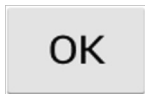
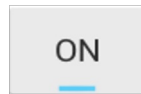


# Button Controls

Button



ToggleButton



Switch



CheckBox



CheckedTextView



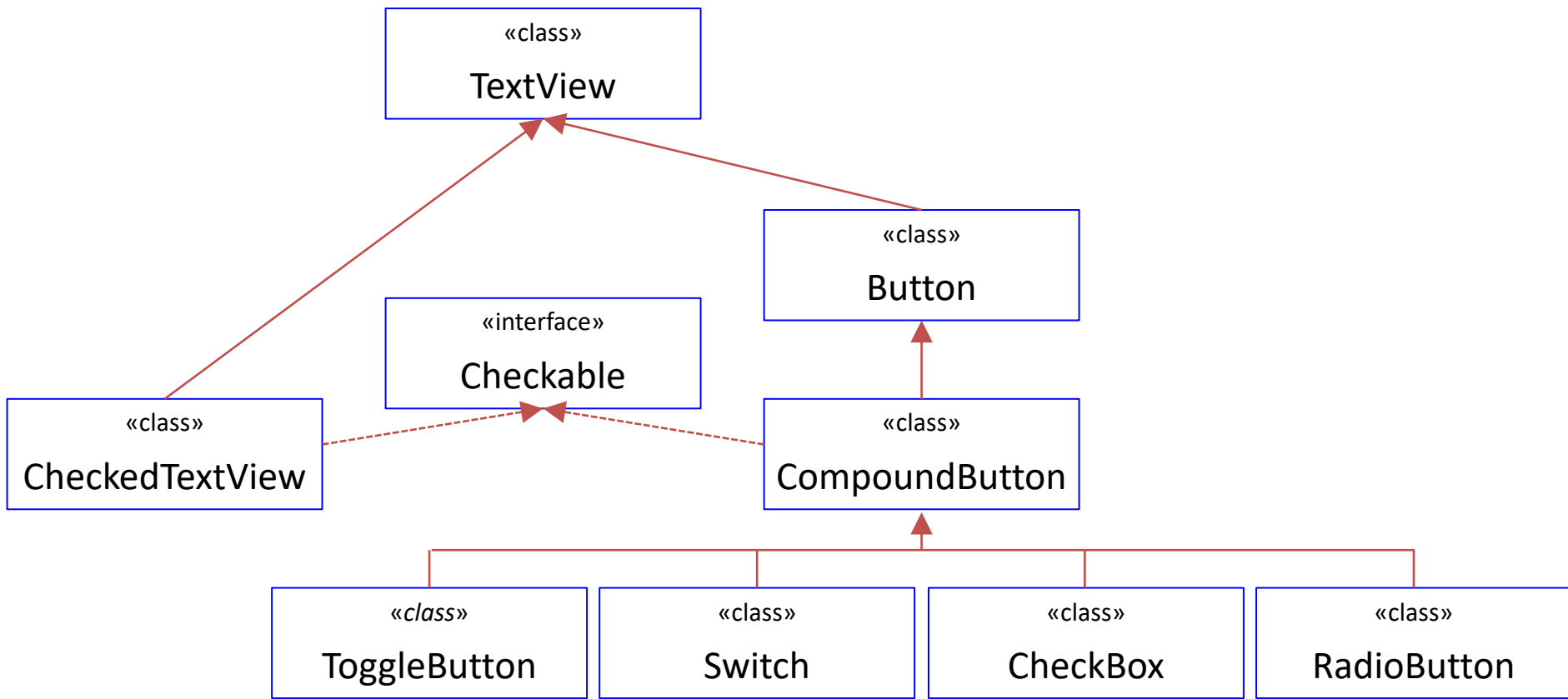
RadioButton



ImageButton



# Button Controls



# Button



## Definición en Layout

```
<Button android:id="@+id/bt_ok"  
    android:text="@android:string/ok"  
    android:onClick="nombreMetodo"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"/>
```

**text** representa el texto del botón. Puede ser un literal o la referencia a un recurso string (preferible)

**onclick** representa el nombre del método del Activity que se ejecutará cuando un usuario pulse el botón. La firma de ese método debe ser:  
`public void nombreMetodo(View view)`  
donde el parámetro `view` representa la vista pulsada (en este caso al botón)

## Código

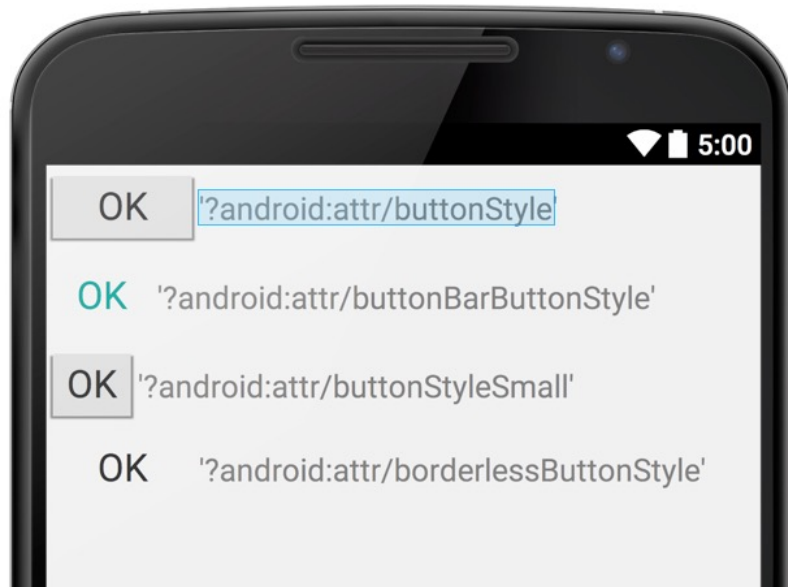
```
import android.widget.Button;  
...  
val btOK : Button = findViewById<Button>(R.id.bo_ok)
```

# Button

OK

## Definición en Layout — atributos visuales

```
<Button android:id="@+id/bt_ok"  
        android:text="@android:string/ok"  
        style="?android:attr/buttonStyle"  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"/>
```



## Métodos

# Button



`setText(text: CharSequence): Unit`

`setText(stringID: Int)`

`getText() : CharSequence`

Establece y recupera el texto que mostrará la vista. El texto puede indicarse a través de una cadena de caracteres o mediante un entero que represente la identidad de un recurso de tipo String (`R.String.xxx`)

`setOnClickListener (listener : View.OnClickListener) : Unit`

Fija un listener que será ejecutado cuando el usuario pulse el botón.

```
val btOK : Button = findViewById<Button>(R.id.bo_ok)
btOk.setOnClickListener { v -> : : }
```

# ImageButton



## Definición en Layout

```
<ImageButton android:id="@+id/ib"  
    android:src="@android:drawable/sym_def_app_icon"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"/>
```

**src**

representa la imagen que se mostrará en el botón. Debe ser una referencia a un recurso drawable

## Código

```
import android.widget.ImageButton;  
...  
val ib : ImageButton = findViewById<ImageButton>(R.id.ib)
```

## Métodos

# ImageButton



`setImageBitmap(bitmap: Bitmap)`

`setImageResource(resourceId: Int)`

`setImageDrawable(drawable: Drawable)`

Establece la imagen que mostrará el botón. Puede indicarse a través del identificador de la imagen (`R.drawable.xxx`) o por medio del propio recurso ( `Context.resources.getDrawable( R.drawable.xxx )` )

`setOnClickListener (listener : View.OnClickListener) : Unit`

Fija un listener que será ejecutado cuando el usuario pulse el botón.

```
val ib : ImageButton = findViewById<ImageButton>(R.id.ib)
```

```
ib.setOnClickListener { v -> :: }
```