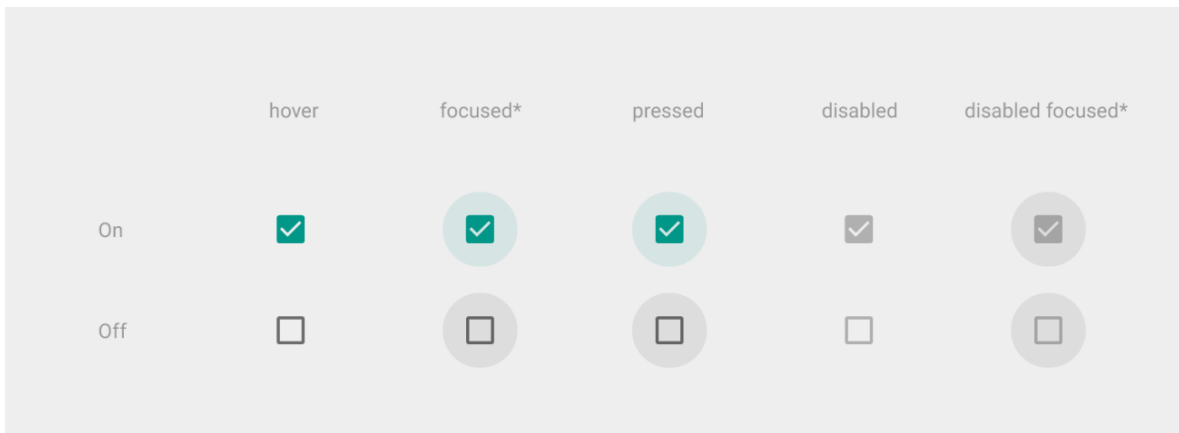


CHECKBOX

Una checkbox es un control de selección, al igual que los radio buttons. Sirven para seleccionar una o más opciones de un conjunto. El retorno es de tipo boolean, donde true significa checked y false unchecked.

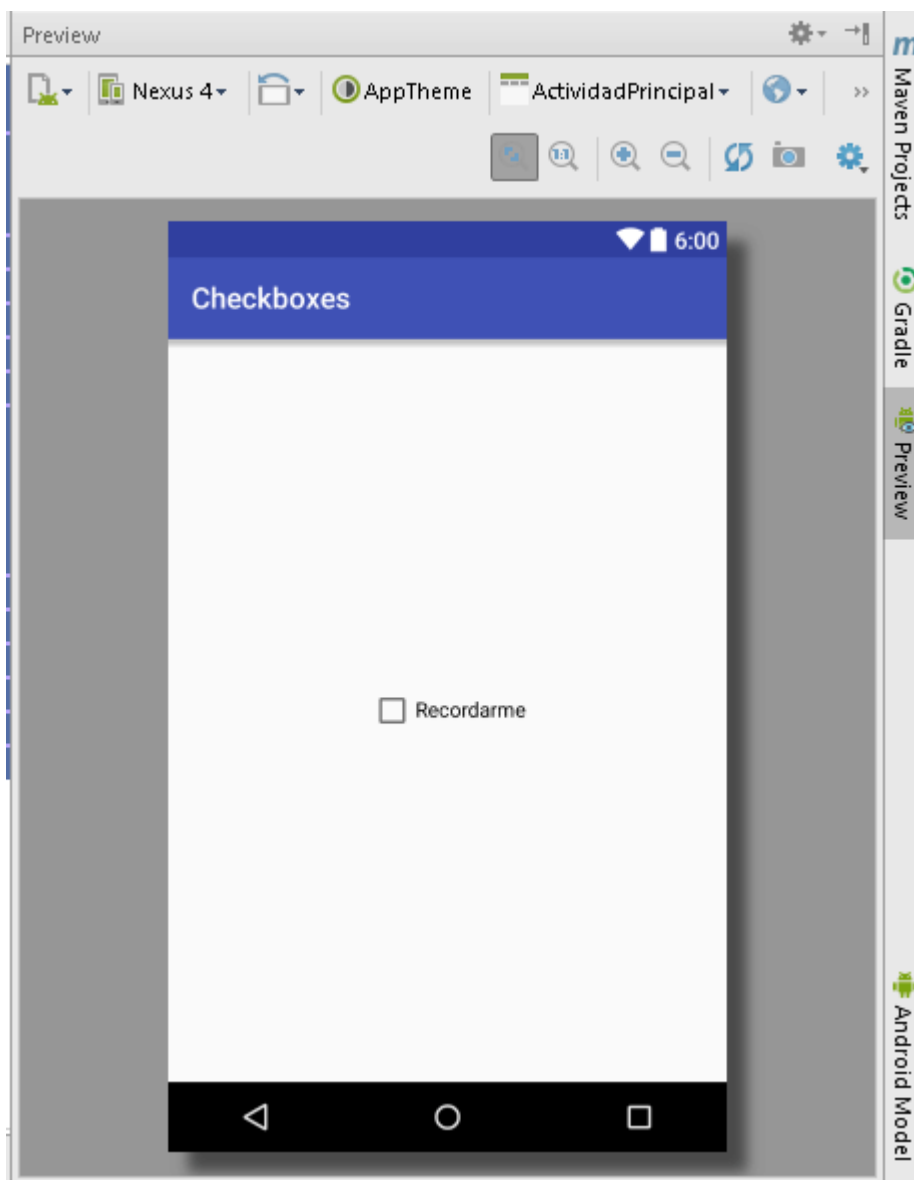
- hover: Estado cuando el Checkbox se encuentra inmóvil
- focused: Cuando la navegación en la UI apunta al checkbox como control de entrada
- pressed: El checkbox se encuentra bajo selección prolongada
- disable: El checkbox pierde su capacidad de cambio entre estados
- disable-focused: El checkbox está deshabilitado pero el sistema de navegación lo enfoca



```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.herprogramacion.checkboxes.ActividadPrincipal">

    <CheckBox
        android:id="@+id/checkBox"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true"
        android:text="Recordarme" />

</RelativeLayout>
```



PRACTICA:

Campo de contraseña con checkbox que revele la contraseña al ser presionado

Archivo xml

```
1. <?xml version="1.0" encoding="utf-8"?>
2. <RelativeLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
3.     android:layout_width="match_parent"
4.     android:layout_height="match_parent"
5.     android:orientation="vertical">
6.
7.     <CheckBox
8.         android:id="@+id/opcion_mostrar"
9.         android:layout_width="wrap_content"
10.        android:layout_height="wrap_content"
11.        android:layout_alignLeft="@+id/campo_contrasena"
12.        android:layout_alignStart="@+id/campo_contrasena"
13.        android:layout_below="@+id/campo_contrasena"
14.        android:onClick="mostrarContraseña"
15.        android:text="¿Mostrar contraseña?" />
16.
17.     <EditText
18.         android:id="@+id/campo_contrasena"
19.         android:layout_width="wrap_content"
20.         android:layout_height="wrap_content"
21.         android:layout_centerHorizontal="true"
22.         android:layout_centerVertical="true"
23.         android:ems="10"
24.         android:hint="Contraseña"
25.         android:inputType="textPassword" />
26. </RelativeLayout>
```

Abre `ActividadPrincipa.java` y obtén las instancias de los views en **onCreate()**. Luego definir el controlador **mostrarContrasena()**

```
1. import android.os.Bundle;
2. import android.support.v7.app.AppCompatActivity;
3. import android.text.InputType;
4. import android.view.View;
5. import android.widget.CheckBox;
6. import android.widget.EditText;
7. public class ActividadPrincipal extends AppCompatActivity {
8.     private CheckBox opcionMostrar;
```

```
9.     private EditText campoContrasena;
10.
11.     @Override
12.     protected void onCreate(Bundle savedInstanceState) {
13.         super.onCreate(savedInstanceState);
14.         setContentView(R.layout.actividad_principal);
15.
16.         opcionMostrar = (CheckBox)findViewById(R.id.opcion_mostrar);
17.         campoContrasena = (EditText)findViewById(R.id.campo_contrasena);
18.     }
19.
20.     public void mostrarContraseña(View v){
21.         // Salvar cursor
22.         int cursor = campoContrasena.getSelectionEnd();
23.
24.         if(opcionMostrar.isChecked()){
25.             campoContrasena.setInputType(InputType.TYPE_CLASS_TEXT
26.                 | InputType.TYPE_TEXT_VARIATION_VISIBLE_PASSWORD);
27.         }else{
28.             campoContrasena.setInputType(InputType.TYPE_CLASS_TEXT
29.                 | InputType.TYPE_TEXT_VARIATION_PASSWORD);
30.         }
31.
32.         // Restaurar cursor
33.         campoContrasena.setSelection(cursor);
34.     }
35.
36. }
```

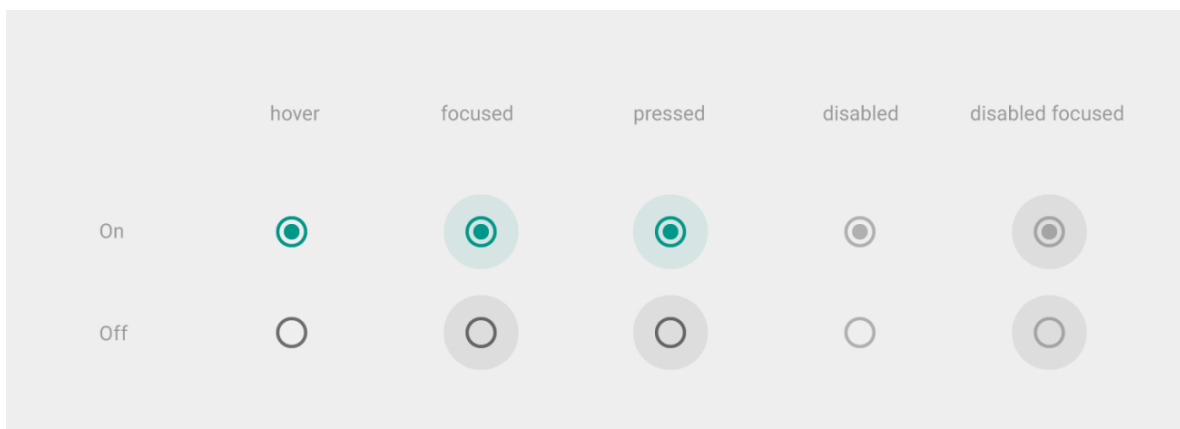


RADIO BUTTON

Un RadioButton es un control de selección que proporciona la interfaz de Android para permitir a los usuarios seleccionar una opción de un conjunto.

Son ideales para elegir uno de varios elementos con exclusión mutua, es decir, la selección de un radio button obliga a descartar la de otro, permitiendo solo a un ítem estar activo.

- hover: el radio button está a la espera de eventos
- focused: el sistema de navegación de android enfocó al radio button
- pressed: el usuario mantiene presionado el radio button
- disabled: el radio button no está habilitado para interacción
- disabled-focused: combinación de disabled + focused.



PRACTICA

Mostrar los campos Razón social, Representante y Número de empleados si se selecciona **Ciente corporativo**. De lo contrario, mostrar los campos para nombre completo y profesión para la opción **Particular**.

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
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">

<LinearLayout
    android:layout_width="368dp"
    android:layout_height="495dp"
    android:orientation="vertical"
    tools:layout_editor_absoluteX="8dp"
    tools:layout_editor_absoluteY="8dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="35dp"
        android:orientation="vertical">

        <TextView
            android:id="@+id/textView2"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textSize="20sp"
            android:text="Tipo de cliente" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="84dp"
        android:orientation="vertical">

        <RadioGroup
            android:layout_width="wrap_content"

            android:layout_height="wrap_content"
            android:checkedButton="@id/rb_part"
```

```
>  
<RadioButton  
    android:id="@+id/rb_part"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="Particular"  
    android:onClick="onRadioButtonClicked"/>  
  
<RadioButton  
    android:id="@+id/rb_corp"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="Corporativo"  
    android:onClick="onRadioButtonClicked"/>  
</RadioGroup>  
</LinearLayout>
```

```
<LinearLayout  
    android:id="@+id/contenedor_particular"  
    android:layout_width="match_parent"  
    android:layout_height="138dp"  
    android:orientation="vertical">
```

```
<EditText  
    android:id="@+id/et_nombre"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:ems="10"  
    android:inputType="textPersonName"  
    android:hint="Nombre Completo" />
```

```
<EditText  
    android:id="@+id/et_profesion"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:ems="10"  
    android:inputType="textPersonName"  
    android:hint="Profesión" />  
</LinearLayout>
```

```
<LinearLayout  
    android:id="@+id/contenedor_corporativo"  
    android:layout_width="match_parent"
```

```
android:layout_height="match_parent"
android:orientation="vertical"
android:visibility="gone">
```

```
<EditText
    android:id="@+id/et_razon"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="textPersonName"
    android:hint="Razón Social" />
```

```
<EditText
    android:id="@+id/et_representante"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="textPersonName"
    android:hint="Representante" />
```

```
<EditText
    android:id="@+id/et_nempleados"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="textPersonName"
    android:hint="Numero de Empleados" />
```

```
</LinearLayout>
```

```
</LinearLayout>
```

```
</android.support.constraint.ConstraintLayout>
```

```
<RadioButton
    android:id="@+id/rb_corporativo"
    android:onClick="onRadioButtonClicked"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Corporativo" />
```

```
<RadioButton
    android:id="@+id/rb_particular"
    android:onClick="onRadioButtonClicked"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Particular" />
```

Alt + Enter

- Create 'onRadioButtonClicked(View)' in 'ActividadPrincipal' ▶
- Create onClick event handler ▶
- Inject language or reference ▶
- Override Resource in Other Configuration... ▶

Crear método


```
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.RadioButton;

public class MainActivity extends AppCompatActivity {

    private View contenedorParticular;
    private View contenedorCorporativo;

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

        contenedorParticular = findViewById(R.id.contenedor_particular);
        contenedorCorporativo = findViewById(R.id.contenedor_corporativo);
    }

    public void onRadioButtonClicked(View view) {
        boolean marcado = ((RadioButton) view).isChecked();
        switch (view.getId()) {
            case R.id.rb_corp:
                if (marcado) {
                    mostrarParticular(false);
                }
                break;
            case R.id.rb_part:
                if (marcado) {
                    mostrarParticular(true);
                }
                break;
        }
    }

    private void mostrarParticular(boolean b) {
        contenedorParticular.setVisibility(b ? View.VISIBLE : View.GONE);

        contenedorCorporativo.setVisibility(b ? View.GONE : View.VISIBLE);
    }
}
```

}

6:57

RadioButtons

Tipo de cliente:

☒ Corporativo

☐ Particular

Razón social

Representante

Número de empleados

6:57

RadioButtons

Tipo de cliente:

☐ Corporativo

☒ Particular

Nombre completo

Profesión