioButton) view).getText(), Toast.LENGTH_SHORT).show();
                break;
        }
    }
}
```

TOGGLE BUTTON


6:38
ToggleButtonTest
토글 OFF
6:37
ToggleButtonTest
토글 ON
토글 Checked
6:37
ToggleButtonTest
토글 OFF
토글 Not checked

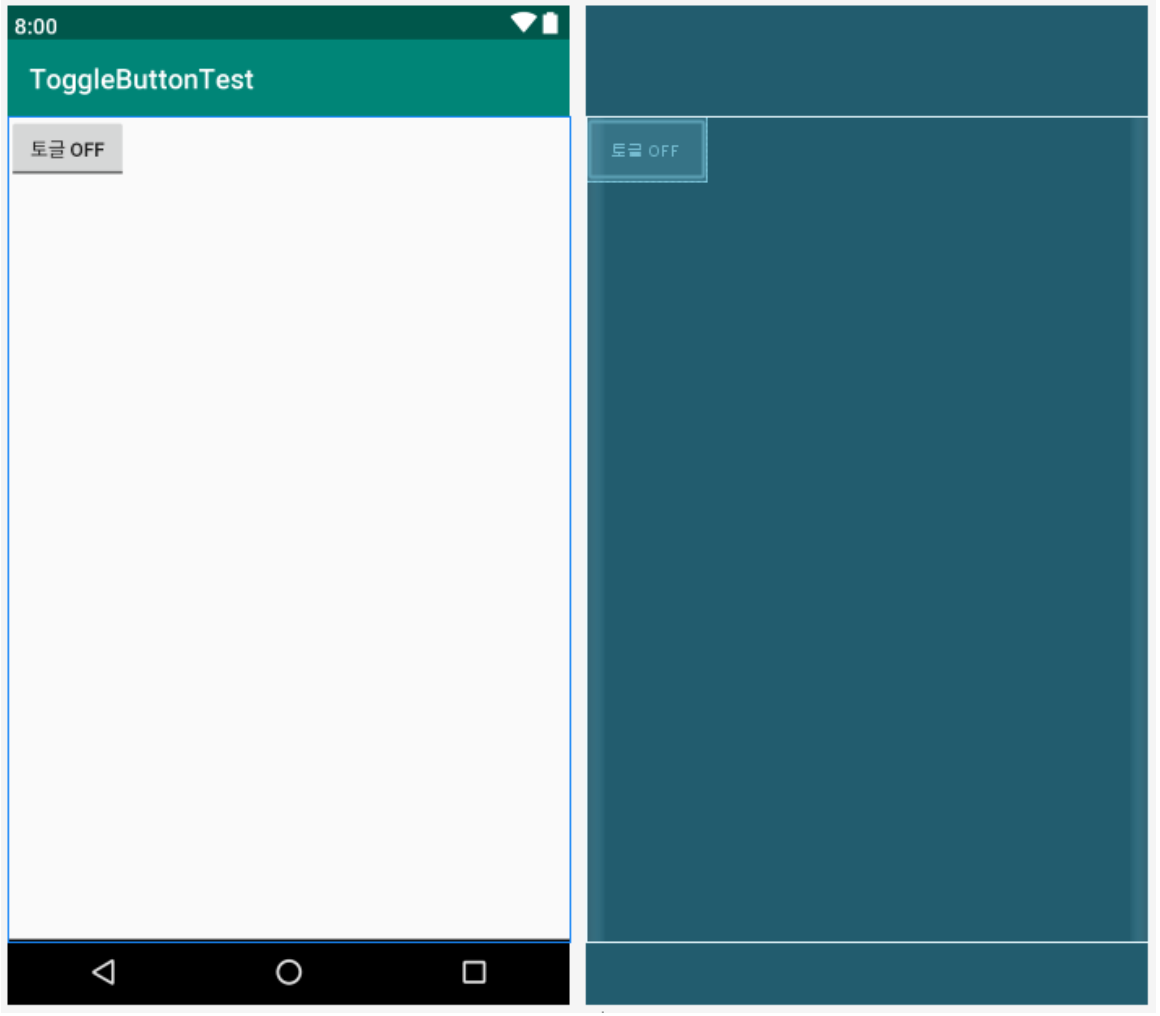
activity_main.xml


```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent" >

    <ToggleButton
        android:id="@+id/togglebutton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textOn="토글 ON"
        android:textOff="토글 OFF"
        android:onClick="onToggleClicked"
        />
</LinearLayout>
```



activity_main.xml

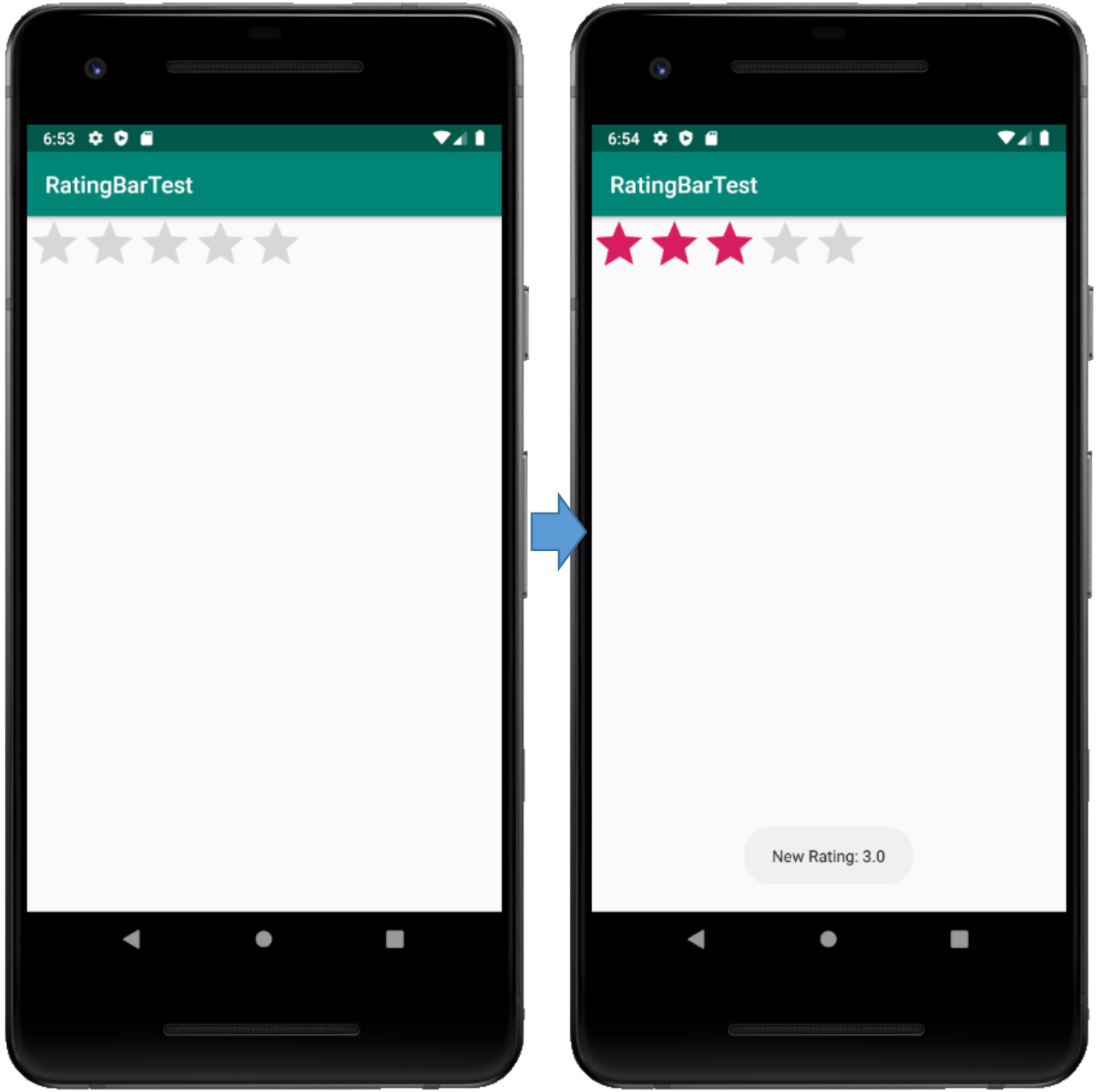

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent" >

    <ToggleButton
        android:id="@+id/togglebutton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textOn="토글 ON"
        android:textOff="토글 OFF"
        android:onClick="onToggleClicked"
        />
</LinearLayout>
```



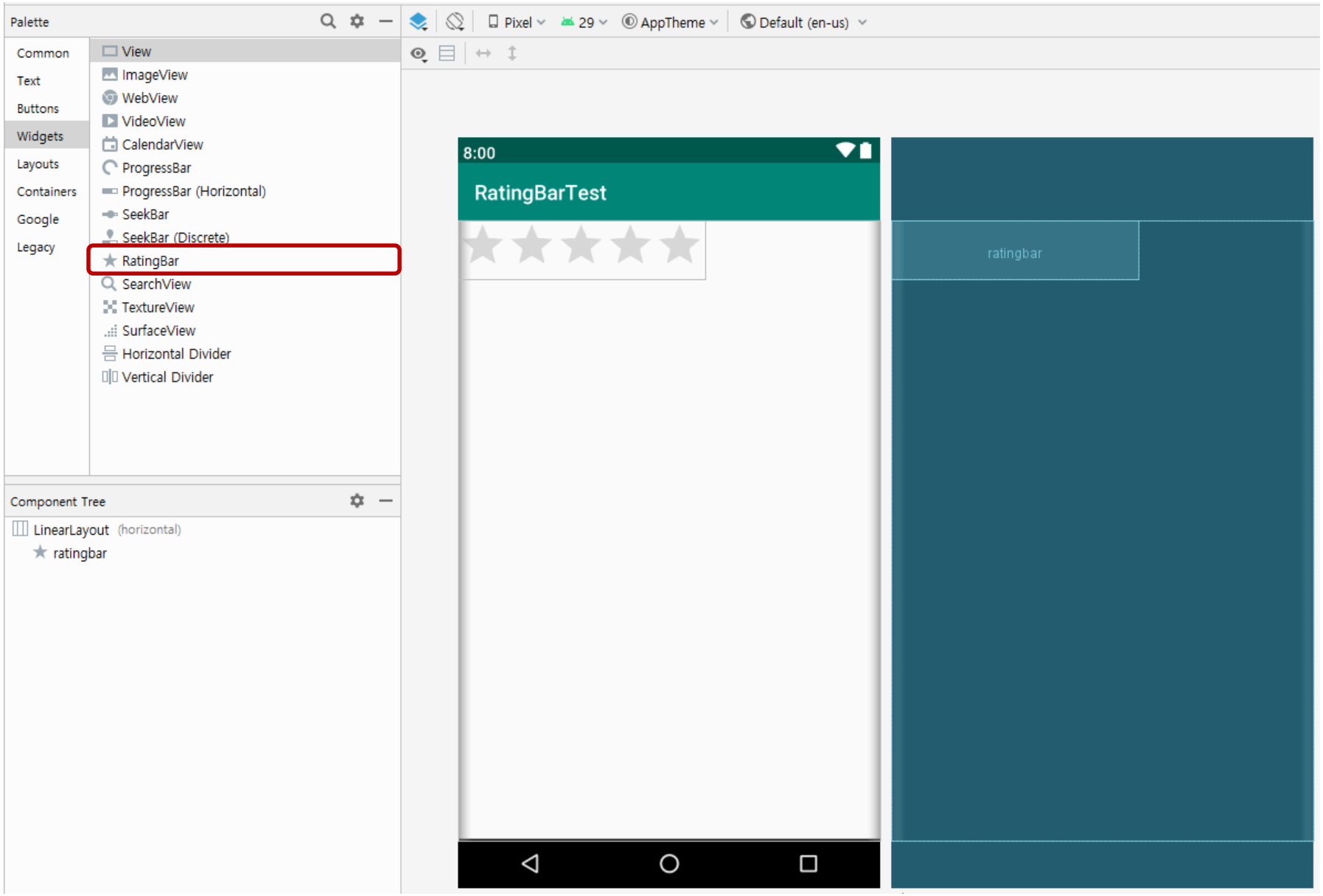
MainActivty.java


```
public class MainActivity extends AppCompatActivity {

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

    public void onToggleClicked(View view) {
        boolean on = ((ToggleButton) view).isChecked();

        if (on) {
            Toast.makeText(getApplicationContext(), "토글 Checked", Toast.LENGTH_SHORT).show();
        } else {
            Toast.makeText(getApplicationContext(), "토글 Not checked", Toast.LENGTH_SHORT).show();
        }
    }

}
```

RATING BAR

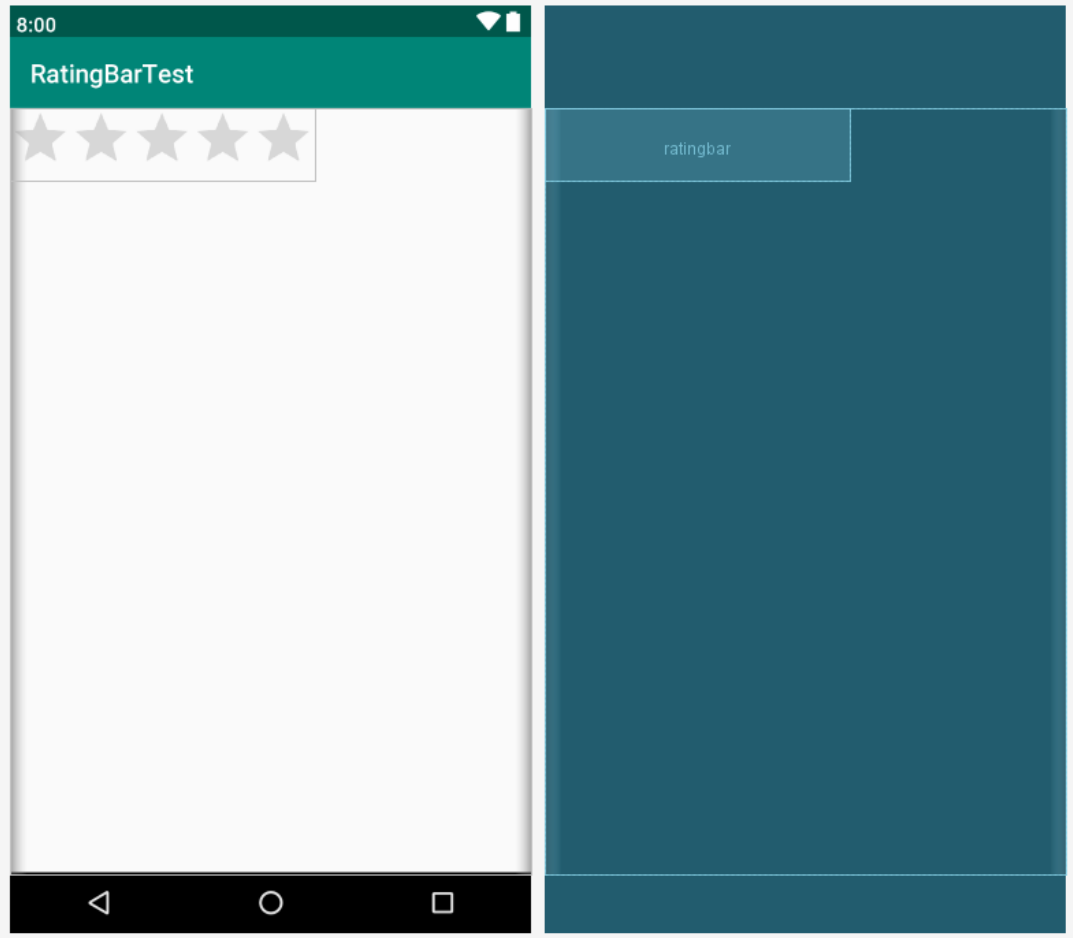

6:53
RatingBarTest
6:54
RatingBarTest
New Rating: 3.0

activity_main.xml



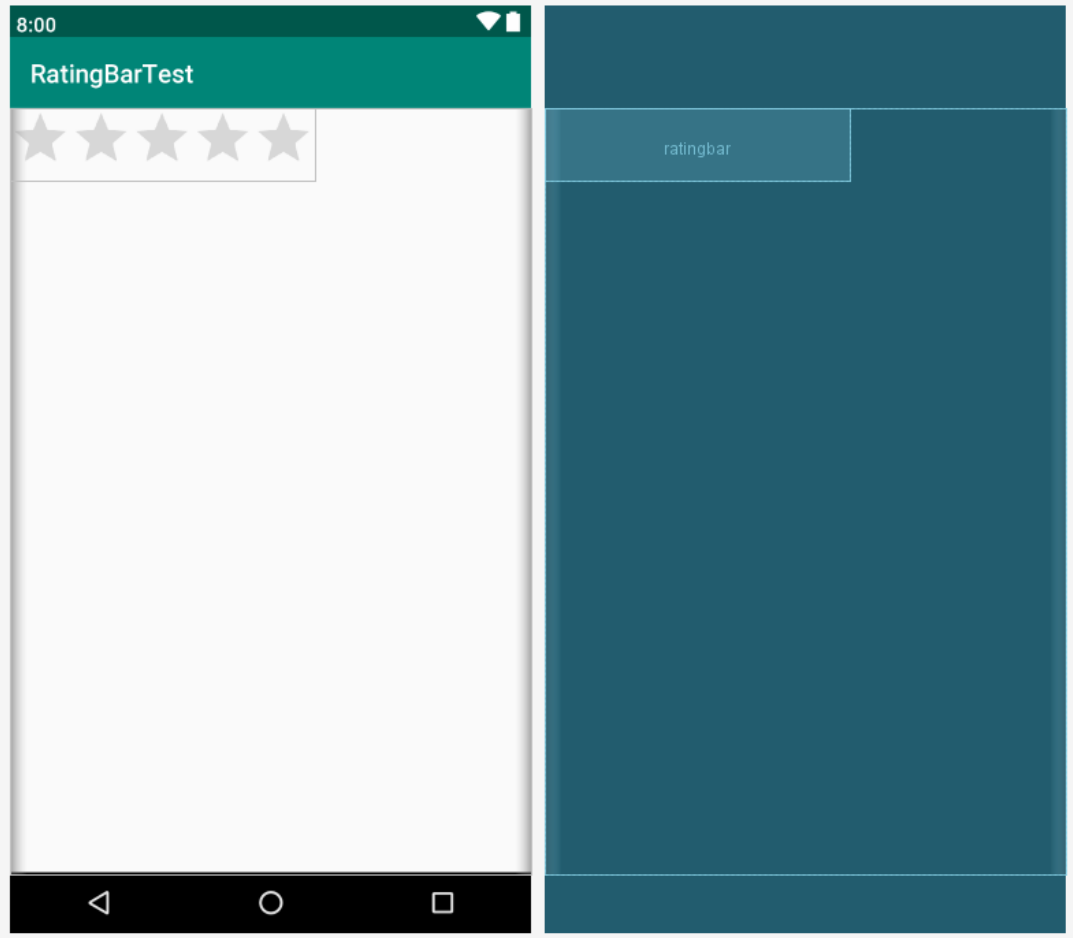
activity_main.xml


```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

    android:layout_width="match_parent"
    android:layout_height="match_parent" >

    <RatingBar
        android:id="@+id/ratingbar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:numStars="5"
        android:stepSize="1.0" />

</LinearLayout>
```



activity_main.xml


```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

    android:layout_width="match_parent"
    android:layout_height="match_parent" >

    <RatingBar
        android:id="@+id/ratingbar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:numStars="5"
        android:stepSize="1.0" />

</LinearLayout>
```



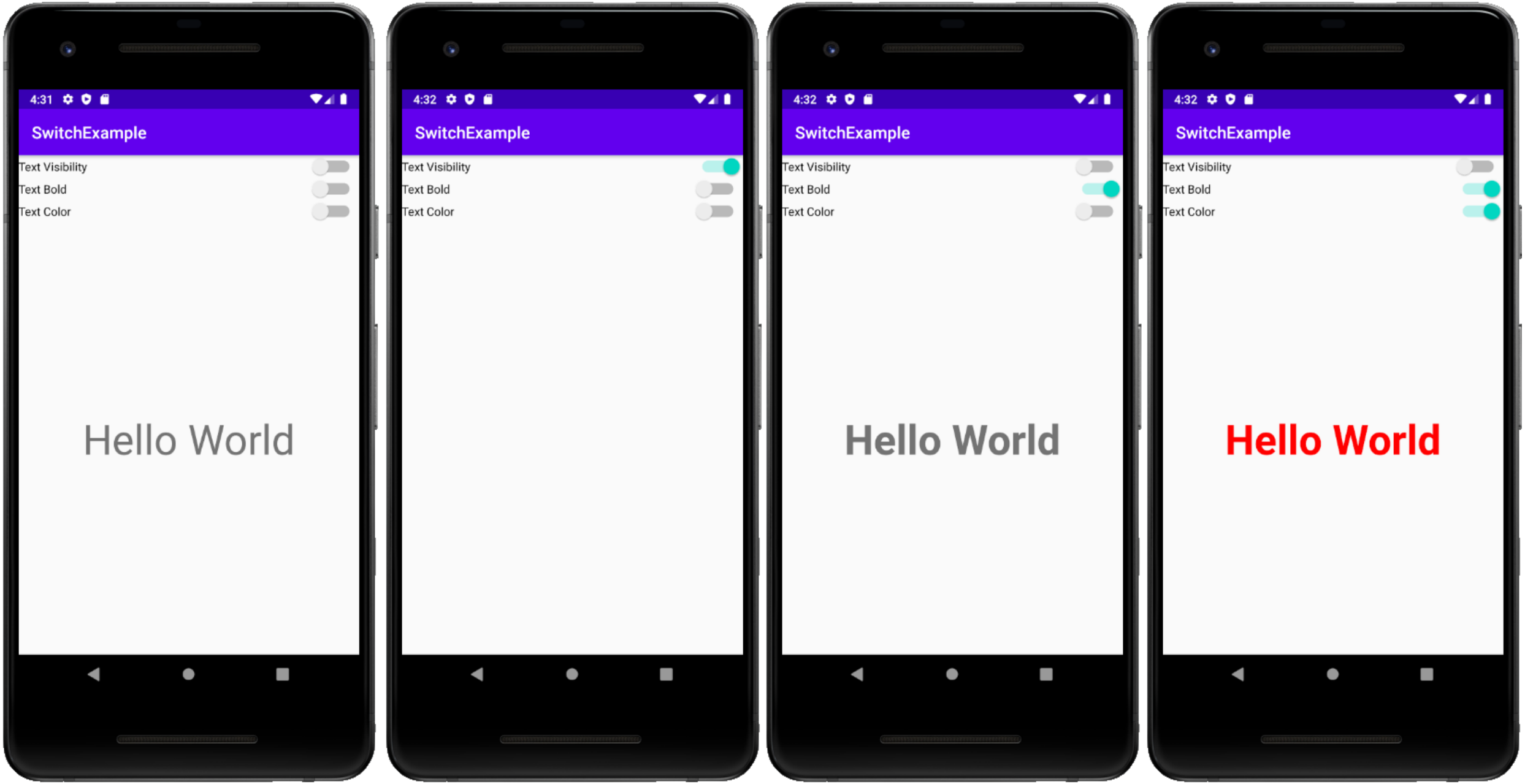
MainActivty.java


```
public class MainActivity extends AppCompatActivity {
    RatingBar ratingbar;
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ratingbar = findViewById(R.id.ratingbar);
        ratingbar.setOnRatingBarChangeListener(new RatingBar.OnRatingBarChangeListener() {
            @Override
            public void onRatingChanged(RatingBar ratingBar, float v, boolean b) {
                Toast.makeText(getApplicationContext(), "New Rating: " + v, Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

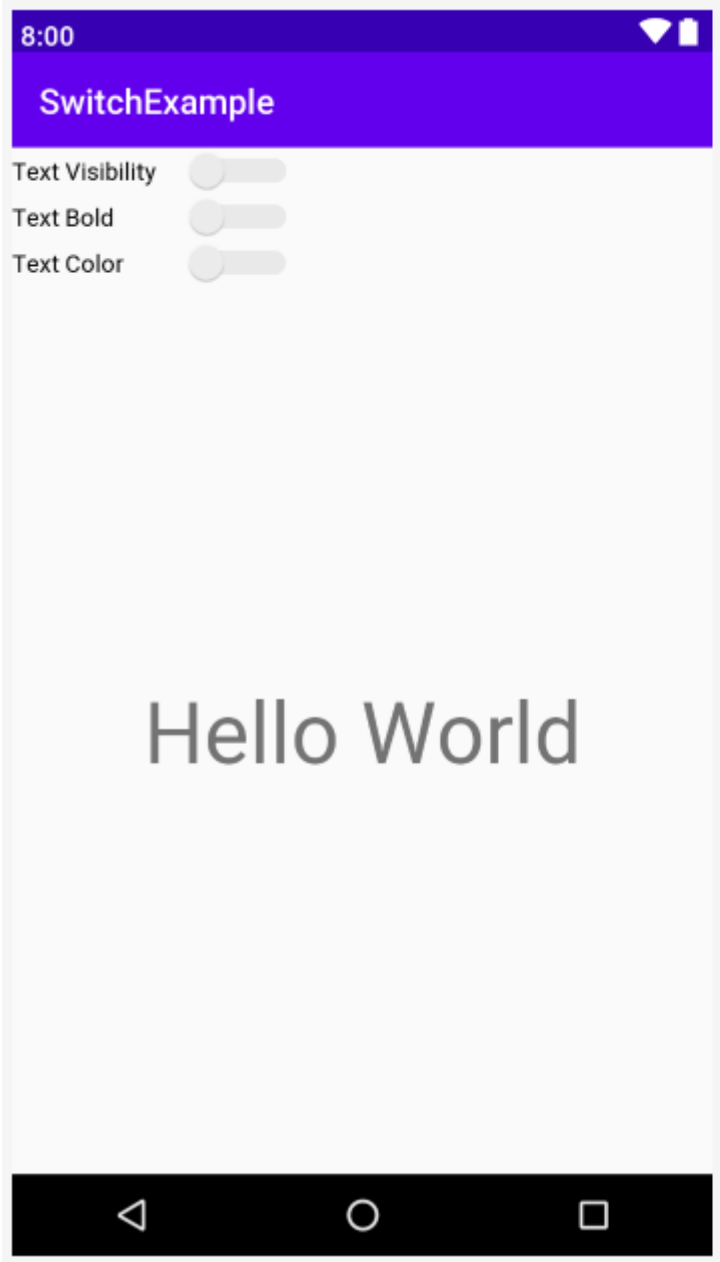
MainActivty.java

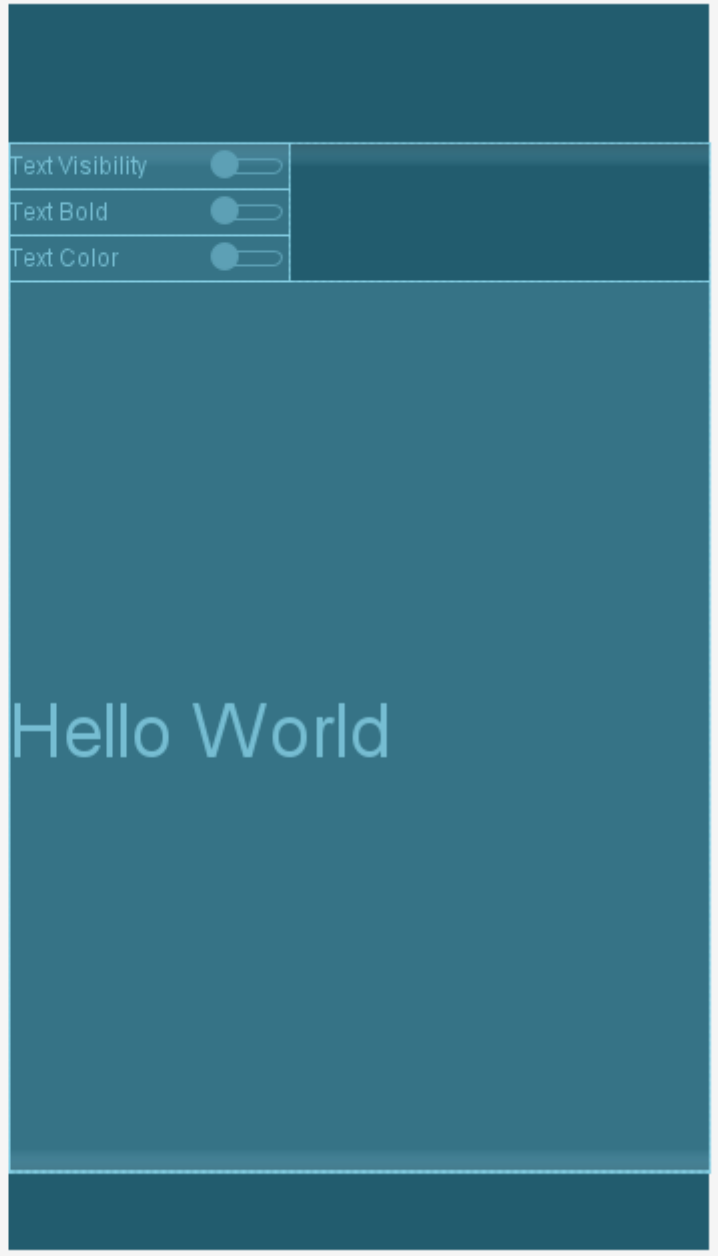

```
public class MainActivity extends AppCompatActivity {
    RatingBar ratingbar;
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ratingbar = findViewById(R.id.ratingbar);
        ratingbar.setOnRatingBarChangeListener(new RatingBar.OnRatingBarChangeListener() {
            @Override
            public void onRatingChanged(RatingBar ratingBar, float v, boolean b) {
                Toast.makeText(getApplicationContext(), "New Rating: " + v, Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

SWITCH

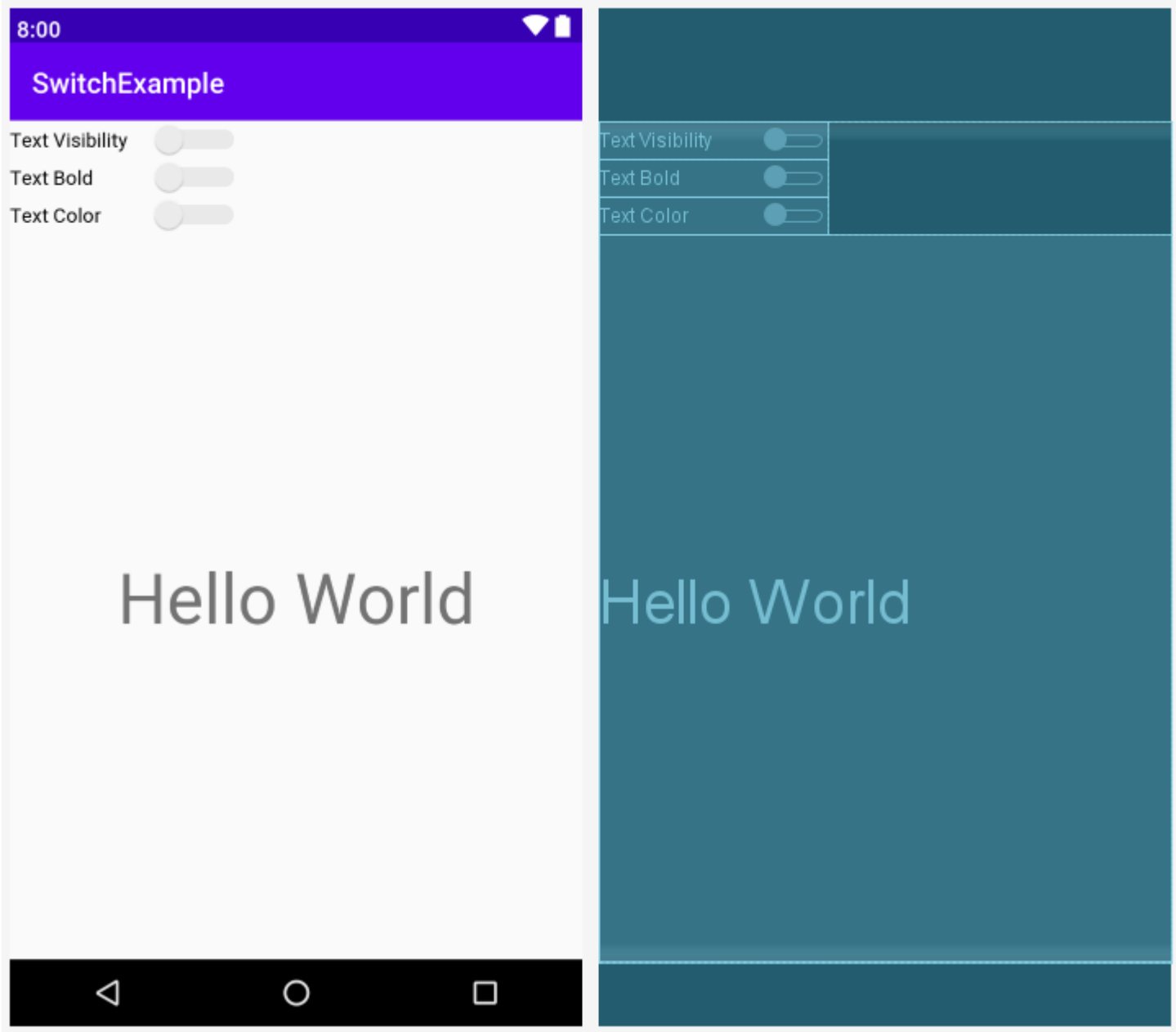

4:31
SwitchExample
Text Visibility
Text Bold
Text Color
Hello World
4:32
SwitchExample
Text Visibility
Text Bold
Text Color
4:32
SwitchExample
Text Visibility
Text Bold
Text Color
Hello World
4:32
SwitchExample
Text Visibility
Text Bold
Text Color
Hello World

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity" >

    <Switch
        android:id="@+id/switch_visibility"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:switchMinWidth="60dp"
        android:switchPadding="20dp"
        android:text="Text Visibility" />

    <Switch
        android:id="@+id/switch_bold"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:switchMinWidth="60dp"
        android:switchPadding="45dp"
        android:text="Text Bold"  />

    <Switch
        android:id="@+id/switch_color"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:switchMinWidth="60dp"
        android:switchPadding="39dp"
        android:text="Text Color"  />

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center_vertical|center_horizontal"
        android:text="Hello World"
        android:textSize="50dp"  />
</LinearLayout>
```





```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity" >

    <Switch
        android:id="@+id/switch_visibility"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:switchMinWidth="60dp"
        android:switchPadding="20dp"
        android:text="Text Visibility" />

    <Switch
        android:id="@+id/switch_bold"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:switchMinWidth="60dp"
        android:switchPadding="45dp"
        android:text="Text Bold"  />

    <Switch
        android:id="@+id/switch_color"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:switchMinWidth="60dp"
        android:switchPadding="39dp"
        android:text="Text Color"  />

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center_vertical|center_horizontal"
        android:text="Hello World"
        android:textSize="50dp"  />
</LinearLayout>
```



https://developer.android.com/reference/android/widget/Switch

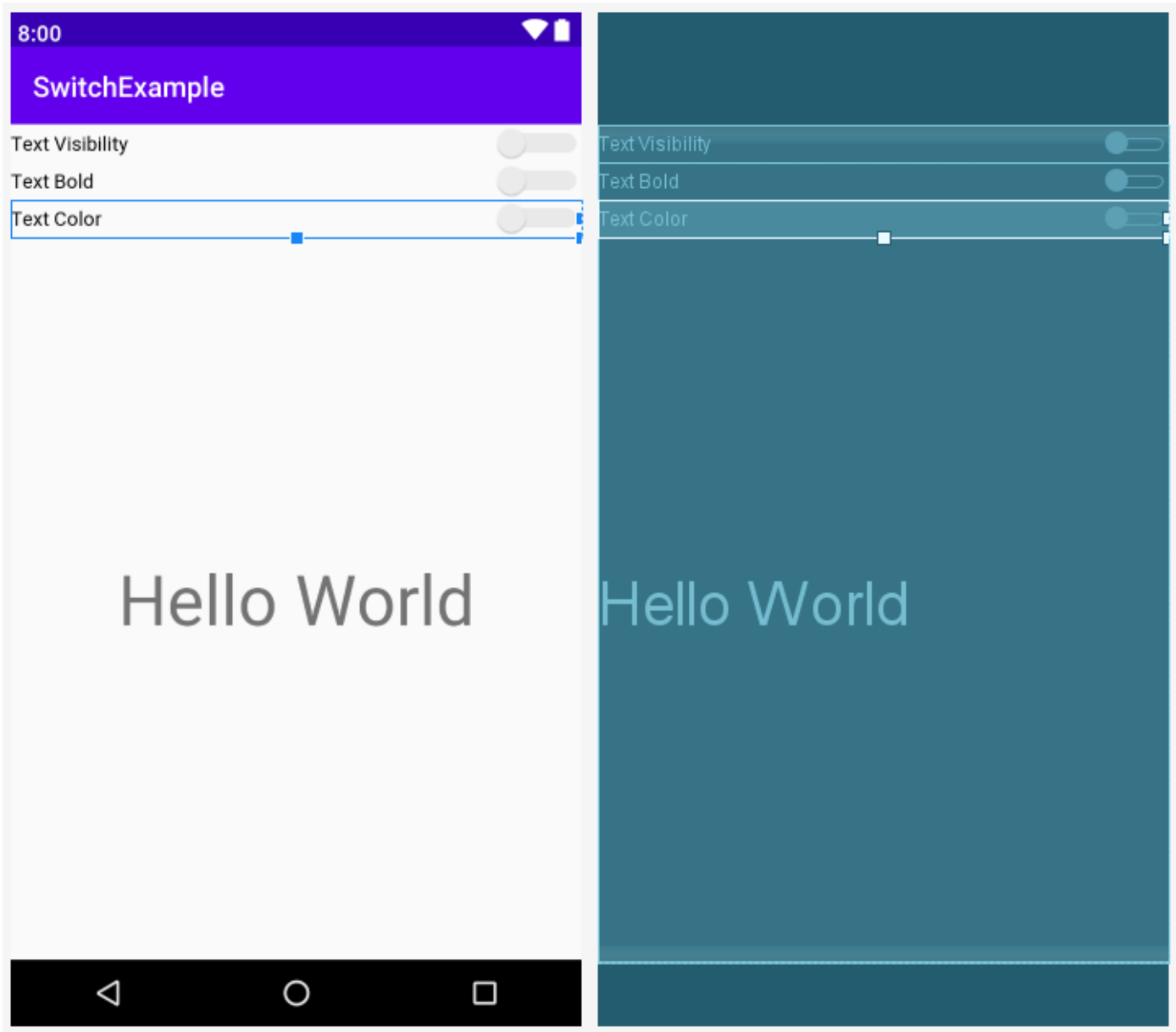
XML attributes	
android:showText	Whether to draw on/off text.
android:splitTrack	Whether to split the track and leave a gap for the thumb drawable.
android:switchMinWidth	Minimum width for the switch component.
android:switchPadding	Minimum space between the switch and caption text.
android:switchTextAppearance	TextAppearance style for text displayed on the switch thumb.
android:textOff	Text to use when the switch is in the unchecked/"off" state.
android:textOn	Text to use when the switch is in the checked/"on" state.
android:textStyle	Style (normal, bold, italic, bold|italic) for the text.
android:thumb	Drawable to use as the "thumb" that switches back and forth.
android:thumbTextPadding	Amount of padding on either side of text within the switch thumb.
android:thumbTint	Tint to apply to the thumb.
android:thumbTintMode	Blending mode used to apply the thumb tint.
android:track	Drawable to use as the "track" that the switch thumb slides within.
android:trackTint	Tint to apply to the track.
android:trackTintMode	Blending mode used to apply the track tint.
android:typeface	Typeface (normal, sans, serif, monospace) for the text.

MainActivity.java


```
public class MainActivity extends AppCompatActivity {
    TextView textView;
    Switch switch_bold;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        textView = findViewById(R.id.textView);
        switch_bold = findViewById(R.id.switch_bold);
        switch_bold.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
            @Override
            public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
                if (isChecked)
                    textView.setTypeface(null, Typeface.BOLD);
                else
                    textView.setTypeface(null, Typeface.NORMAL);
            }
        });
    }
}
```

```java
public class MainActivity extends AppCompatActivity {
    TextView textView;
    Switch switch_visibility, switch_bold, switch_color;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        textView = findViewById(R.id.textView);
        switch_bold = findViewById(R.id.switch_bold);
        switch_bold.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
            @Override
            public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
                if (isChecked)
                    textView.setTypeface(null, Typeface.BOLD);
                else
                    textView.setTypeface(null, Typeface.NORMAL);
            }
        });
        switch_color = findViewById(R.id.switch_color);
        switch_color.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
            @Override
            public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
                if (isChecked)
                    textView.setTextColor(Color.RED);
                else
                    textView.setTextColor(Color.BLACK);
            }
        });
        switch_visibility = findViewById(R.id.switch_visibility);
        switch_visibility.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
            @Override
            public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
                if (isChecked)
                    textView.setVisibility(View.INVISIBLE);
                else
                    textView.setVisibility(View.VISIBLE);
            }
        });
    }
}
```

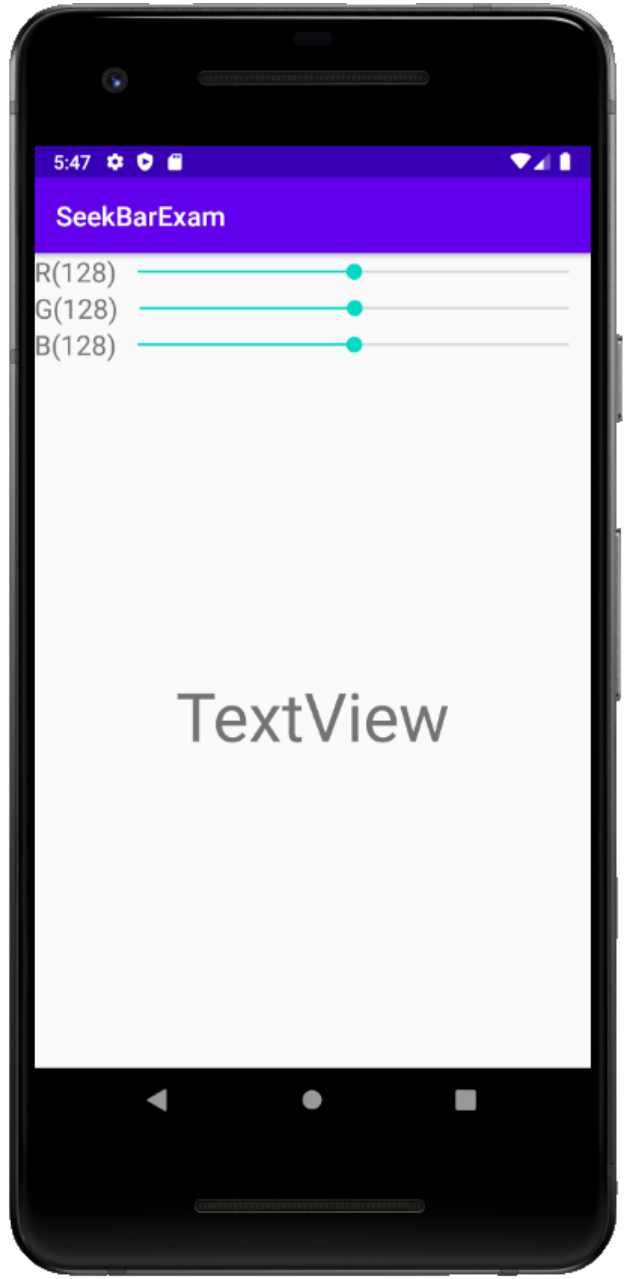
```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity" >

    <Switch
        android:id="@+id/switch_visibility"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:switchMinWidth="60dp"
        android:onClick="onSwitchChanged"
        android:text="Text Visibility" />

    <Switch
        android:id="@+id/switch_bold"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:switchMinWidth="60dp"
        android:onClick="onSwitchChanged"
        android:text="Text Bold"  />

    <Switch
        android:id="@+id/switch_color"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:switchMinWidth="60dp"
        android:onClick="onSwitchChanged"
        android:text="Text Color"  />

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center_vertical|center_horizontal"
        android:text="Hello World"
        android:textSize="50dp"  />
</LinearLayout>
```

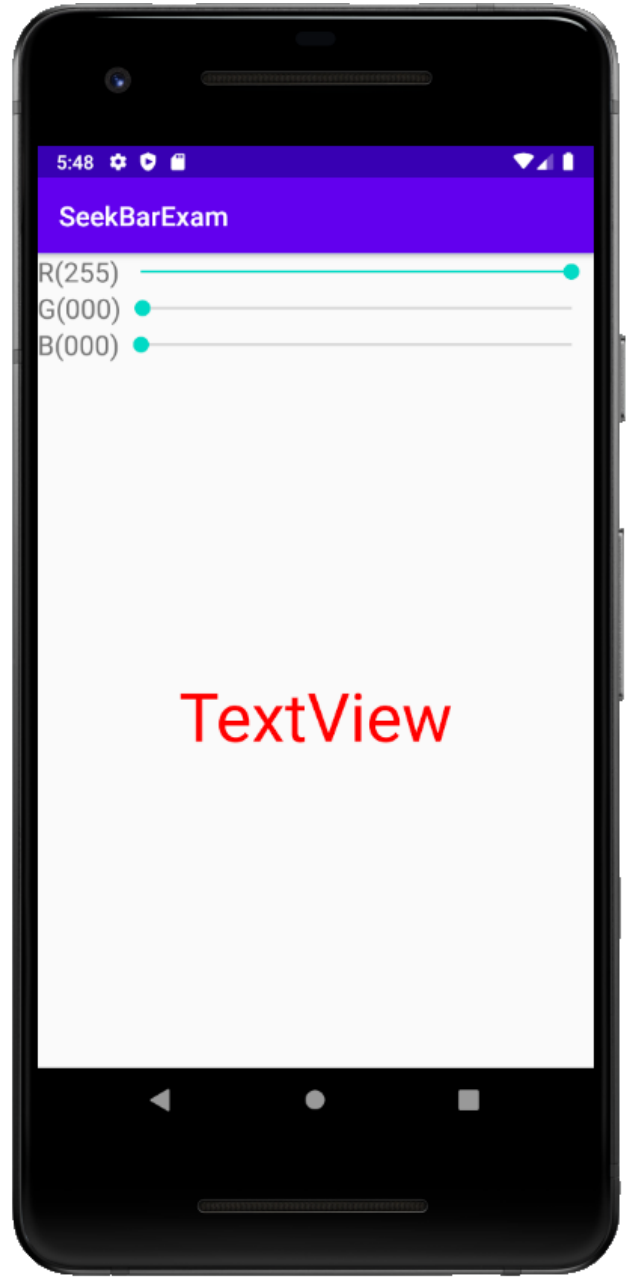


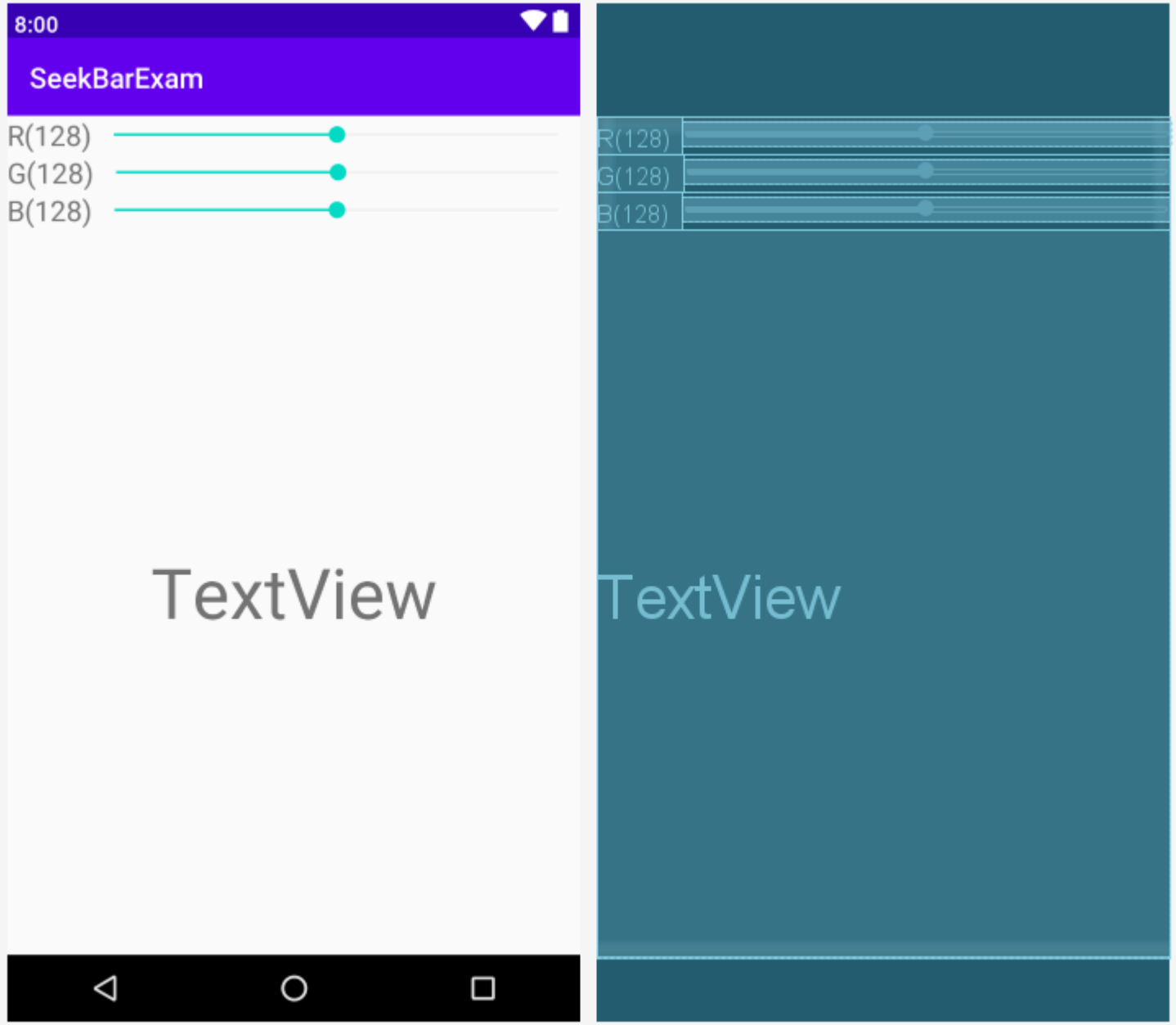
```java
public class MainActivity extends AppCompatActivity {
    TextView textView;

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

    public void onSwitchChanged(View view) {
        boolean checked = ((Switch) view).isChecked();
        switch (view.getId()){
            case R.id.switch_visibility:
                if(checked)
                    textView.setVisibility(view.INVISIBLE);
                else
                    textView.setVisibility(view.VISIBLE);
                break;
            case R.id.switch_bold:
                if(checked)
                    textView.setTypeface(null, Typeface.BOLD);
                else
                    textView.setTypeface(null, Typeface.NORMAL);
                break;
            case R.id.switch_color:
                if(checked)
                    textView.setTextColor(Color.RED);
                else
                    textView.setTextColor(Color.BLACK);
                break;
        }
    }
}
```

1. editText를 추가
2. 글자넣기 버튼을 클릭하면 입력된 내용이 Hello World에 반영됨
3. 스위치를 변경하면 반영됨

SEEKBAR


5:47
SeekBarExam
R(128)
G(128)
B(128)
TextView


5:48
SeekBarExam
R(255)
G(000)
B(000)
TextView


8:00
SeekBarExam
R(128)
G(128)
B(128)
TextView
R(128)
G(128)
B(128)
TextView

setOnSeekBarChangeListener(new SeekBar.OnSeekBar ...)


```
public class MainActivity extends AppCompatActivity {
    SeekBar seekbar_r;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        seekbar_r = findViewById(R.id.seekbar_r);
        seekbar_r.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
            }

            @Override
            public void onStartTrackingTouch(SeekBar seekBar) {
            }

            @Override
            public void onStopTrackingTouch(SeekBar seekBar) {

            }
        });
    }
}
```

```java
public class MainActivity extends AppCompatActivity {
    TextView textView;
    SeekBar seekbar_r;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        textView = findViewById(R.id.textView);
        seekbar_r = findViewById(R.id.seekbar_r);
        seekbar_r.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
                textview_r.setText(String.format("R(%03d)",progress));
                red = progress;
                textView.setTextColor(Color.rgb(red,green,blue));
            }

            @Override
            public void onStartTrackingTouch(SeekBar seekBar) {
            }

            @Override
            public void onStopTrackingTouch(SeekBar seekBar) {

            }
        });
    }
}
```

MainActivity.java


```
public class MainActivity extends AppCompatActivity {
    TextView textView;
    SeekBar seekbar_r, seekbar_g, seekbar_b;
    TextView textview_r, textview_g, textview_b;
    int red=128,green=128,blue=128;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        textView = findViewById(R.id.textView);
        textview_r = findViewById(R.id.textview_r);
        textview_g = findViewById(R.id.textview_g);
        textview_b = findViewById(R.id.textview_b);

        seekbar_r = findViewById(R.id.seekbar_r);
        seekbar_r.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
                textview_r.setText(String.format("R(%03d)",progress));
                red = progress;
                textView.setTextColor(Color.rgb(red,green,blue));
            }

            @Override
            public void onStartTrackingTouch(SeekBar seekBar) {
            }

            @Override
            public void onStopTrackingTouch(SeekBar seekBar) {

            }
        });
```

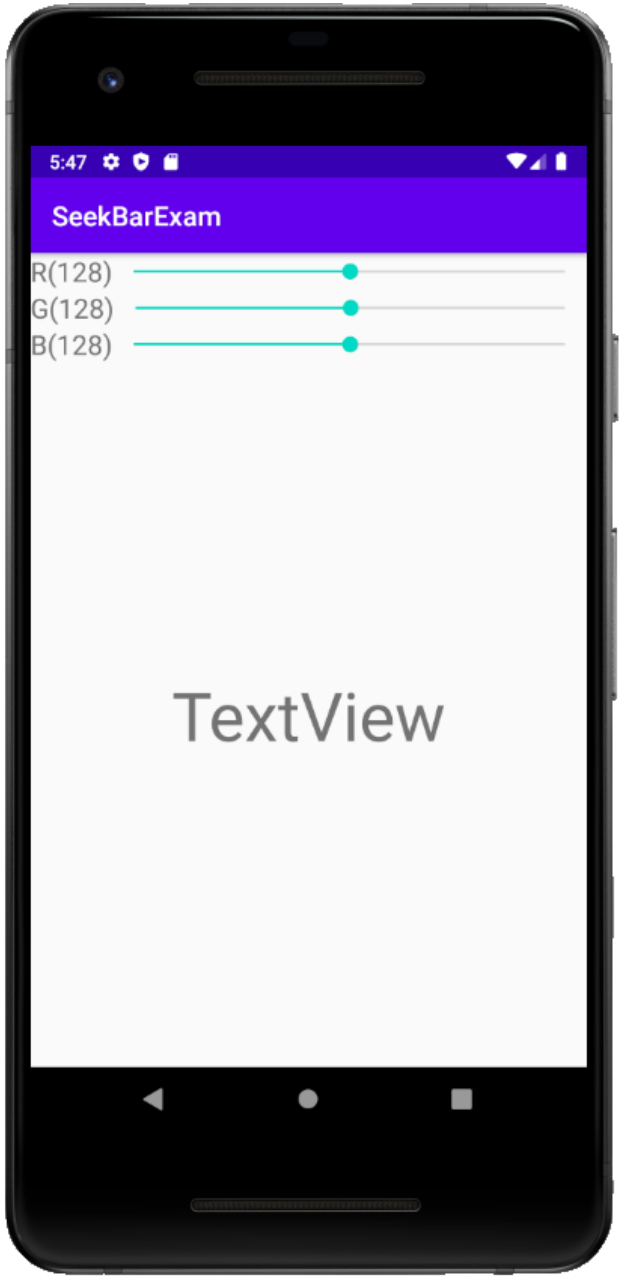
MainActivity.java


```
seekbar_g = findViewById(R.id.seekbar_g);
seekbar_g.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
    @Override
    public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
        textview_g.setText(String.format("G(%03d)",progress));
        green = progress;
        textView.setTextColor(Color.rgb(red,green,blue));
    }

    @Override
    public void onStartTrackingTouch(SeekBar seekBar) {

    }

    @Override
    public void onStopTrackingTouch(SeekBar seekBar) {

    }
});
```

MainActivity.java


```
        seekbar_b = findViewById(R.id.seekbar_b);
        seekbar_b.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
                textview_b.setText(String.format("B(%03d)",progress));
                blue = progress;
                textView.setTextColor(Color.rgb(red,green,blue));
            }

            @Override
            public void onStartTrackingTouch(SeekBar seekBar) {

            }

            @Override
            public void onStopTrackingTouch(SeekBar seekBar) {

            }
        });
    }


}
```

추가작업

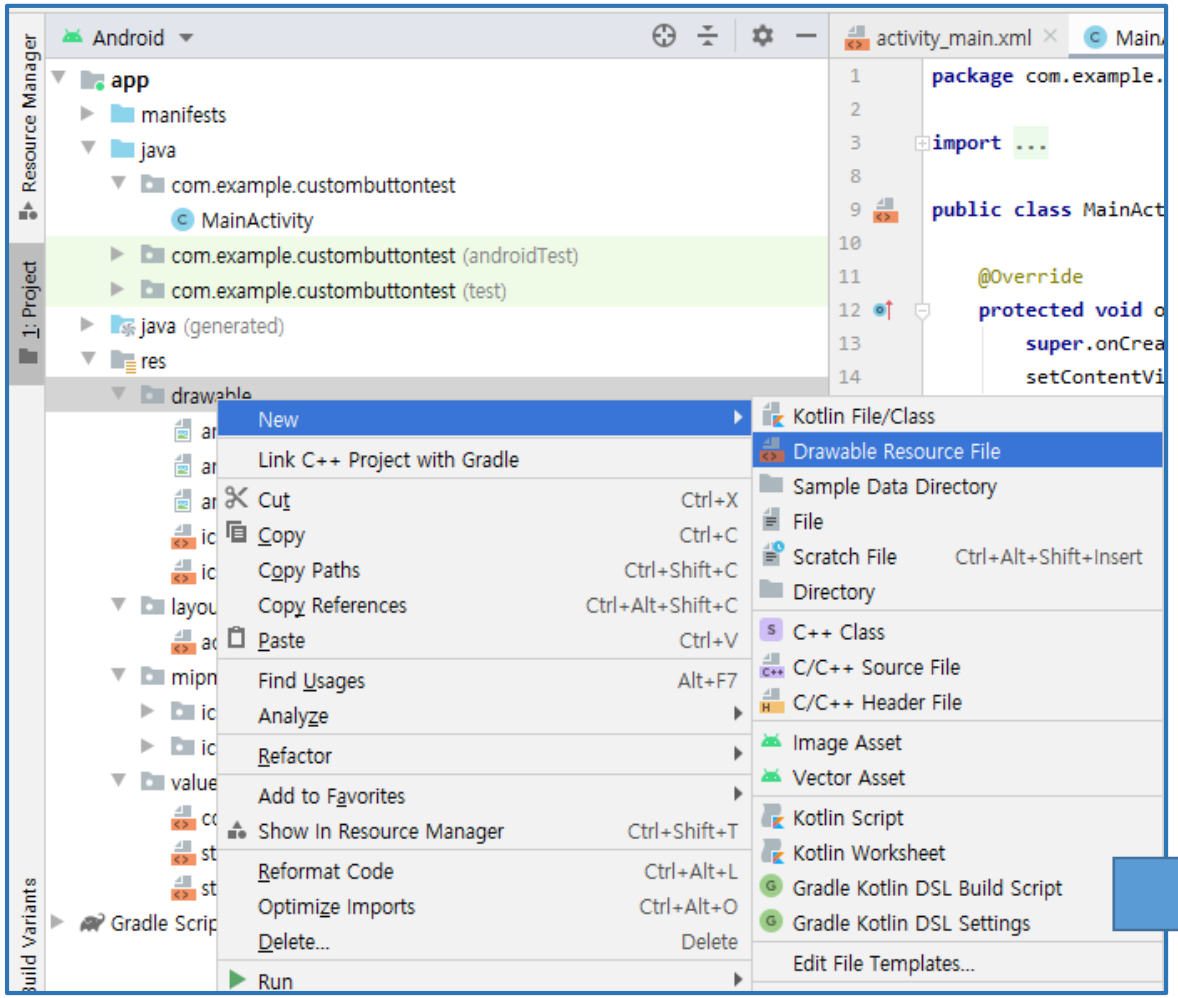


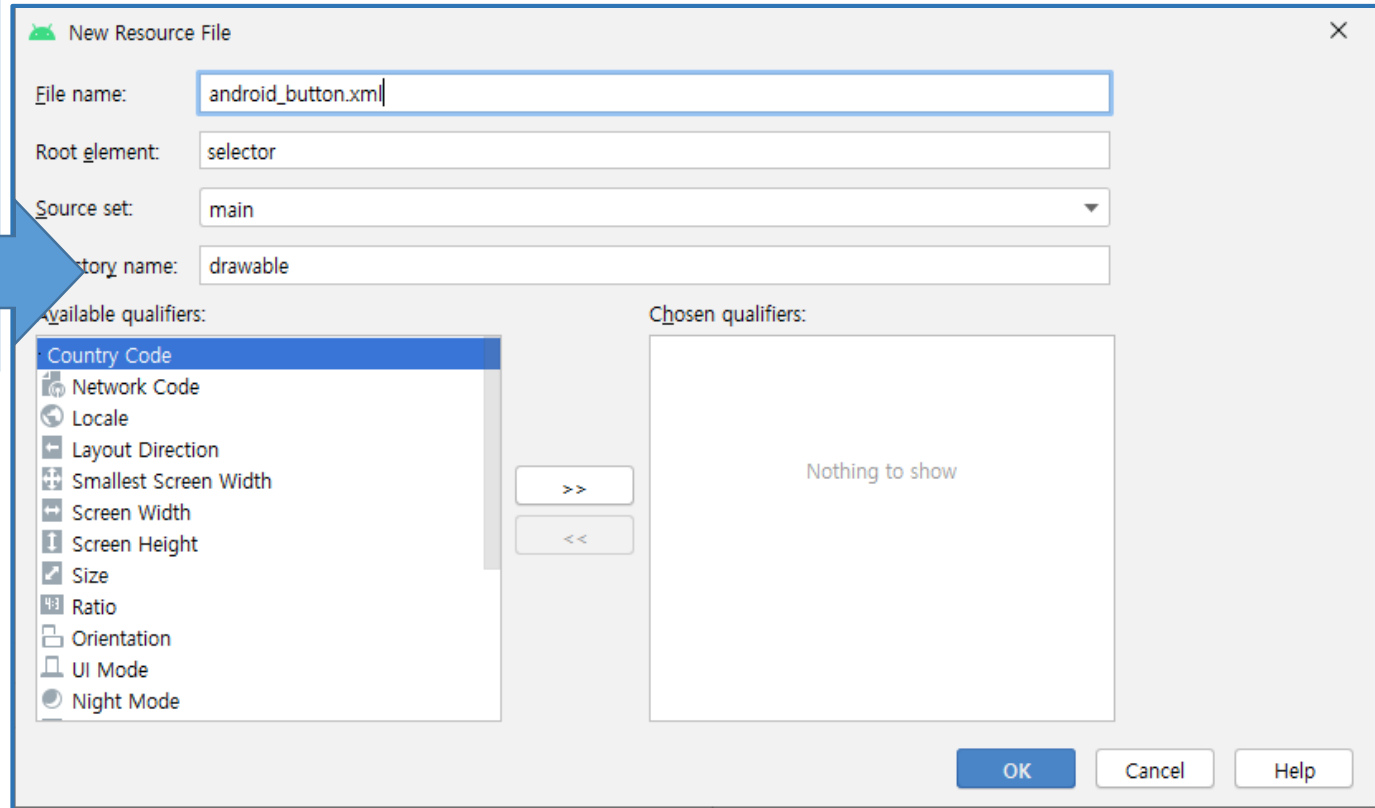
1. 시크바 좌우에 버튼을 생성
2. 버튼을 누르면 -10 / +10씩 증가시키고 증가분을 반영
3. 값이 10보다 작을경우 좌측버튼을 누르면 0
4. 값이 245보다 클경우 우측버튼을 누르면 255

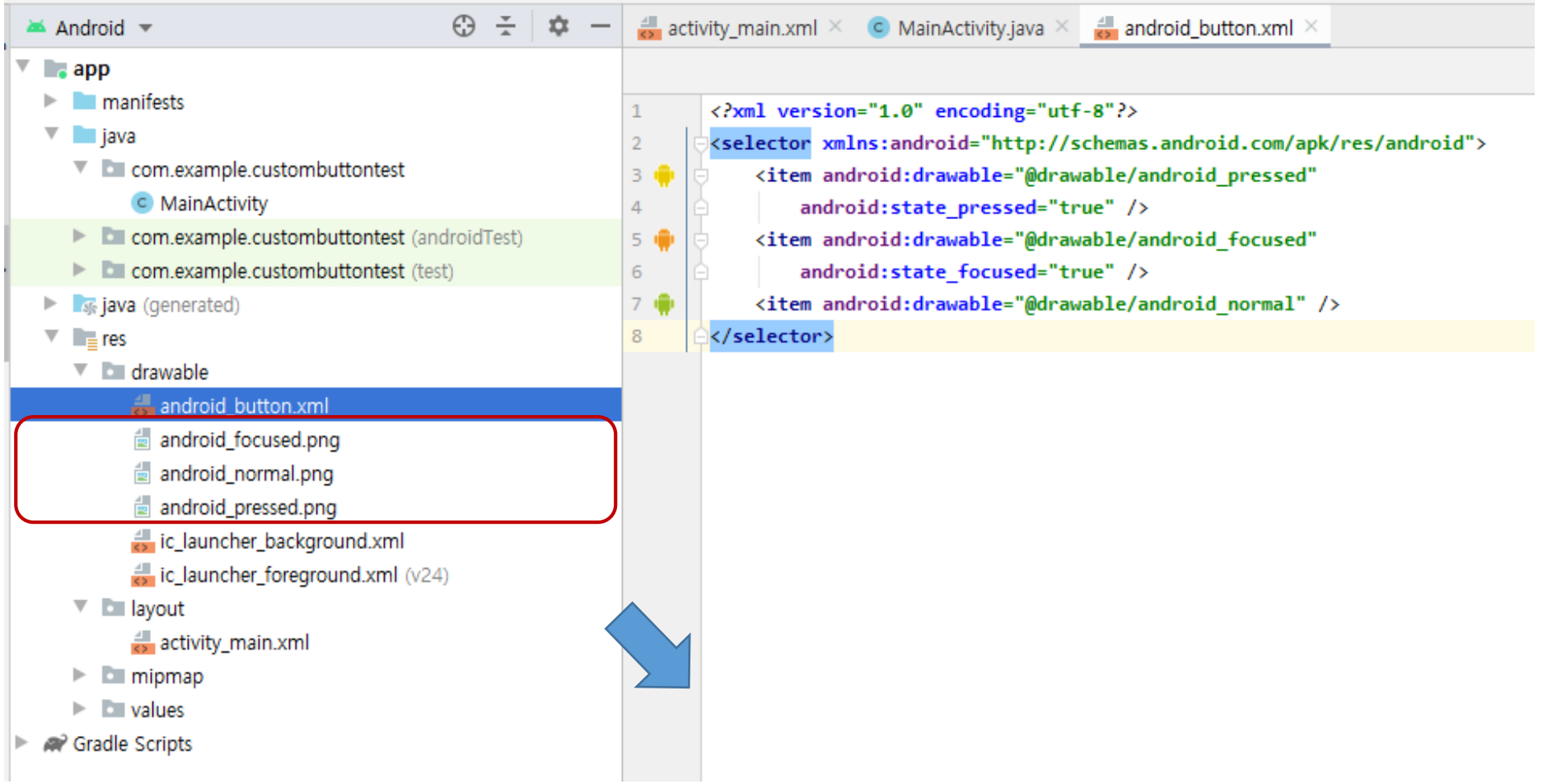
CUSTOM BUTTON


7:20
CustomButtonTest


7:22
CustomButtonTest
클릭


Android
app
manifests
java
com.example.custombuttontest
MainActivity
com.example.custombuttontest (androidTest)
com.example.custombuttontest (test)
java (generated)
res
drawable
activity_main.xml
package com.example.
import ...
public class MainAct
@Override
protected void o
super.onCrea
setContentVi
New
Link C++ Project with Gradle
Cut Ctrl+X
Copy Ctrl+C
Copy Paths Ctrl+Shift+C
Copy References Ctrl+Alt+Shift+C
Paste Ctrl+V
Find Usages Alt+F7
Analyze
Refactor
Add to Favorites
Show In Resource Manager Ctrl+Shift+T
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Delete... Delete
Run
Kotlin File/Class
Drawable Resource File
Sample Data Directory
File
Scratch File Ctrl+Alt+Shift+Insert
Directory
C++ Class
C/C++ Source File
C/C++ Header File
Image Asset
Vector Asset
Kotlin Script
Kotlin Worksheet
Gradle Kotlin DSL Build Script
Gradle Kotlin DSL Settings
Edit File Templates...
Resource Manager
1: Project
Build Variants


New Resource File
File name: android_button.xml
Root element: selector
Source set: main
Directory name: drawable
Available qualifiers:
Country Code
Network Code
Locale
Layout Direction
Smallest Screen Width
Screen Width
Screen Height
Size
Ratio
Orientation
UI Mode
Night Mode
>>
<<
Chosen qualifiers:
Nothing to show
OK
Cancel
Help



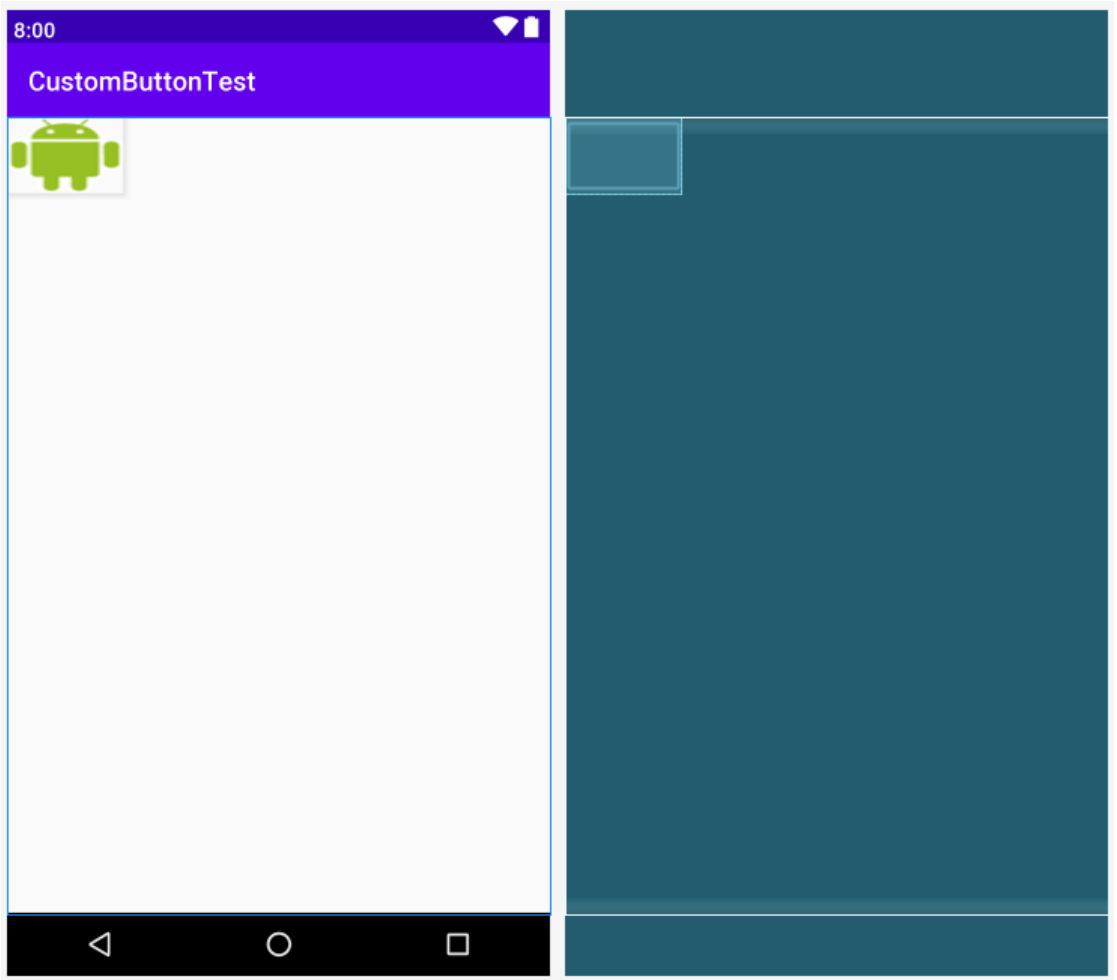

```xml
<?xml version="1.0" encoding="utf-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:drawable="@drawable/android_pressed"
        android:state_pressed="true" />
    <item android:drawable="@drawable/android_focused"
        android:state_focused="true" />
    <item android:drawable="@drawable/android_normal" />
</selector>
```

android_button.xml


```xml
<?xml version="1.0" encoding="utf-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:drawable="@drawable/android_pressed"
        android:state_pressed="true" />
    <item android:drawable="@drawable/android_focused"
        android:state_focused="true" />
    <item android:drawable="@drawable/android_normal" />
</selector>
```

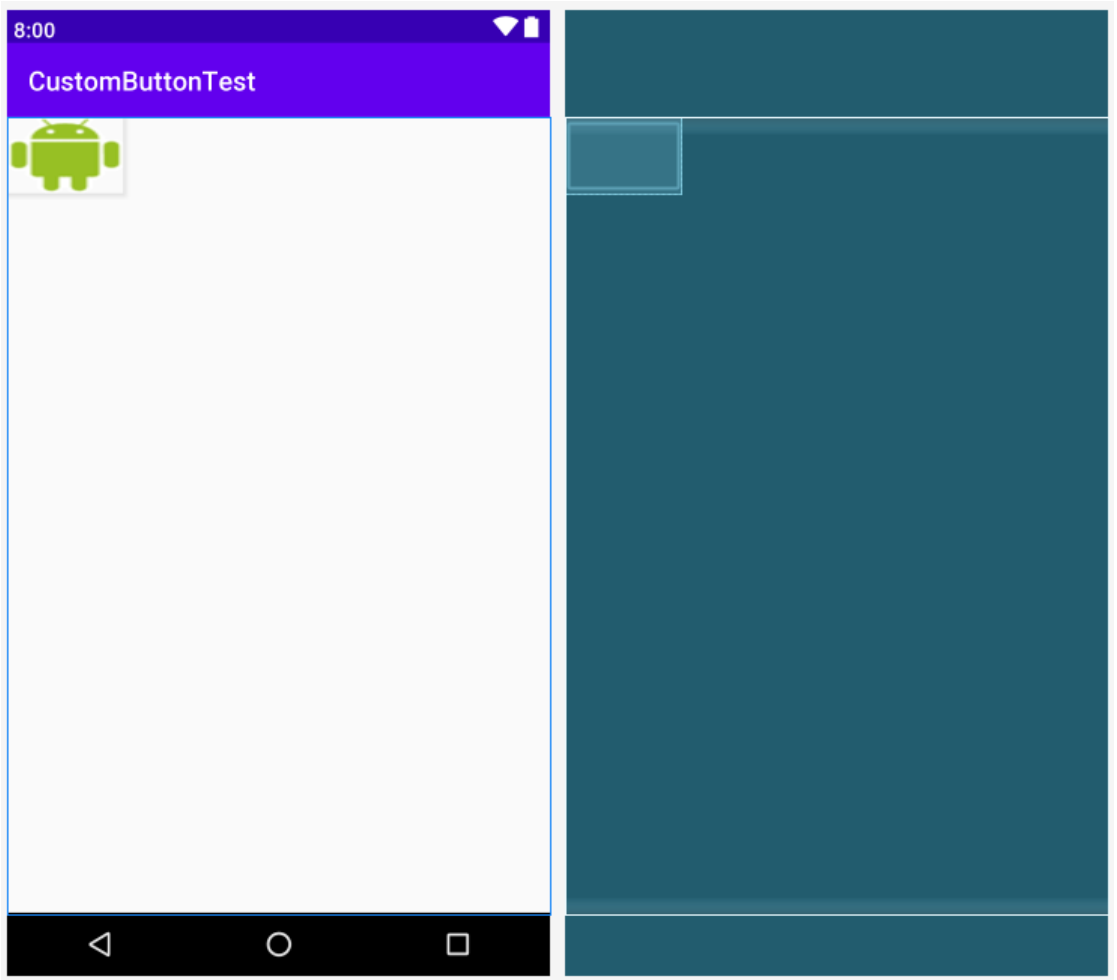
activity_main.xml


```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity" >

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="onClick"
        android:background="@drawable/android_button" />
</LinearLayout>
```



activity_main.xml

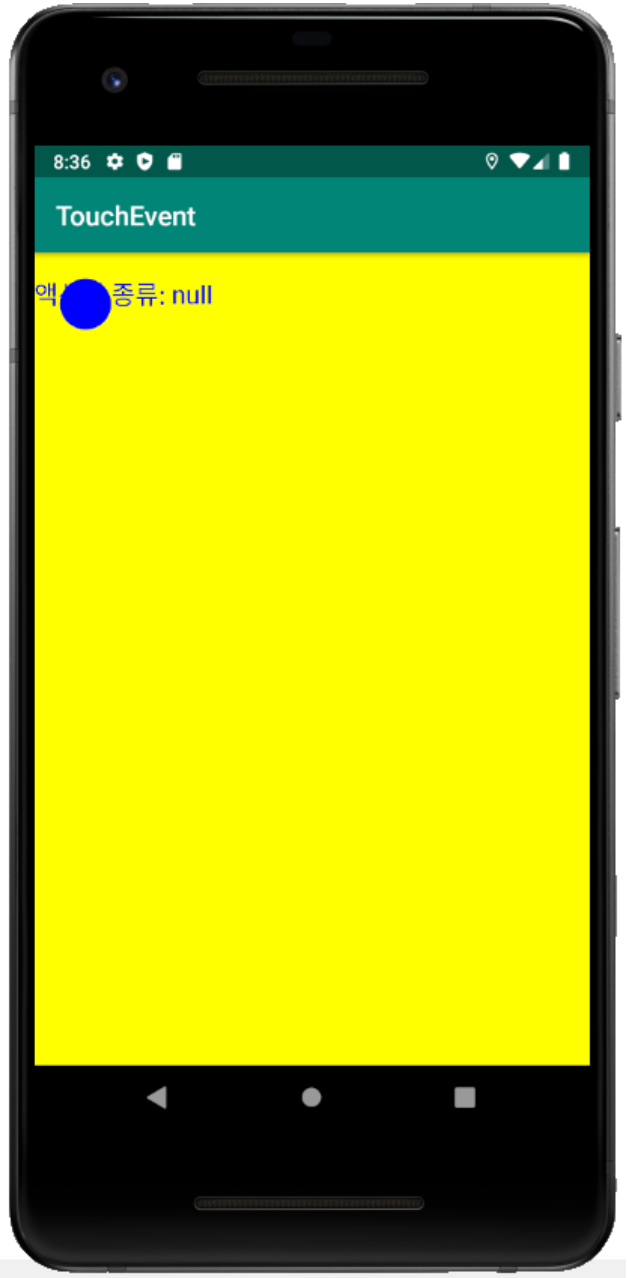

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity" >

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="onClick"
        android:background="@drawable/android_button" />
</LinearLayout>
```

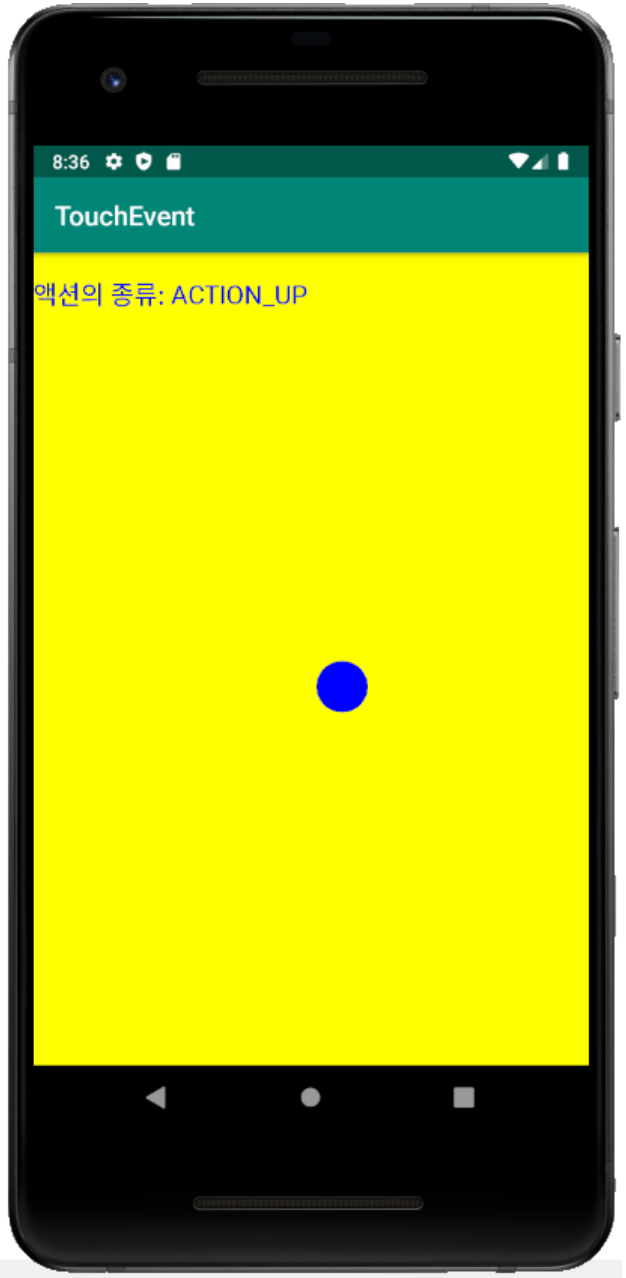


MainActivity.java


```
public class MainActivity extends AppCompatActivity {

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

    public void onClick(View view) {
        Toast.makeText(getApplicationContext(),"클릭",Toast.LENGTH_SHORT).show();
    }
}
```

ONTOUCH EVENT


8:36
TouchEvent
종류: null


8:36
TouchEvent
액션의 종류: ACTION_UP

```
public class MainActivity extends AppCompatActivity {
    protected class MyView extends View {
        int x = 100, y = 100;

        String str;
        public MyView(Context context) {
            super(context);
            setBackgroundColor(Color.YELLOW);     //setBackgroundColor(Color.rgb(200,200,200));
        }

        @Override
        protected void onDraw(Canvas canvas) {
            Paint paint = new Paint();              // paint.setColor(Color.BLUE);
            paint.setColor(Color.rgb(0,0,255));
            canvas.drawCircle(x, y, 50, paint);
            paint.setTextSize(50);
            canvas.drawText("액션의 종류: " + str, 0, 100, paint);
        }

        @Override
        public boolean onTouchEvent(MotionEvent event) {
            x = (int) event.getX();
            y = (int) event.getY();

            if (event.getAction() == MotionEvent.ACTION_UP)
                str = "ACTION_UP";
            if (event.getAction() == MotionEvent.ACTION_DOWN)
                str = "ACTION_DOWN";
            if (event.getAction() == MotionEvent.ACTION_MOVE)
                str = "ACTION_MOVE";

            invalidate();
            return true;
        }
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        MyView myView = new MyView(this);
        setContentView(myView);
    }
}
```

MainActivity.java


```
public class MainActivity extends AppCompatActivity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        MyView myView = new MyView(this);
        setContentView(myView);
    }
}
```

MainActivity.java


```
protected class MyView extends View {
        int x = 100, y = 100;

        String str;
        public MyView(Context context) {
            super(context);
            setBackgroundColor(Color.YELLOW);
        }

        @Override
        protected void onDraw(Canvas canvas) {
            Paint paint = new Paint();      // paint.setColor(Color.BLUE);
            paint.setColor(Color.rgb(0,0,255));
            canvas.drawCircle(x, y, 50, paint);
            paint.setTextSize(50);
            canvas.drawText("액션의 종류: " + str, 0, 100, paint);
        }
```

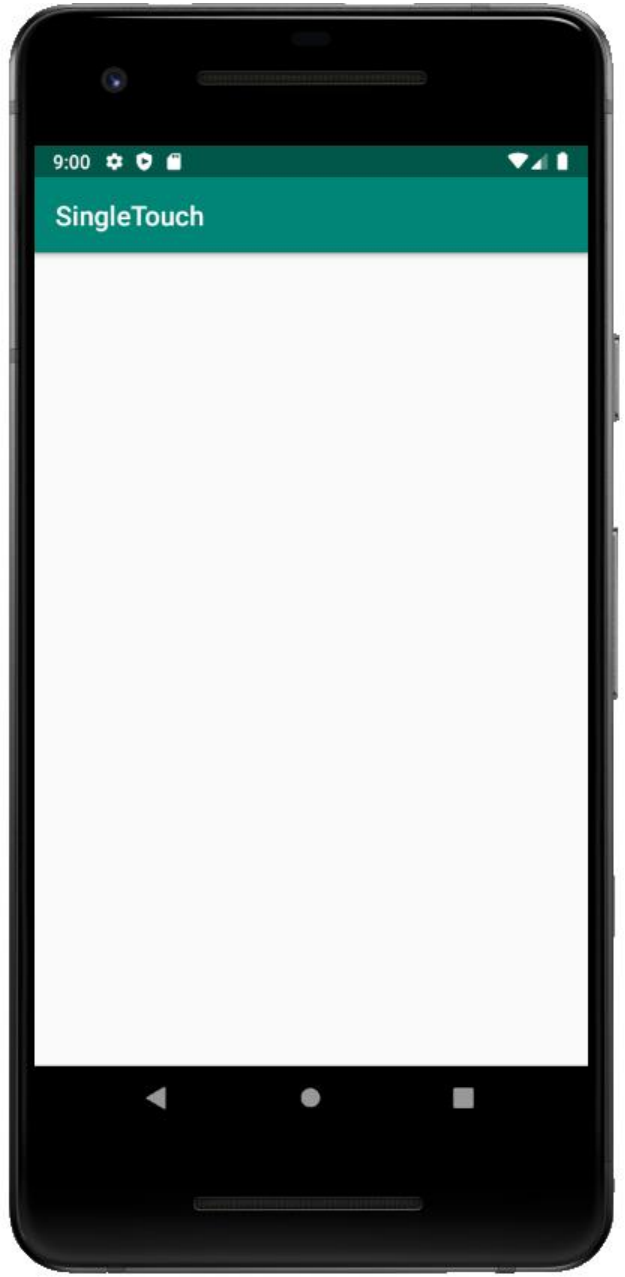
MainActivity.java

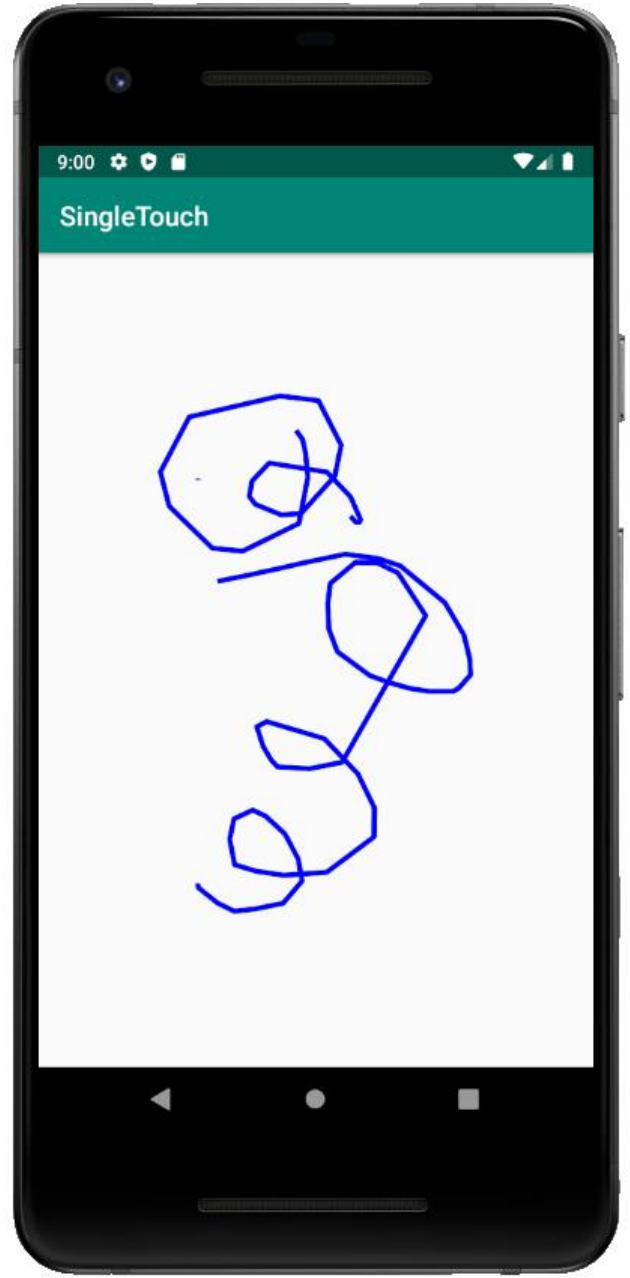

```
    @Override
    public boolean onTouchEvent(MotionEvent event) {
        x = (int) event.getX();
        y = (int) event.getY();

        if (event.getAction() == MotionEvent.ACTION_UP)
            str = "ACTION_UP";
        if (event.getAction() == MotionEvent.ACTION_DOWN)
            str = "ACTION_DOWN";
        if (event.getAction() == MotionEvent.ACTION_MOVE)
            str = "ACTION_MOVE";

        invalidate();
        return true;
    }
}
```

액션	설명
ACTION_DOWN	누르는 동작이 시작
ACTION_UP	누르고 있다가 땠을때 발생
ACTION_MOVE	누르는 도중 움직임
ACTION_CANCEL	터치 동작이 취소
ACTION_OUTSIDE	터치가 현재의 위젯을 벗어남

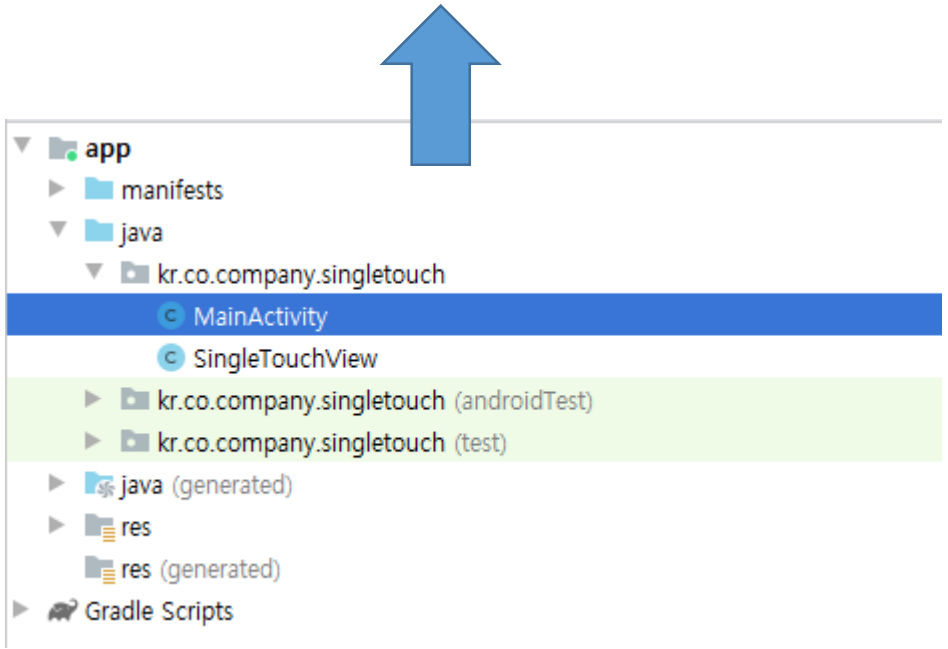
```
public class MainActivity extends AppCompatActivity {
    protected class MyView extends View {
        int x = 100, y = 100;

        String str;
        public MyView(Context context) {
            super(context);
            setBackgroundColor(Color.YELLOW);     //setBackgroundColor(Color.rgb(200,200,200));
        }

        @Override
        protected void onDraw(Canvas canvas) {
            Paint paint = new Paint();              // paint.setColor(Color.BLUE);
            paint.setColor(Color.rgb(0,0,255));
            canvas.drawCircle(x, y, 50, paint);
            paint.setTextSize(50);
            canvas.drawText("액션의 종류: " + str, 0, 100, paint);
        }

        @Override
        public boolean onTouchEvent(MotionEvent event) {
            x = (int) event.getX();
            y = (int) event.getY();

            if (event.getAction() == MotionEvent.ACTION_UP)
                str = "ACTION_UP";
            if (event.getAction() == MotionEvent.ACTION_DOWN)
                str = "ACTION_DOWN";
            if (event.getAction() == MotionEvent.ACTION_MOVE)
                str = "ACTION_MOVE";

            invalidate();
            return true;
        }
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        MyView myView = new MyView(this);
        setContentView(myView);
    }
}
```

SINGLE TOUCH


9:00
SingleTouch

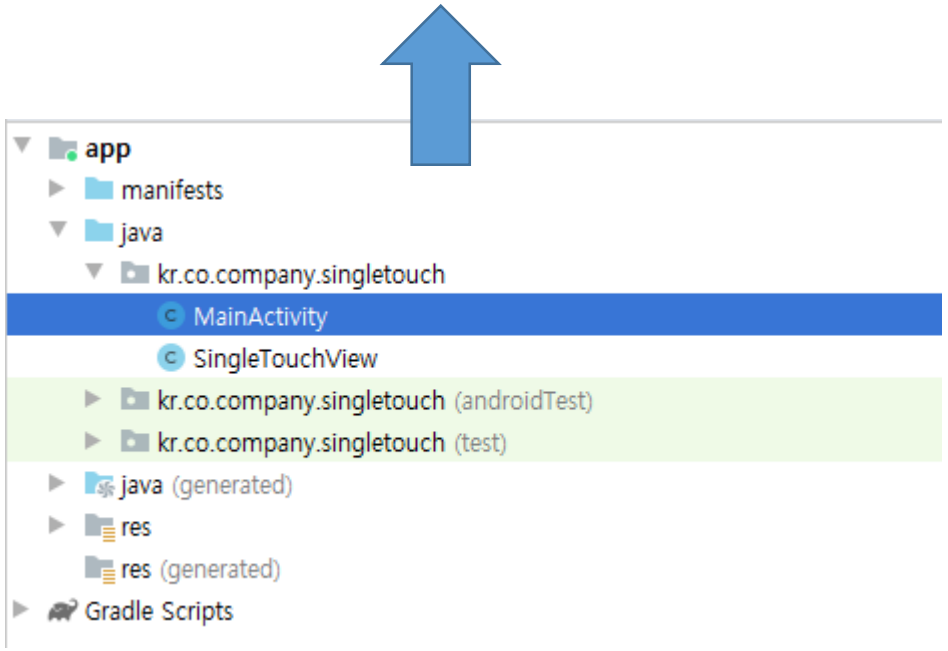

9:00
SingleTouch

```java
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new SingleTouchView(this, null));
    }
}
```




```java
public class SingleTouchView extends View {
    private Paint paint = new Paint();
    private Path path = new Path();

    public SingleTouchView(Context context, AttributeSet attrs) {
        super(context, attrs);
        paint.setAntiAlias(true);
        paint.setStrokeWidth(10f);
        paint.setColor(Color.BLUE);
        paint.setStyle(Paint.Style.STROKE);
        paint.setStrokeJoin(Paint.Join.ROUND);
    }

    @Override
    protected void onDraw(Canvas canvas) {
        canvas.drawPath(path, paint);
    }

    @Override
    public boolean onTouchEvent(MotionEvent event) {
        float eventX = event.getX();
        float eventY = event.getY();

        switch (event.getAction()) {
            case MotionEvent.ACTION_DOWN:
                path.moveTo(eventX, eventY);
                return true;
            case MotionEvent.ACTION_MOVE:
                path.lineTo(eventX, eventY);
                break;
            case MotionEvent.ACTION_UP:
                break;
            default:
                return false;
        }

        invalidate();
        return true;
    }
}
```

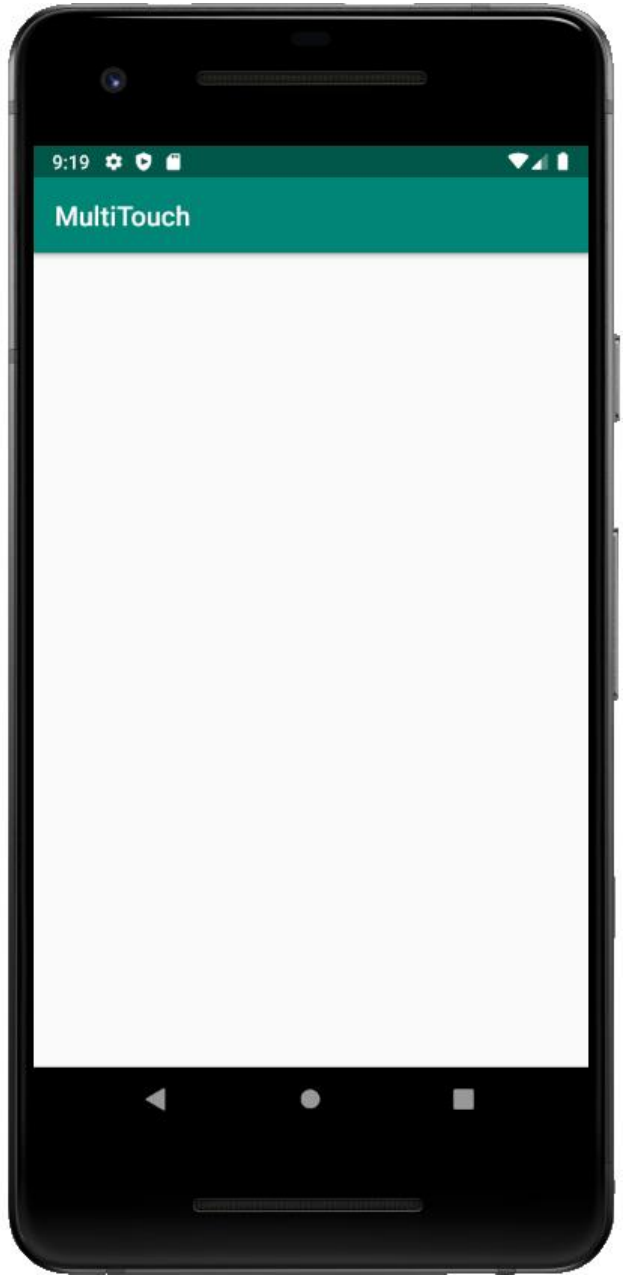
```java
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new SingleTouchView(this, null));
    }
}
```

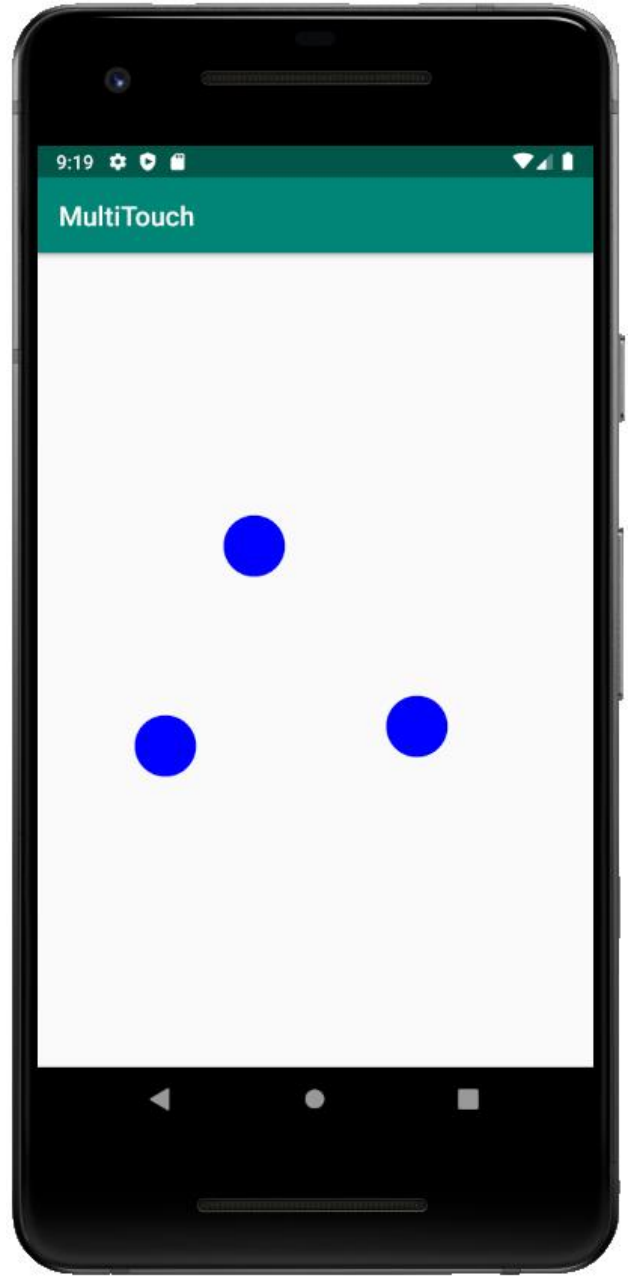



```java
public class SingleTouchView extends View {
    private Paint paint = new Paint();
    private Path path = new Path();

    public SingleTouchView(Context context, AttributeSet attrs) {
        super(context, attrs);
        paint.setAntiAlias(true);
        paint.setStrokeWidth(10f);
        paint.setColor(Color.BLUE);
        paint.setStyle(Paint.Style.STROKE);
        paint.setStrokeJoin(Paint.Join.ROUND);
    }

    @Override
    protected void onDraw(Canvas canvas) {
        canvas.drawPath(path, paint);
    }

    @Override
    public boolean onTouchEvent(MotionEvent event) {
        float eventX = event.getX();
        float eventY = event.getY();

        switch (event.getAction()) {
            case MotionEvent.ACTION_DOWN:
                path.moveTo(eventX, eventY);
                return true;
            case MotionEvent.ACTION_MOVE:
                path.lineTo(eventX, eventY);
                break;
            case MotionEvent.ACTION_UP:
                break;
            default:
                return false;
        }

        invalidate();
        return true;
    }
}
```

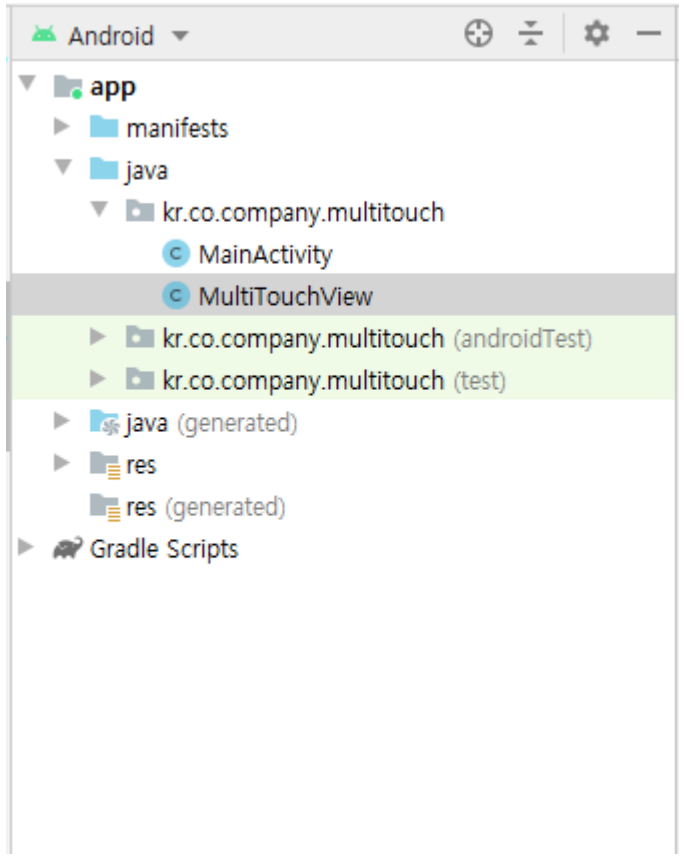
`moveTo(x, y)`
펜을 x와 y 로 지정된 좌표로 옮깁니다.

`lineTo(x, y)`
현재의 드로잉 위치에서 x와 y로 지정된 위치까지 선을 그립니다.


9:19
MultiTouch


9:19
MultiTouch

MainActivity.java




```java
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new MultiTouchView(this, null));
    }
}
```

MultiTouchView.java

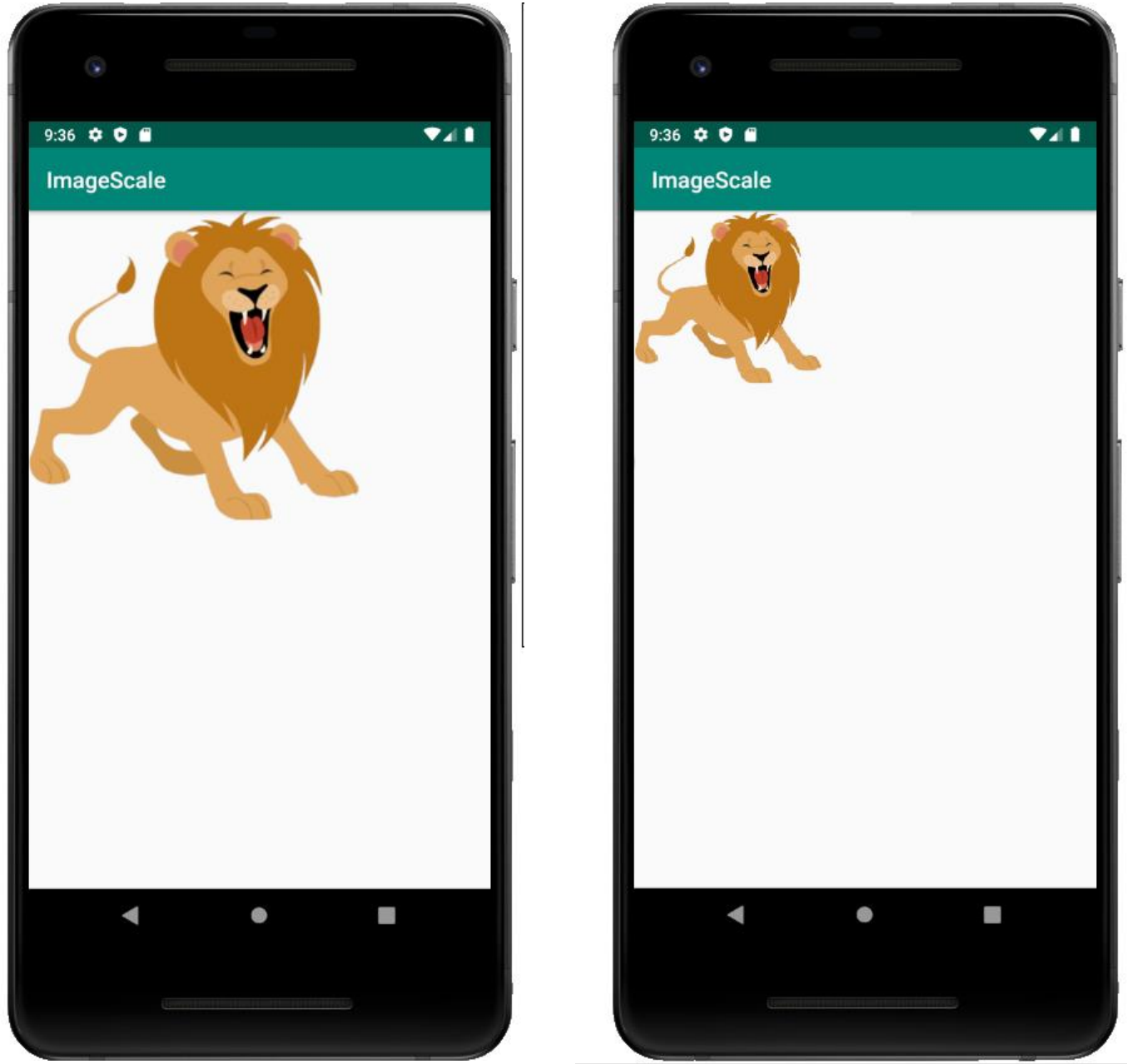

```
public class MultiTouchView extends View {

    private static final int SIZE = 60;
    final int MAX_POINTS = 10;
    float[] x = new float[MAX_POINTS];
    float[] y = new float[MAX_POINTS];
    boolean[] touching = new boolean[MAX_POINTS];

    private Paint mPaint;

    public MultiTouchView(Context context, AttributeSet attrs) {
        super(context, attrs);
        mPaint = new Paint(Paint.ANTI_ALIAS_FLAG);
        mPaint.setColor(Color.BLUE);
        mPaint.setStyle(Paint.Style.FILL_AND_STROKE);
    }
```

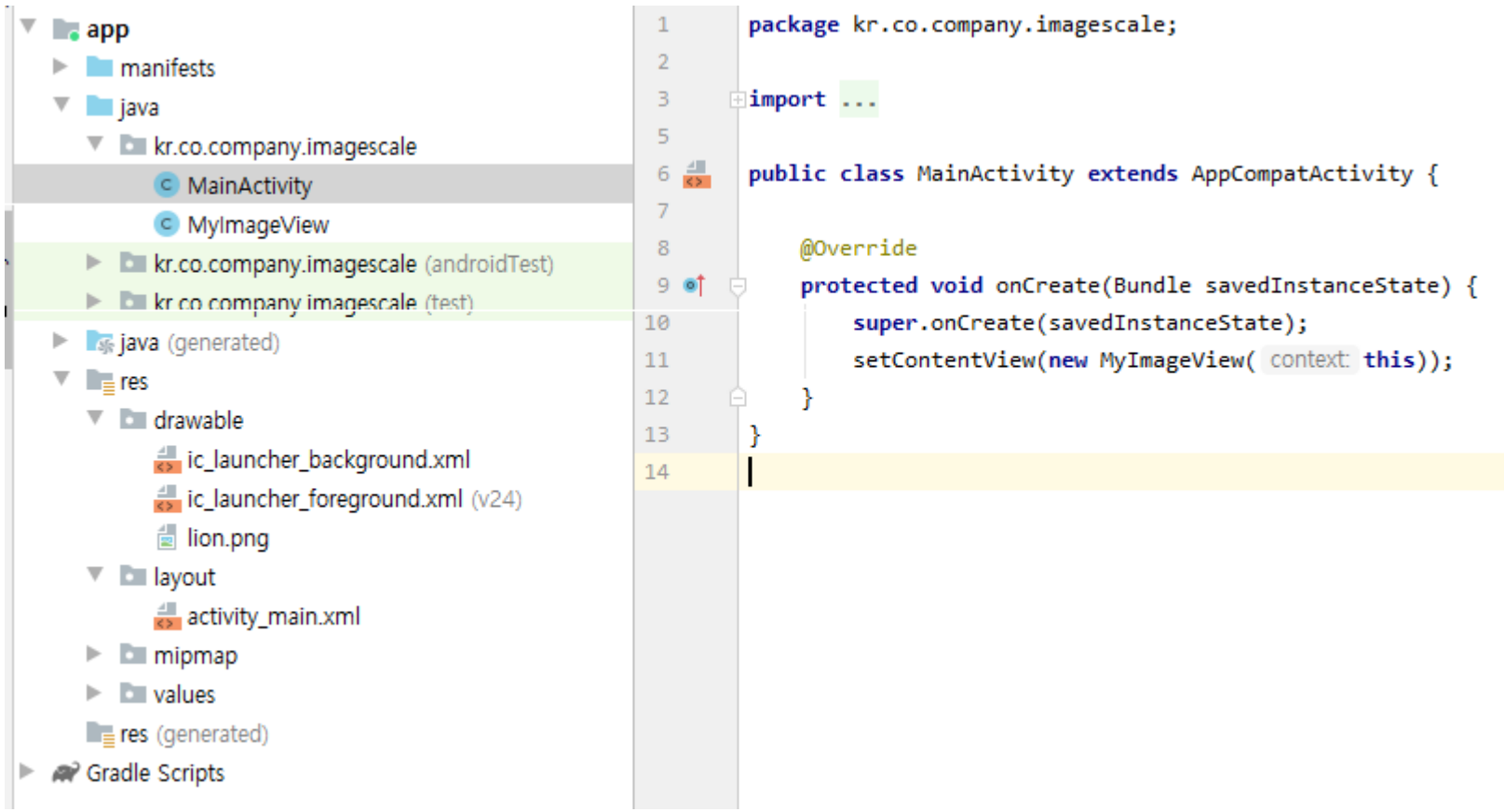
```
@Override
    public boolean onTouchEvent(MotionEvent event) {

        int index = event.getActionIndex();
        int id = event.getPointerId(index);
        int action = event.getActionMasked();
        switch (action) {
            case MotionEvent.ACTION_DOWN:
            case MotionEvent.ACTION_POINTER_DOWN:
                x[id] = (int) event.getX(index);
                y[id] = (int) event.getY(index);
                touching[id] = true;
                break;
            case MotionEvent.ACTION_MOVE:
                break;
            case MotionEvent.ACTION_UP:
            case MotionEvent.ACTION_POINTER_UP:
            case MotionEvent.ACTION_CANCEL:
                    touching[id] = false;
                break;
        }
        invalidate();
        return true;
    }
```

MultiTouchView.java


```
    @Override
    protected void onDraw(Canvas canvas) {
        super.onDraw(canvas);

        for (int i = 0; i < MAX_POINTS; i++) {
            if (touching[i]) {
                canvas.drawCircle(x[i], y[i], SIZE, mPaint);
            }
        }
    }
}
```

SWITCH

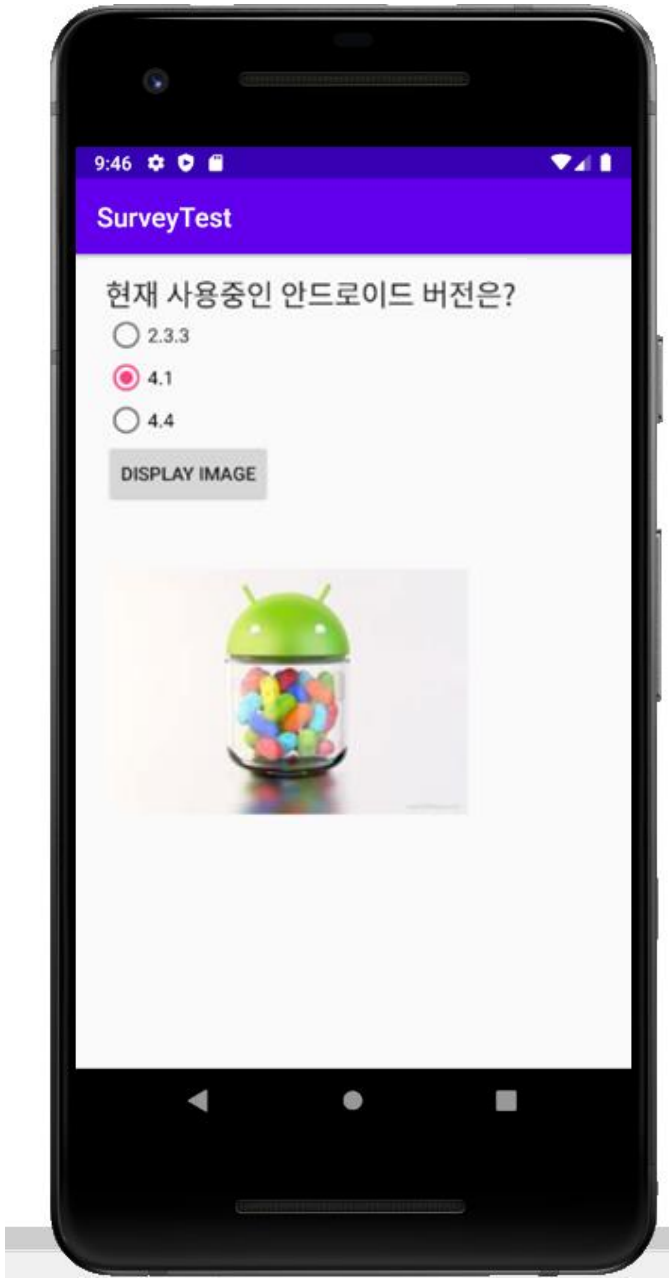

9:36
ImageScale
9:36
ImageScale

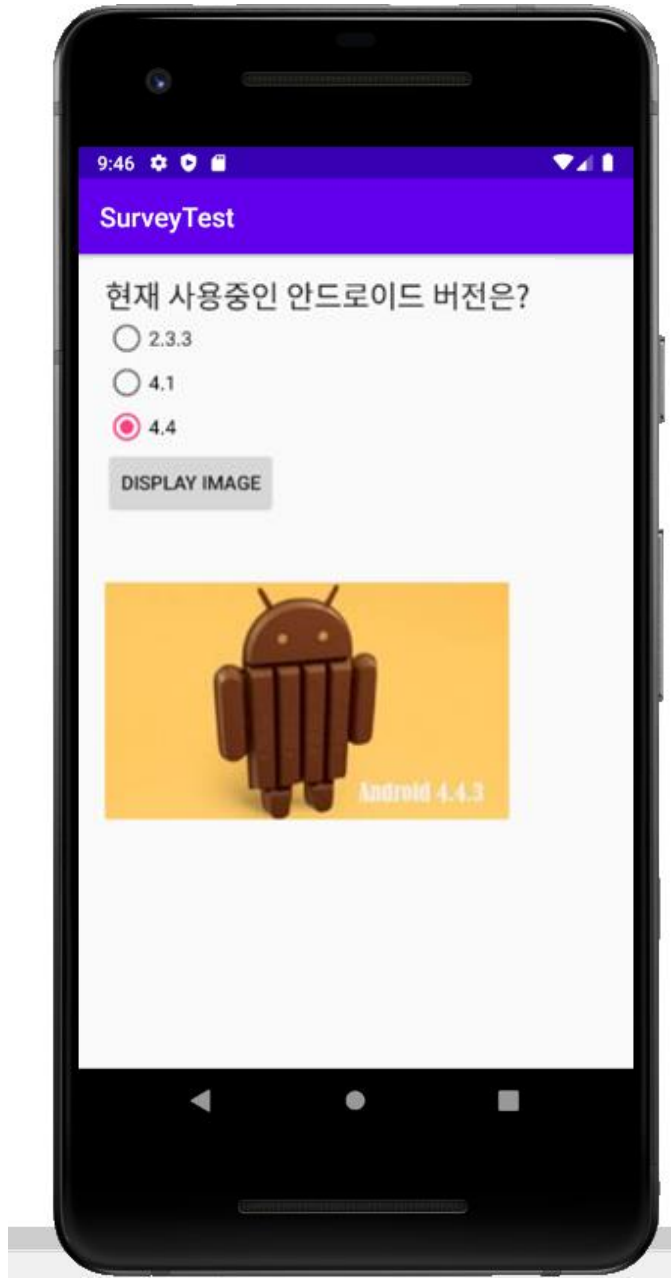

app
manifests
java
kr.co.company.imagescale
MainActivity
MyImageView
kr.co.company.imagescale (androidTest)
kr.co.company.imagescale (test)
java (generated)
res
drawable
ic_launcher_background.xml
ic_launcher_foreground.xml (v24)
lion.png
layout
activity_main.xml
mipmap
values
res (generated)
Gradle Scripts
1 package kr.co.company.imagescale;
2
3 import ...
5
6 public class MainActivity extends AppCompatActivity {
7
8 @Override
9 protected void onCreate(Bundle savedInstanceState) {
10 super.onCreate(savedInstanceState);
11 setContentView(new MyImageView(context: this));
12 }
13 }
14

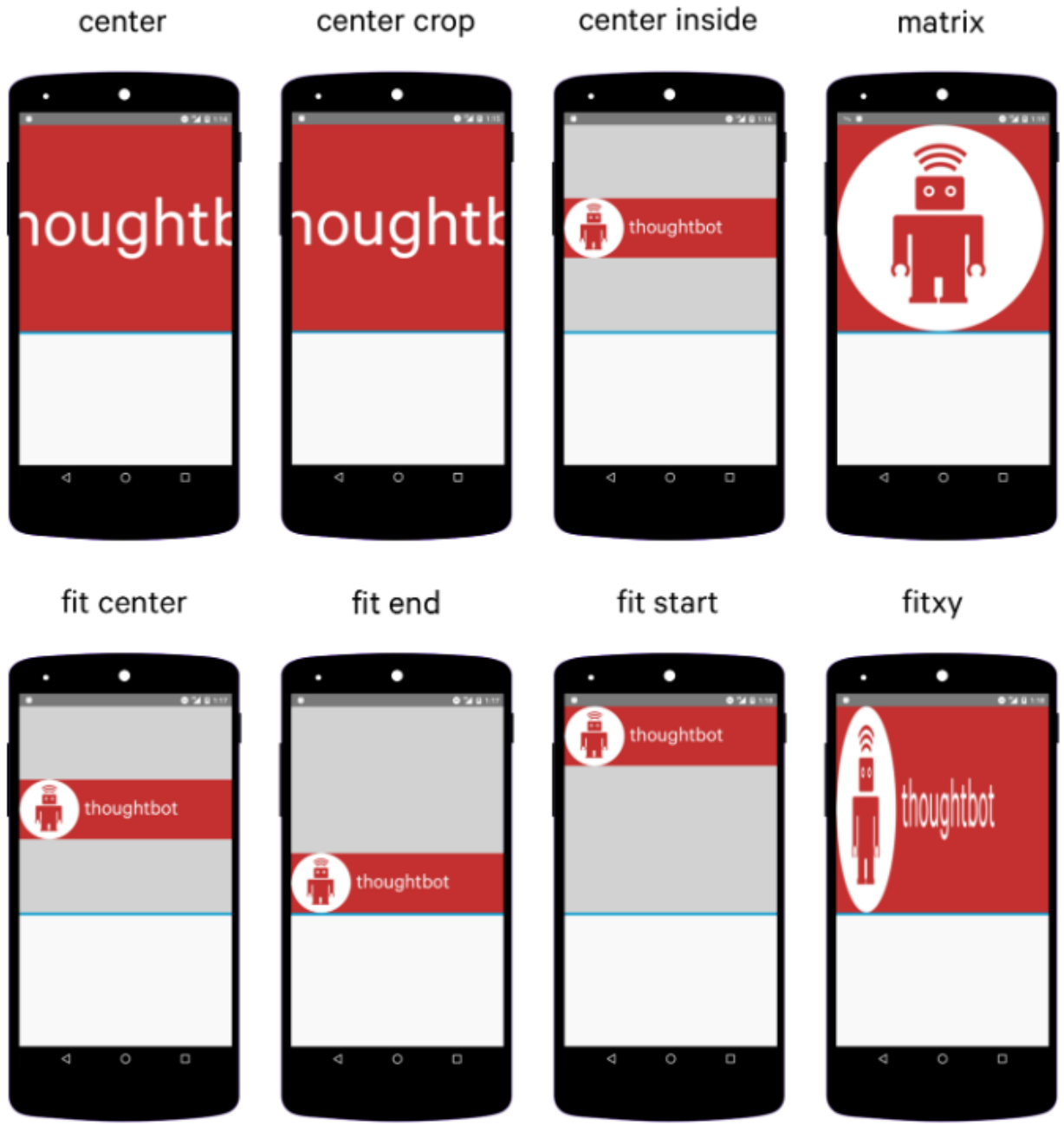
```
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new MyImageView(this));
    }
}
```

```
public class MyImageView extends View {
    private Drawable image;
    private ScaleGestureDetector gestureDetector;
    private float scale = 1.0f;

    public MyImageView(Context context) {
        super(context);
        image = context.getResources().getDrawable(R.drawable.lion);
        setFocusable(true);
        image.setBounds(0, 0, image.getIntrinsicWidth(),
                image.getIntrinsicHeight());
        gestureDetector = new ScaleGestureDetector(context, new ScaleListener());
    }

    @Override
    protected void onDraw(Canvas canvas) {
        super.onDraw(canvas);
        canvas.save();
        canvas.scale(scale, scale);
        image.draw(canvas);
        canvas.restore();
    }
```

```
    @Override
        public boolean onTouchEvent(MotionEvent event) {
            gestureDetector.onTouchEvent(event);
            invalidate();
            return true;
        }

        private class ScaleListener extends
                ScaleGestureDetector.SimpleOnScaleGestureListener {
            @Override
            public boolean onScale(ScaleGestureDetector detector) {
                scale *= detector.getScaleFactor();

                if (scale < 0.1f)
                    scale = 0.1f;
                if (scale > 10.0f)
                    scale = 10.0f;

                invalidate();
                return true;
            }
        }
    }
```


9:46
SurveyTest
현재 사용중인 안드로이드 버전은?
2.3.3
4.1
4.4
DISPLAY IMAGE


9:46
SurveyTest
현재 사용중인 안드로이드 버전은?
2.3.3
4.1
4.4
DISPLAY IMAGE
Android 4.4.3



https://recipes4dev.tistory.com/105