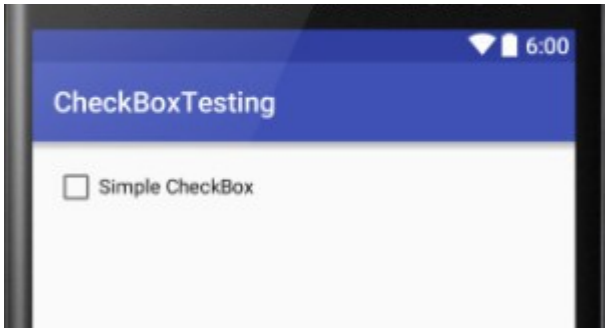
# Android Checkbox

In this lesson, we will learn about how to use android checkbox with example. We will learn about different attributes that are used to customise this widget.




## Getting Started

Android Checkbox can be described as below –

> Android Checkbox is special type of button that has only 2 states i.e. either **checked** or **unchecked**.

## Different Attributes of Android Checkbox Widget

Attributes of checkbox widget are inherited from Textview, View and Compound Button. Some of the popular attributes inherited from Textview are –

| Sr. | XML Attributes | Description |
|---|---|---|
| 1 | android:backgroundTint | Sets tint to the background. |
| 2 | android:clickable | Sets true when you want to make View clickable. Otherwise, set false. |
| 3 | android:drawableBottom | Drawable to be drawn at bottom of the text. |
| 4 | android:drawableEnd | Drawable to be drawn to end of the text. |
| 5 | android:drawableLeft | Drawable to be drawn to left of the text. |
| 6 | android:drawablePadding | Padding of the drawable. |

Attributes of Checkbox inherited from Compound Button are –

| Sr. | XML Attributes | Description |
|---|---|---|
| 1 | android:button | Drawable to be used for button graphic |
| 2 | android:buttonTint | Sets tint to button graphic |

| | | |
|---|---|---|
| 3 | android:buttonTintMode | Blending mode used to apply the button graphic tint. |

Attributes of Checkbox inherited from View are –

| Sr. | XML Attributes | Description |
|---|---|---|
| 1 | android:id | It sets unique identifier for this view. |
| 2 | android:padding | Sets padding of this view. |
| 3 | android:onClick | Defines the operations to perform when this view is clicked |
| 4 | android:visibility | Sets the visibility (visible, gone etc.) of the Checkbox. |
| 5 | android:tooltipText | Defines text displayed in a small popup window on hover or long press. |
| 6 | android:background | Sets background to this view. |
| 7 | android:alpha | Sets alpha in view. |

# Example of Android Checkbox Widget

At first, we will create android application. Then, we will use checkbox widget in this application.

## 1. Creating New Project

Follow steps below to create new android project. Please ignore the steps if you have already created android application.

| Step | Description |
|---|---|
| 1. | Open Android Studio. |
| 2. | Go to **File** => **New** => **New Project**. Write application name as **Checkbox**. Then, click **next** button. |
| 3. | Select minimum SDK you need. However, we have selected 17 as minimum SDK. Then, click **next** button |
| 4. | Then, select **Empty Activity** => click **next** => click **finish**. |
| 5. | If you have followed above process correctly, you will get a newly created project successfully. |

## Use Checkbox in xml file

Open **res/layout/activity_main.xml** file. Then, add below code into it.

```
<?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"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="10dp"
        android:text="Check me Please"/>

</LinearLayout>
```

Notice that we have defined Checkbox widget with id **checkBox** in xml file. Now, we will access this checkbox widget in java file. Then, We will show a toast message when checkbox is clicked.

## Access Checkbox in java file

Open **src/main/java/com.yourpackage/MainActivity.java** file. Then, add below code into it.

```
package com.yourpackage
import android.os.Bundle
import android.widget.CheckBox
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
class Calc : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        val checkBox = findViewById<CheckBox>(R.id.checkBox)
        checkBox?.setOnCheckedChangeListener { buttonView, isChecked ->
          val msg = "You have " + (if (isChecked) "checked" else
         "unchecked") + " this Check it Checkbox."
            Toast.makeText(this, msg, Toast.LENGTH_SHORT).show()
        }
    }}
```

### Run You App