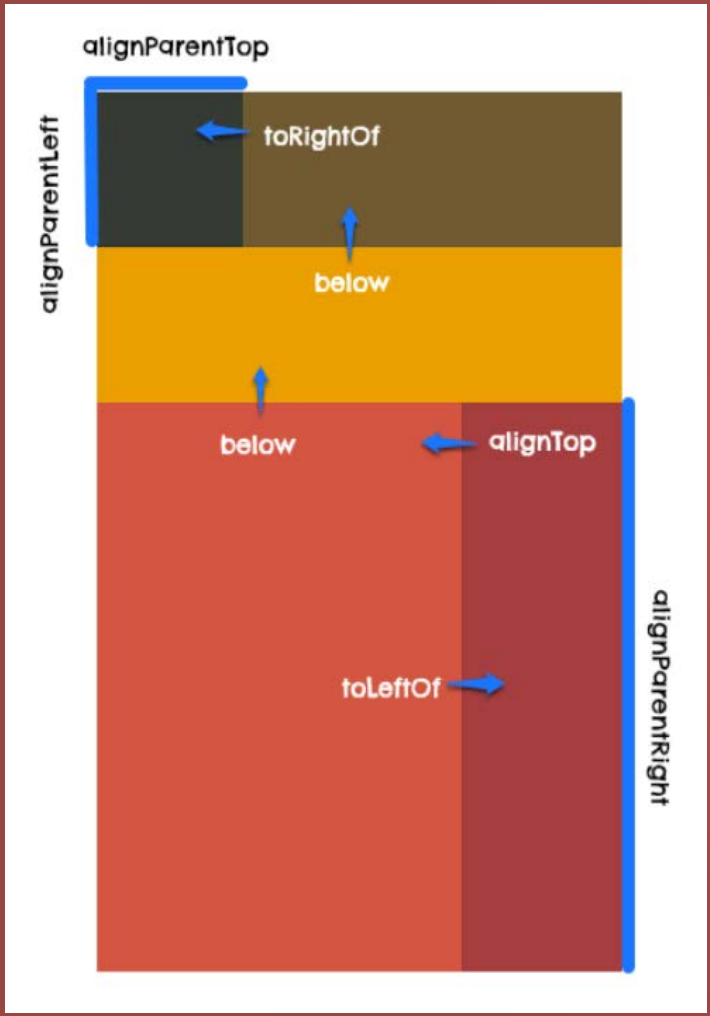
SEMINARIO DE LENGUAJES OPCIÓN ANDROID

Layouts

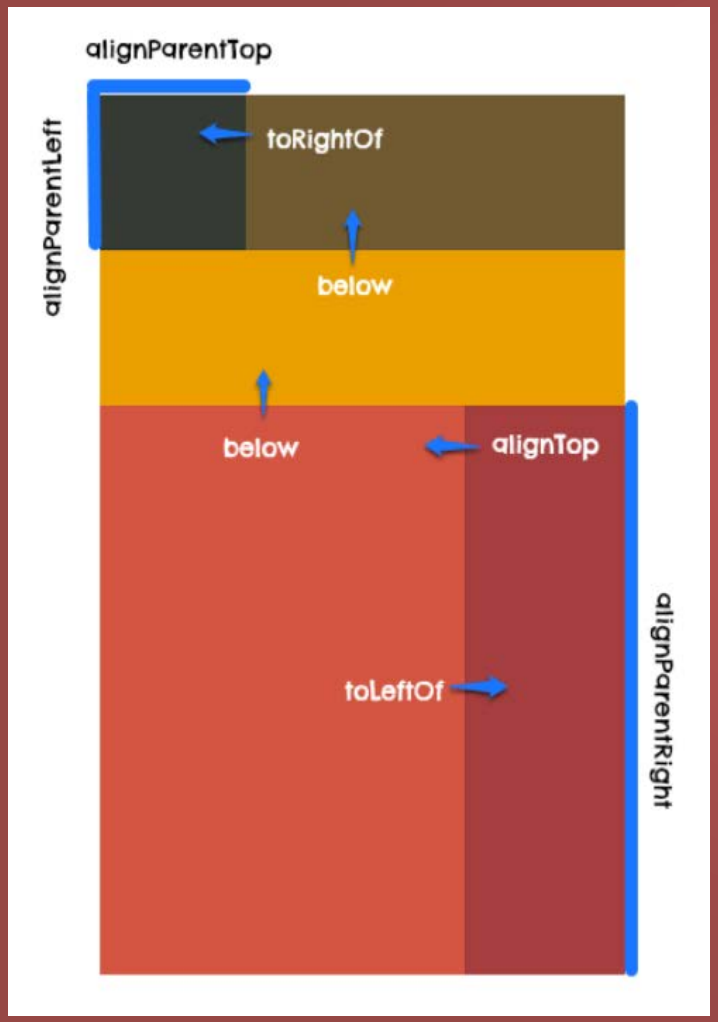
Esp. Delía Lisandro, Mg. Corbalán Leonardo

RelativeLayout



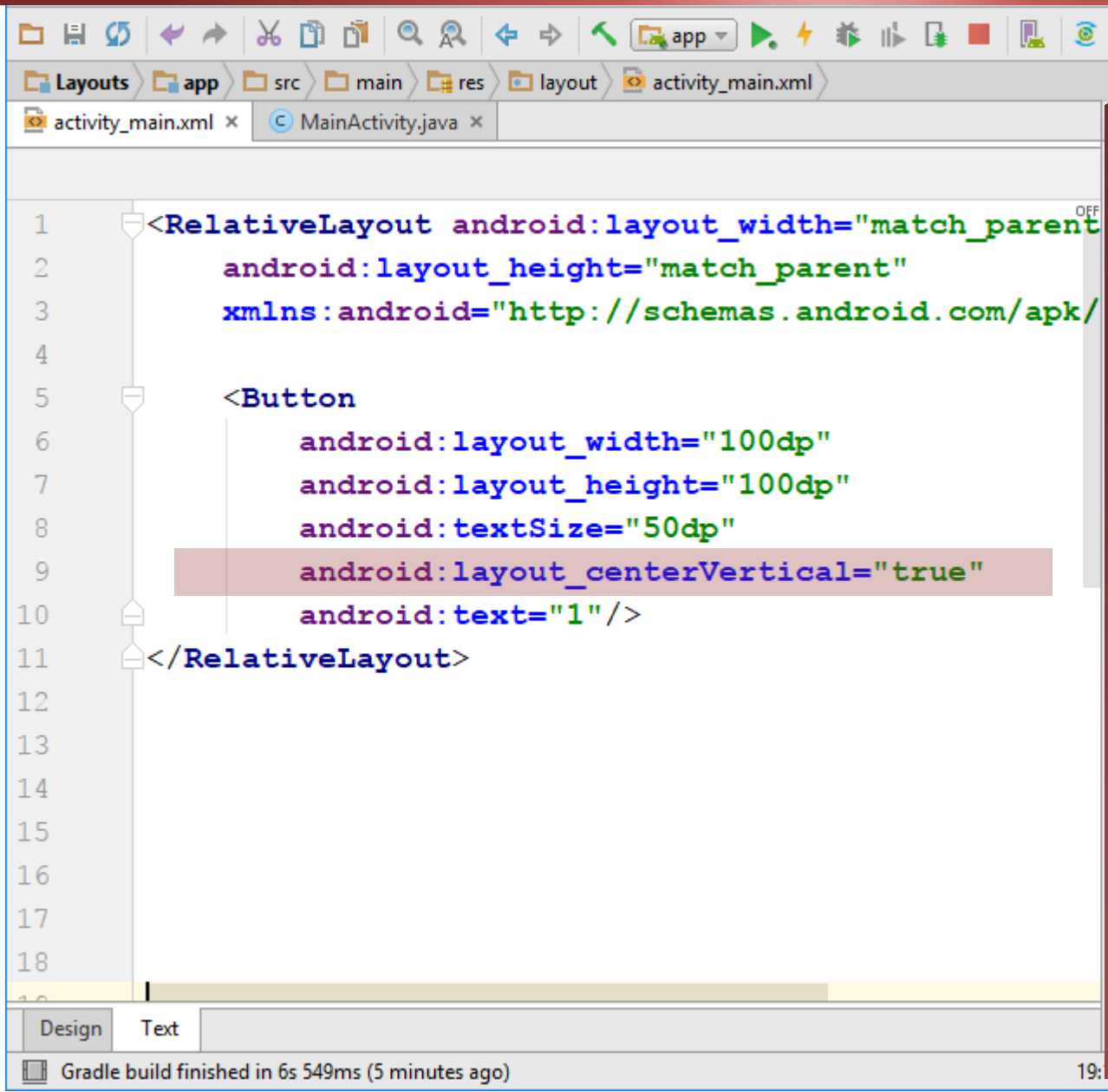
Es un Layout donde las posiciones de los elementos contenidos (hijos) pueden ser descriptas en relación con otros elementos hijos o bien en relación con el padre, es decir, el propio Layout

RelativeLayout



Elimina la necesidad de tener ViewGroups anidados, mejorando la performance

RelativeLayout

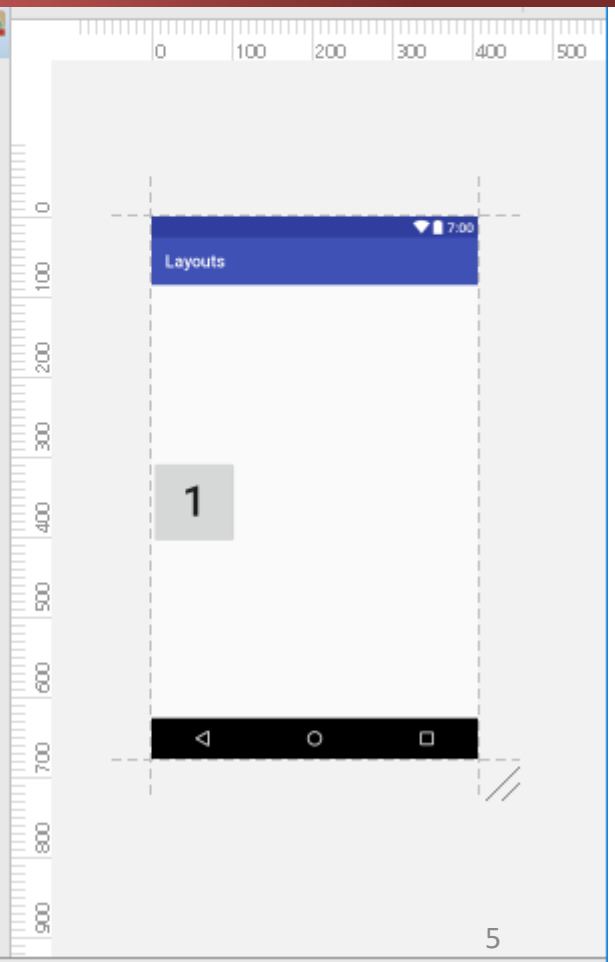


Probar el resultado de la siguiente disposición

RelativeLayout

layout_centerVertical: Centra al View verticalmente con respecto a los límites de su contenedor

```
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/

    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:layout_centerVertical="true"
        android:text="1"/>
</RelativeLayout>
```

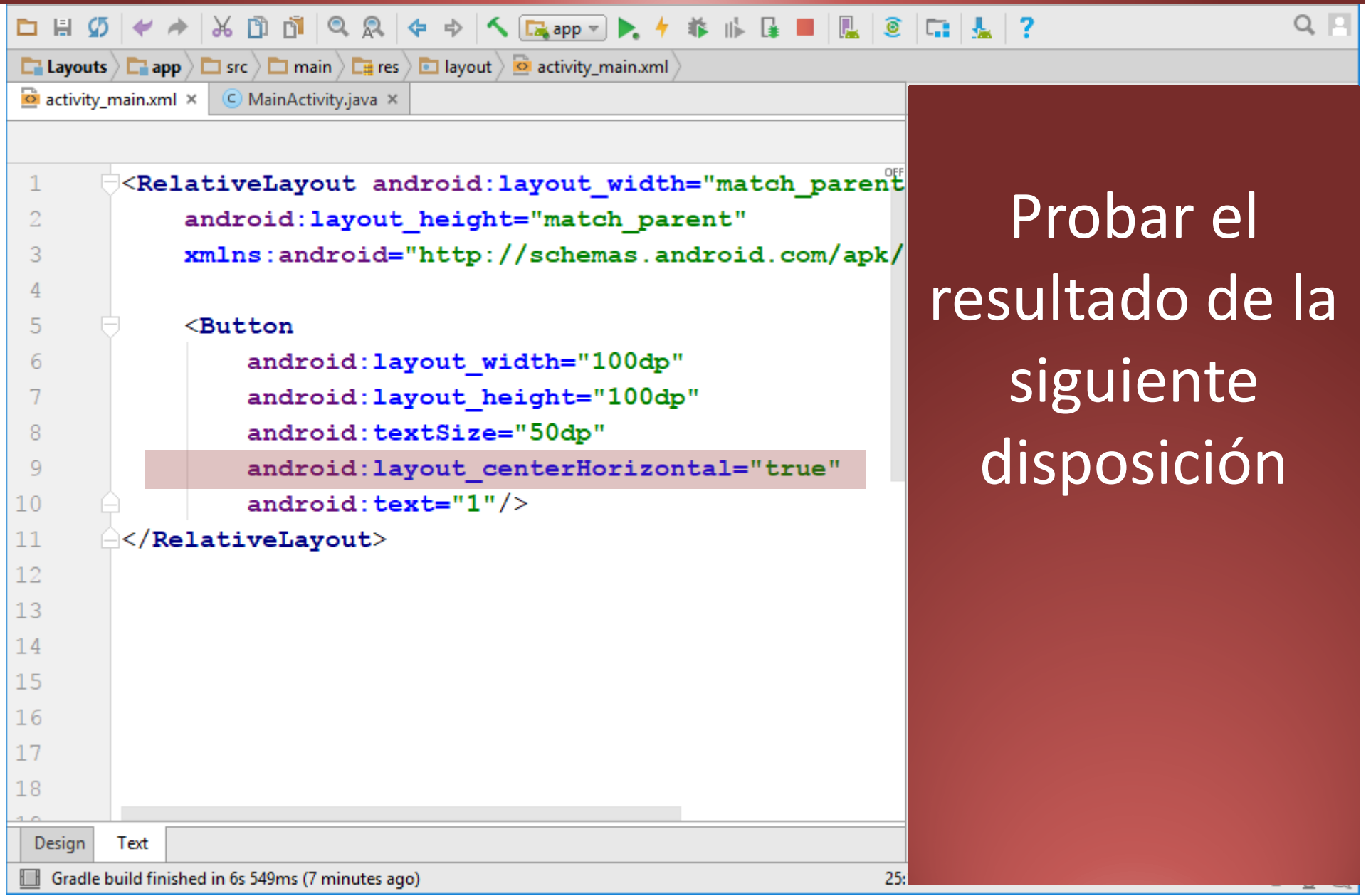


Design Text

Gradle build finished in 6s 549ms (5 minutes ago)

19:1 CRLF UTF-8 Context: <no context>

RelativeLayout

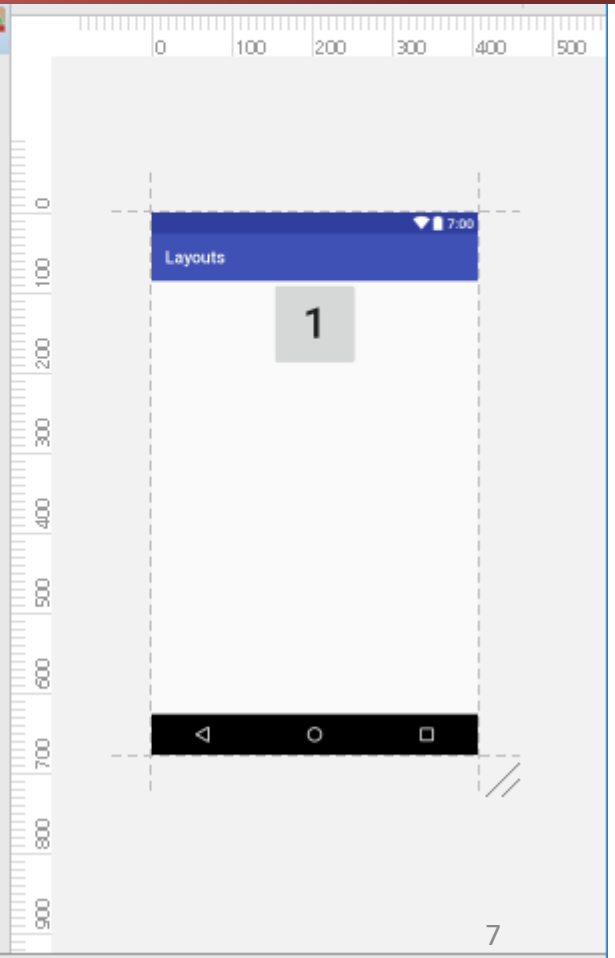


Probar el resultado de la siguiente disposición

RelativeLayout

layout_centerHorizontal: Centra al View horizontalmente con respecto a los límites de su contenedor

```
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/

    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:layout_centerHorizontal="true"
        android:text="1"/>
</RelativeLayout>
```

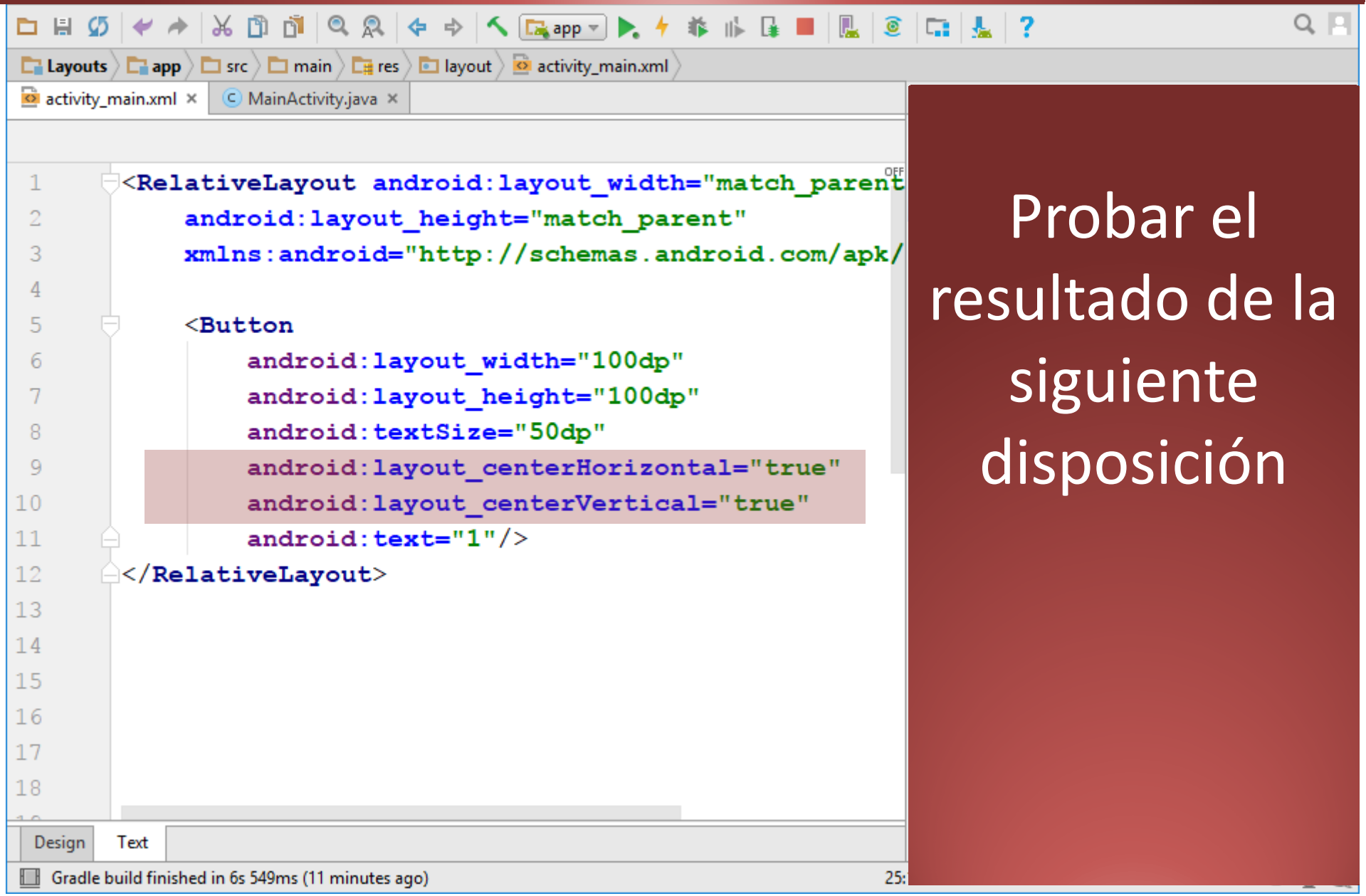


Design Text

Gradle build finished in 6s 549ms (7 minutes ago)

25:1 CRLF UTF-8 Context: <no context>

RelativeLayout



Probar el resultado de la siguiente disposición

RelativeLayout

layout_centerVertical combinado con layout_centerHorizontal central al View con respecto a los límites de su contenedor

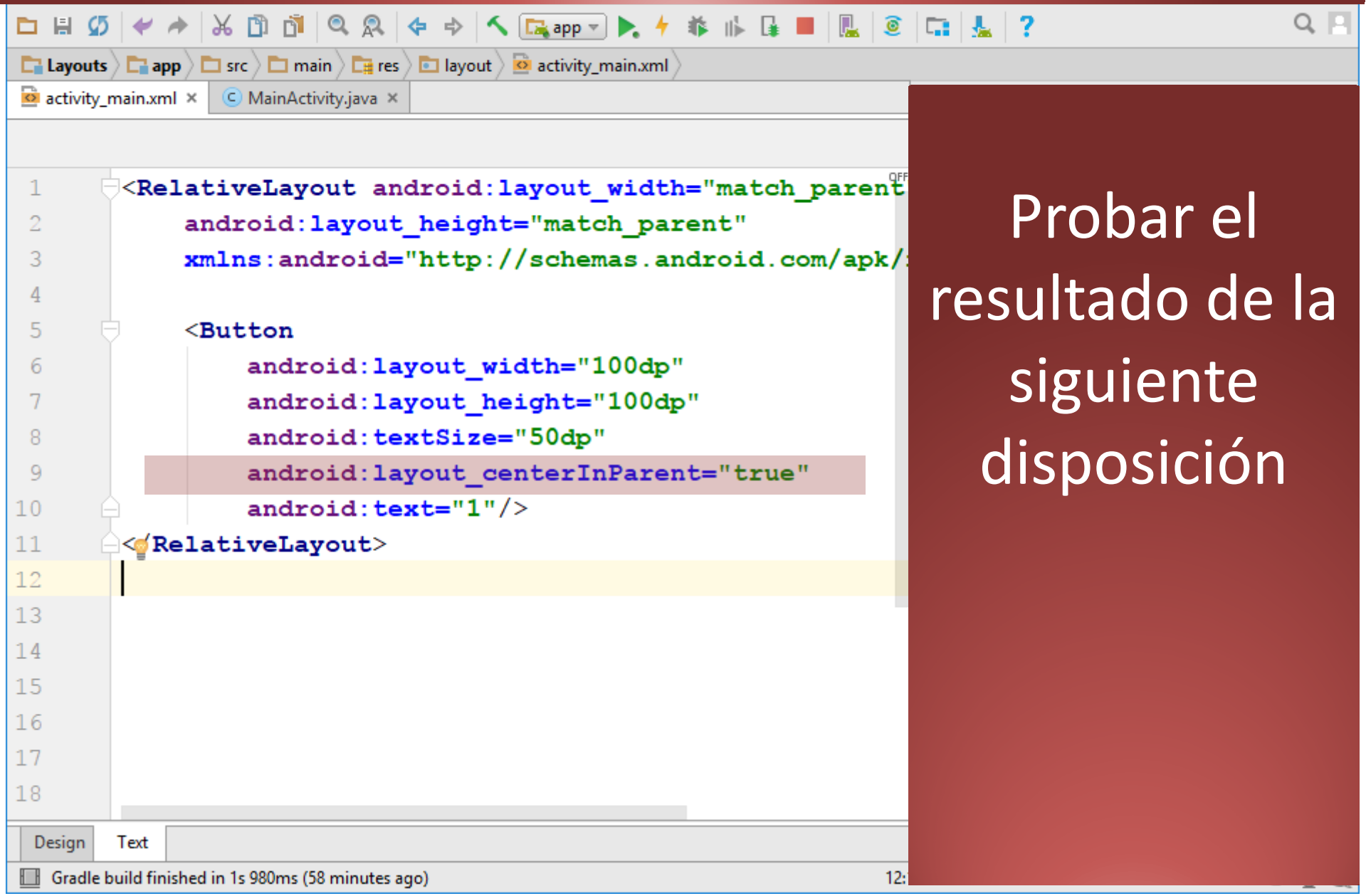
```
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/

    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true"
        android:text="1"/>
</RelativeLayout>
```

Design Text

Gradle build finished in 6s 549ms (11 minutes ago)

25:1 CRLF UTF-8 Context: <no context>

RelativeLayout



Probar el resultado de la siguiente disposición

RelativeLayout

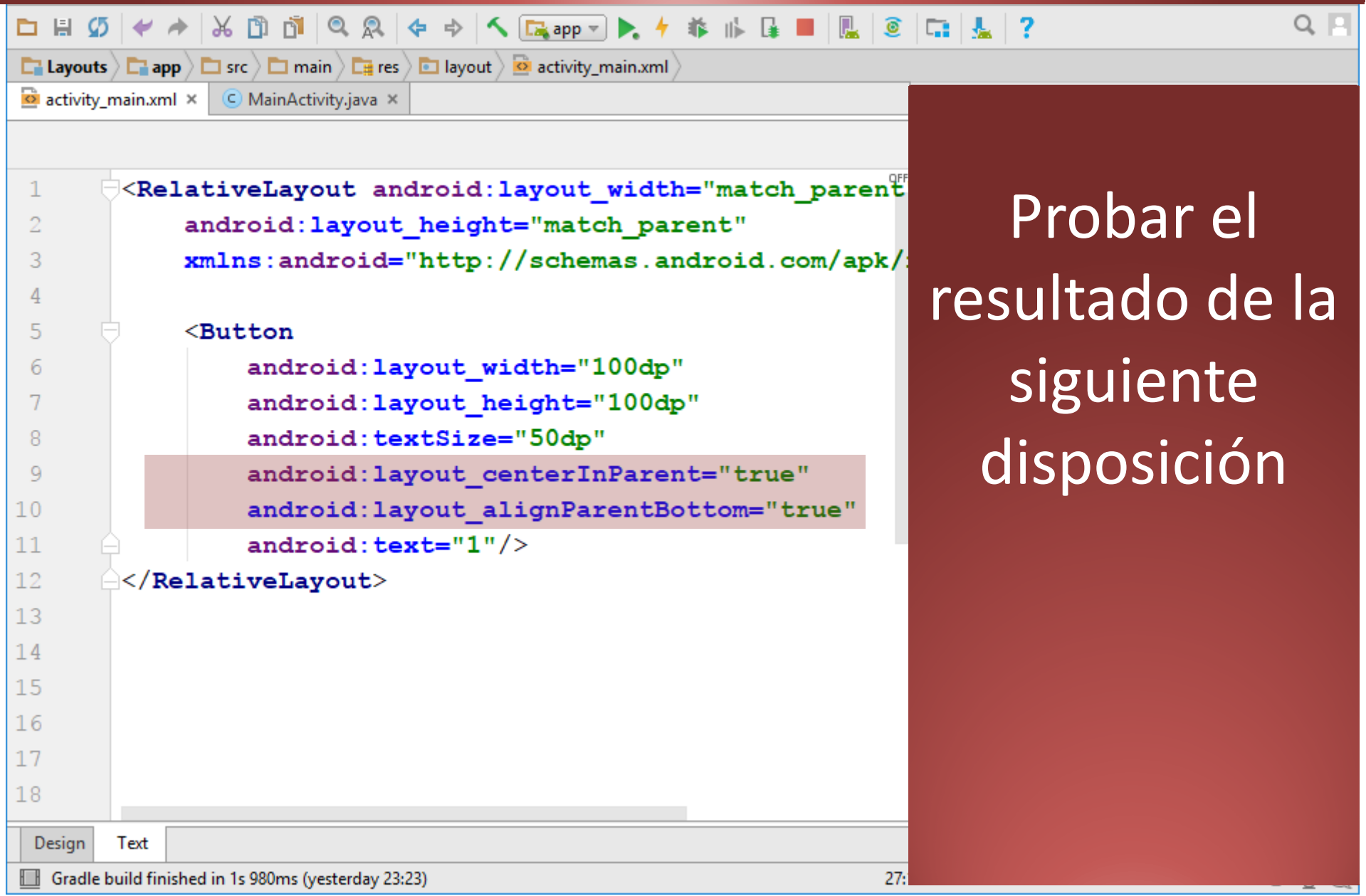
layout_centerInParent: Centra al View con respecto a los límites de su contenedor (mismo efecto que el anterior)

```
<RelativeLayout android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/
    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:layout_centerInParent="true"
        android:text="1"/>
</RelativeLayout>
```

Design Text

Gradle build finished in 1s 980ms (58 minutes ago) 12:1 CRLF UTF-8 Context: <no context>

RelativeLayout



Probar el resultado de la siguiente disposición

RelativeLayout

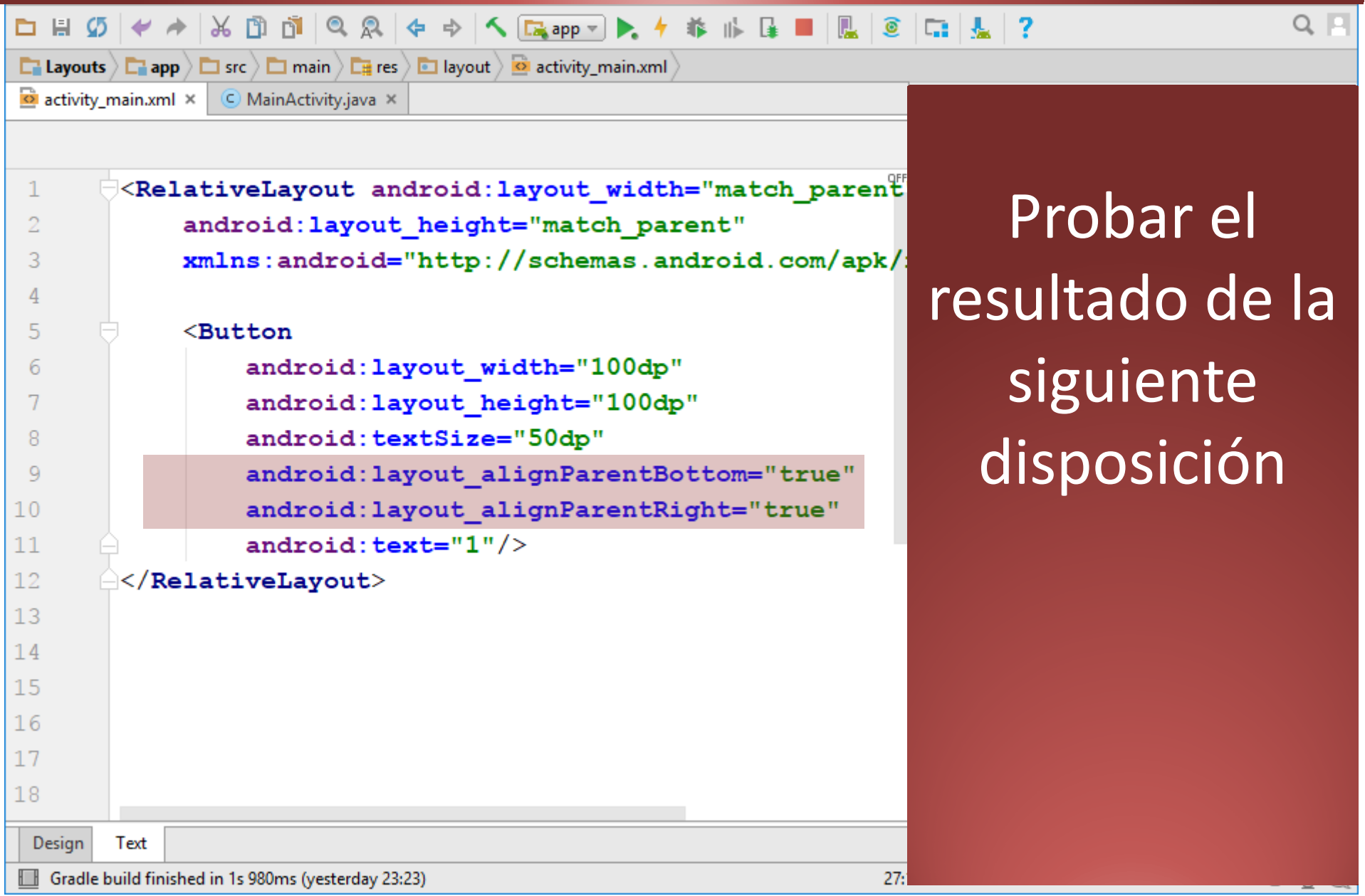
layout_alignParentBottom: Alinea el borde inferior del View con el borde inferior del contenedor

```
<RelativeLayout android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/
    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:layout_centerInParent="true"
        android:layout_alignParentBottom="true"
        android:text="1"/>
</RelativeLayout>
```

Design Text

Gradle build finished in 1s 980ms (yesterday 23:23) 27:1 CRLF UTF-8 Context: <no context>

RelativeLayout

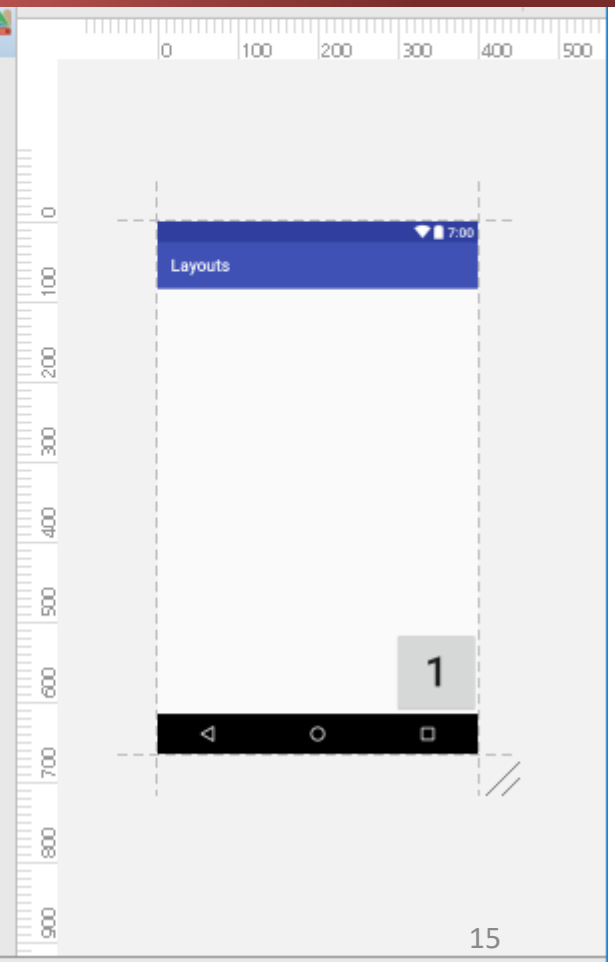


Probar el resultado de la siguiente disposición

RelativeLayout

layout_alignParentRight: Alinea el borde derecho del View con el borde derecho de su contenedor

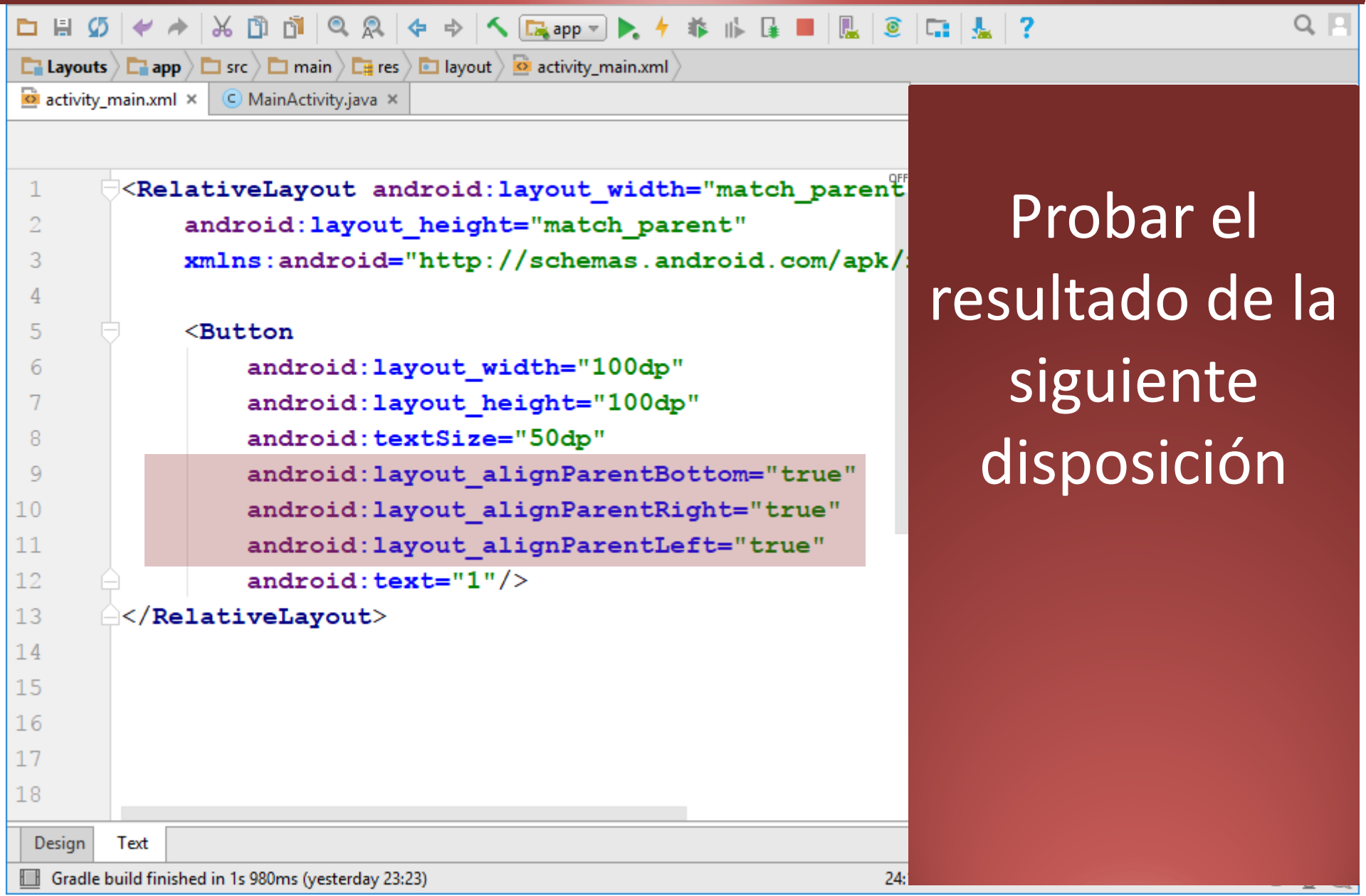
```
<RelativeLayout android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/
    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:text="1"/>
</RelativeLayout>
```



Design Text

Gradle build finished in 1s 980ms (yesterday 23:23) 27:1 CRLF UTF-8 Context: <no context>

RelativeLayout

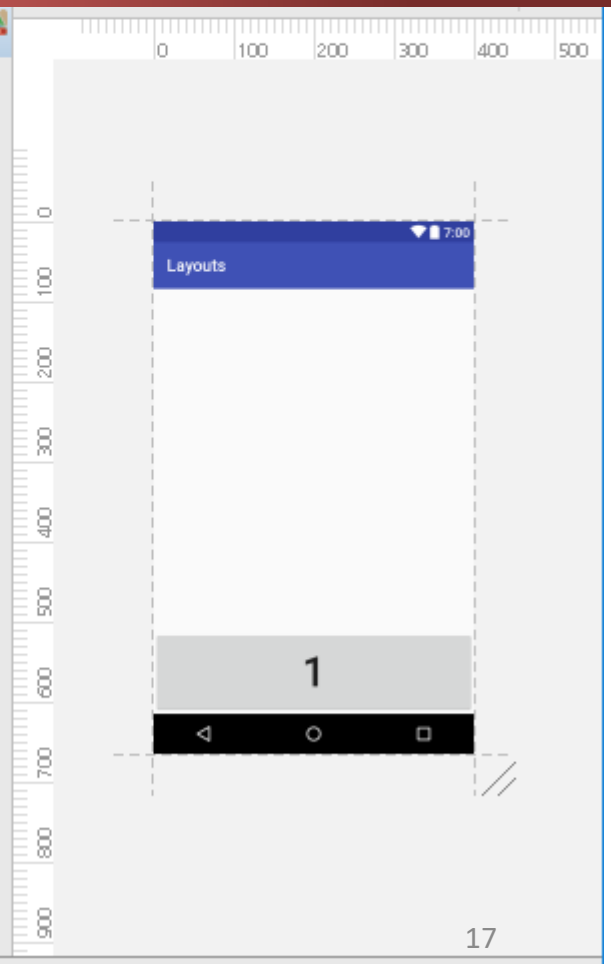


Probar el resultado de la siguiente disposición

RelativeLayout

layout_alignParentLeft: Alinea el borde izquierdo del View con el borde izquierdo de su contenedor

```
<RelativeLayout android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/
    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:layout_alignParentLeft="true"
        android:text="1"/>
</RelativeLayout>
```

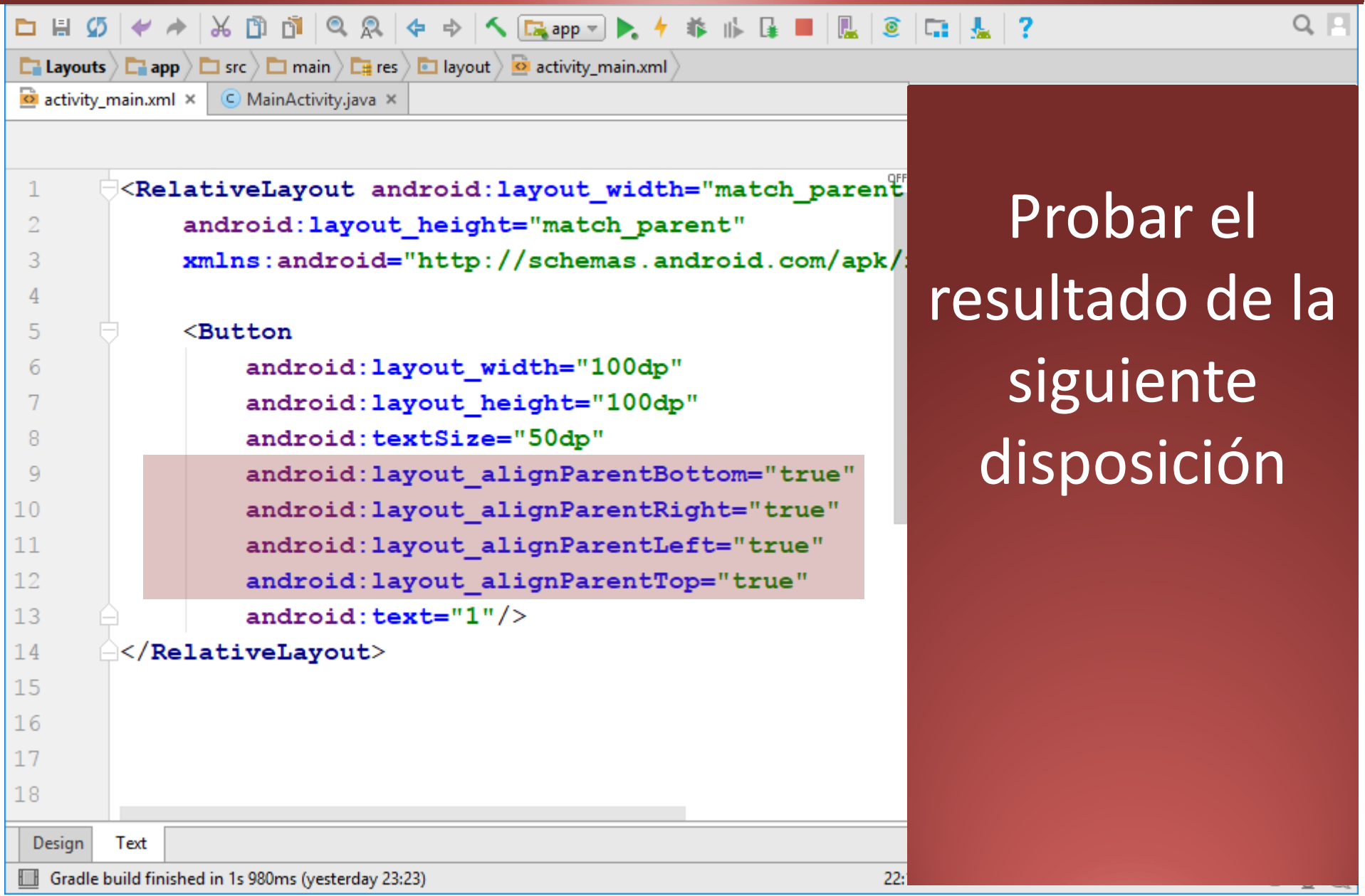


Design Text

Gradle build finished in 1s 980ms (yesterday 23:23) 24:1 CRLF UTF-8 Context: <no context>

La mayoría de los atributos que posicionan los `Views` dentro de un `RelativeLayout` afectan a uno de los bordes del `View`, es por eso que al combinarlos puede cambiar la dimensión de este elemento.

RelativeLayout

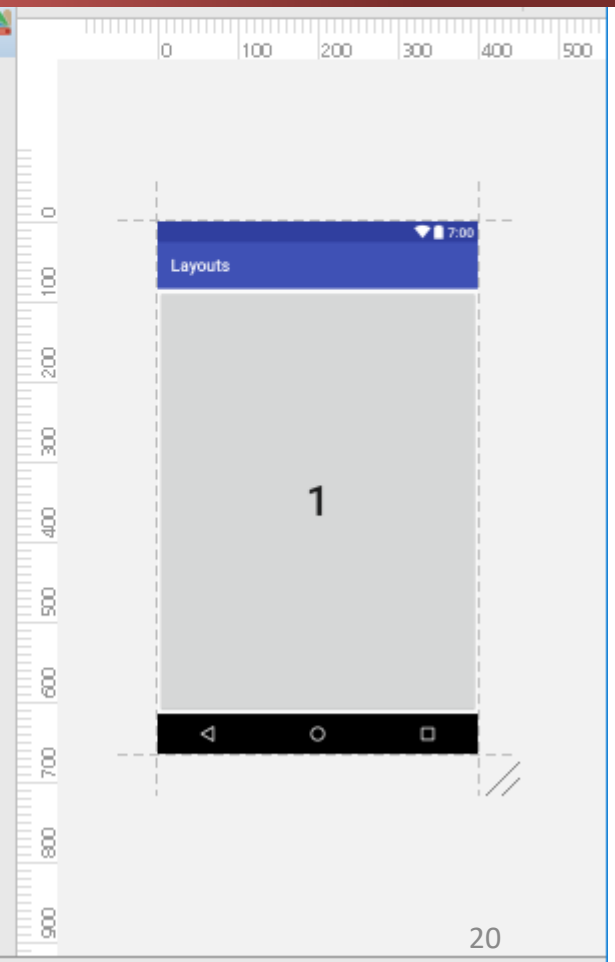


Probar el resultado de la siguiente disposición

RelativeLayout

layout_alignParentTop: Alinea el borde superior del View con el borde superior de su contenedor

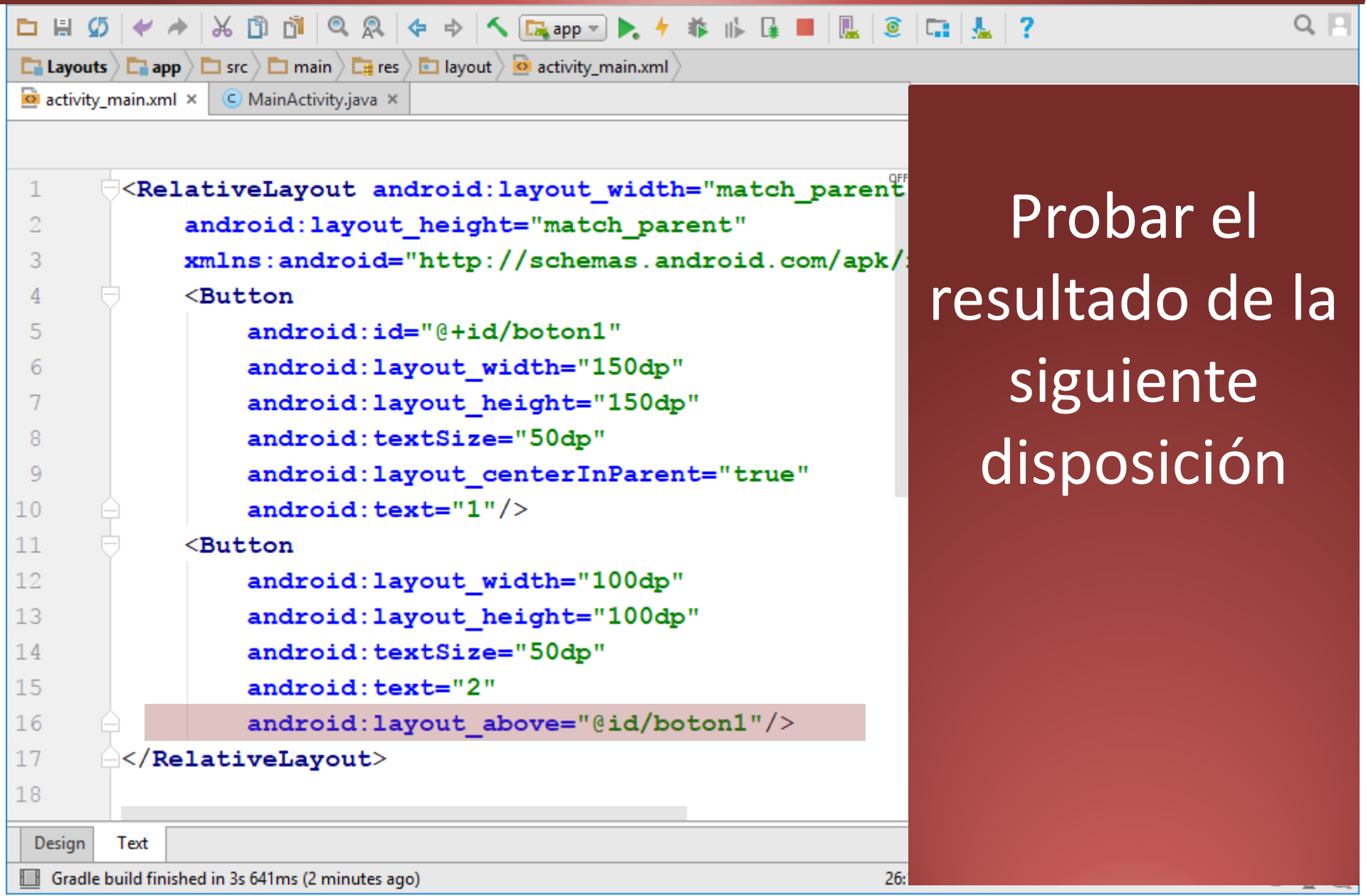
```
<RelativeLayout android:layout_width="match_parent
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/
    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:text="1"/>
</RelativeLayout>
```



Design Text

Gradle build finished in 1s 980ms (yesterday 23:23) 22:1 CRLF UTF-8 Context: <no context>

RelativeLayout

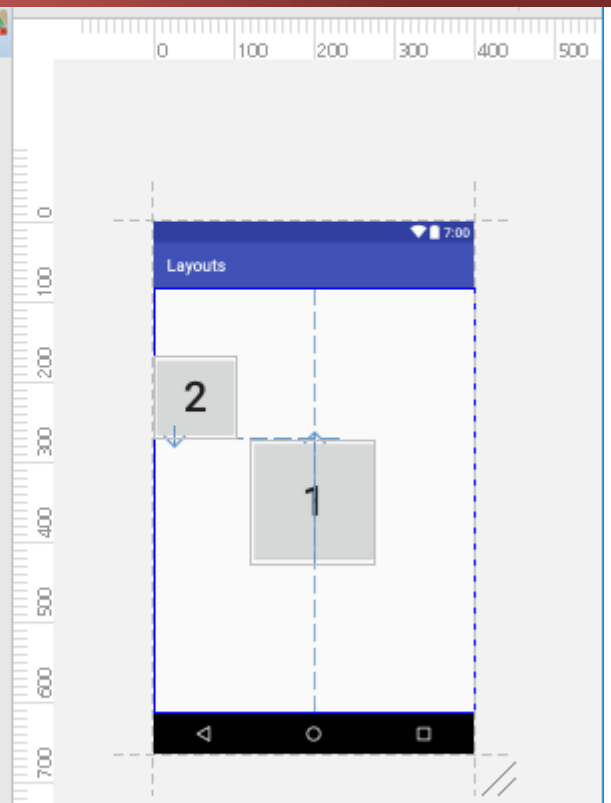


Probar el resultado de la siguiente disposición

RelativeLayout

layout_above: Alinea el borde inferior de un View con el borde superior de otro View

```
<RelativeLayout android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/
    <Button
        android:id="@+id/boton1"
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:textSize="50dp"
        android:layout_centerInParent="true"
        android:text="1"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:text="2"
        android:layout_above="@id/boton1"/>
</RelativeLayout>
```

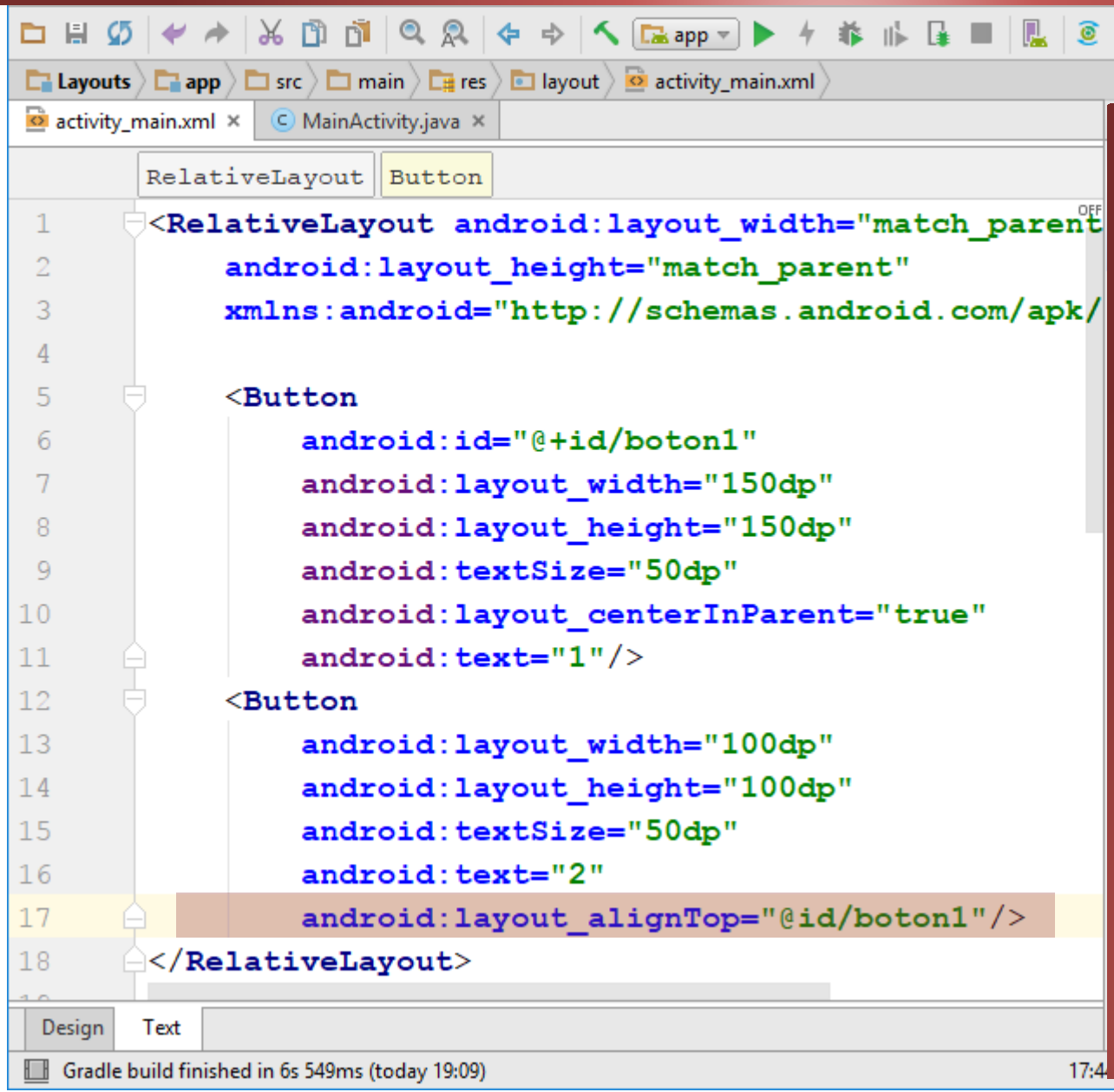


Design Text

Gradle build finished in 3s 641ms (2 minutes ago)

26:1 CRLF UTF-8 Context: <no context>

RelativeLayout

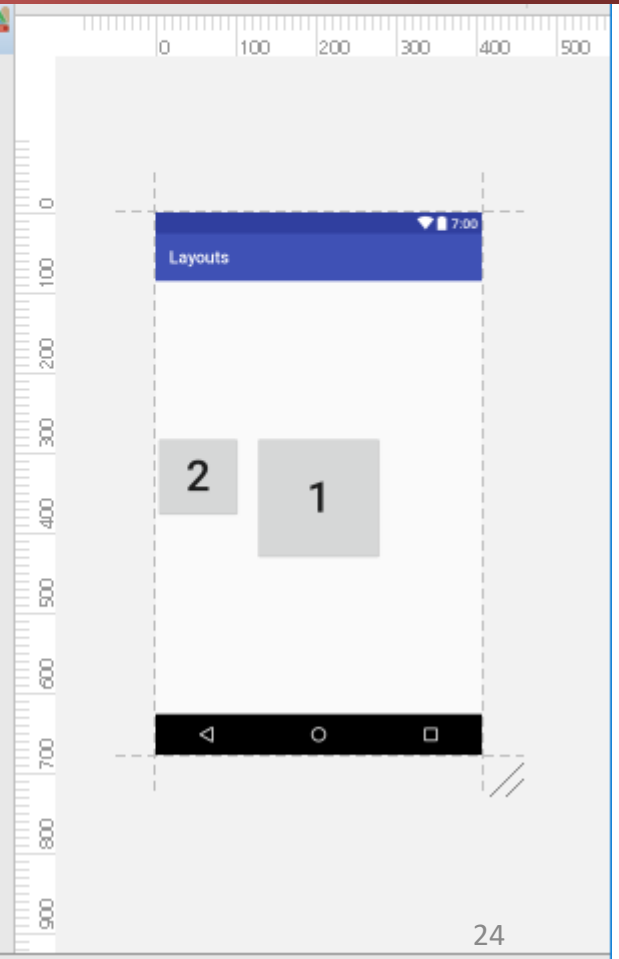


Probar el resultado de la siguiente disposición

RelativeLayout

layout_alignTop: Alinea el borde superior de un View con el borde superior de otro View

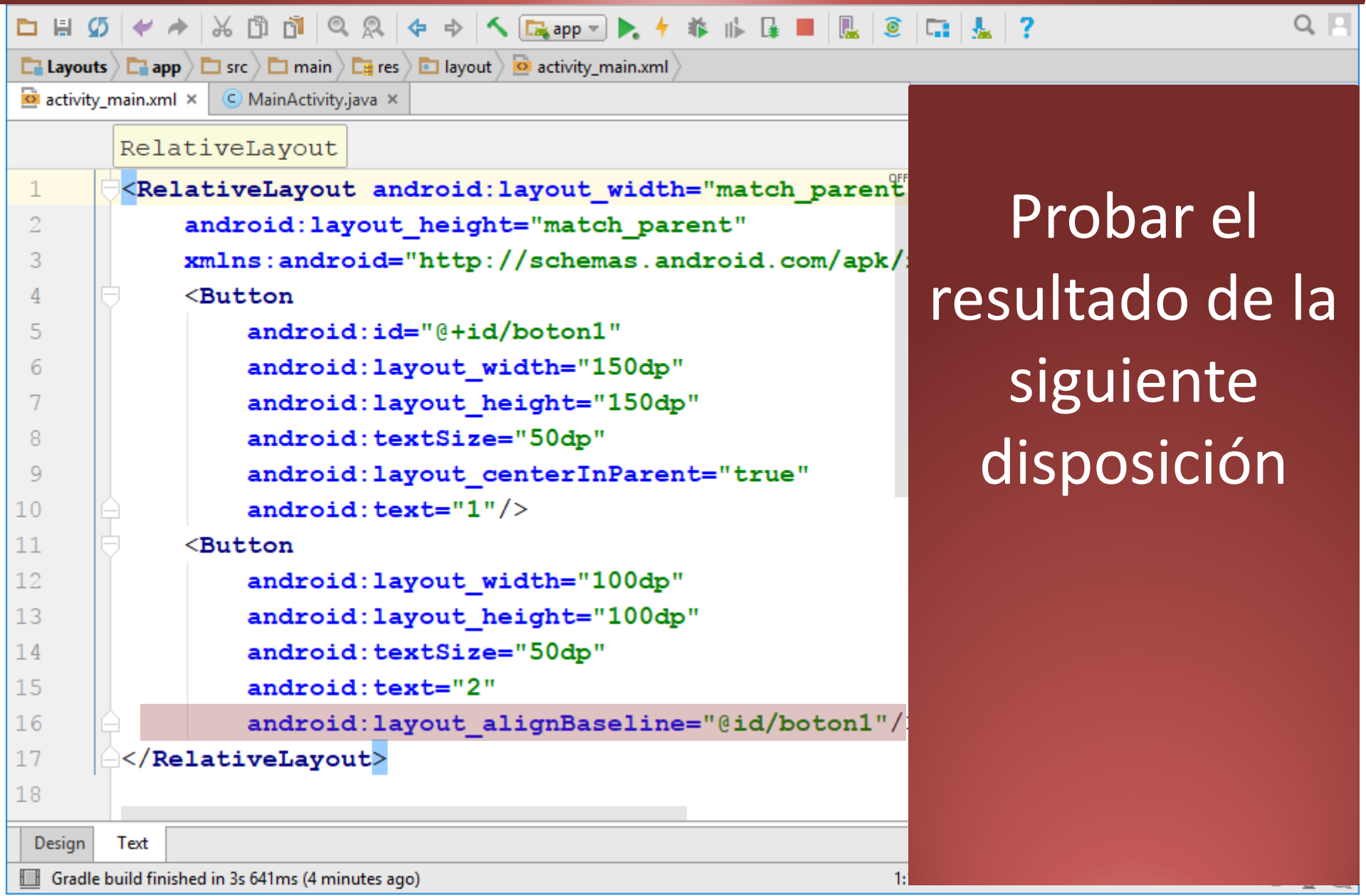
```
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/

    <Button
        android:id="@+id/boton1"
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:textSize="50dp"
        android:layout_centerInParent="true"
        android:text="1"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:text="2"
        android:layout_alignTop="@id/boton1"/>
</RelativeLayout>
```



Design Text

Gradle build finished in 6s 549ms (today 19:09) 17:44 CRLF UTF-8 Context: <no context>

RelativeLayout



Probar el resultado de la siguiente disposición

RelativeLayout

layout_alignBaseline: Alinea la línea base de un View con la línea base de otro View.

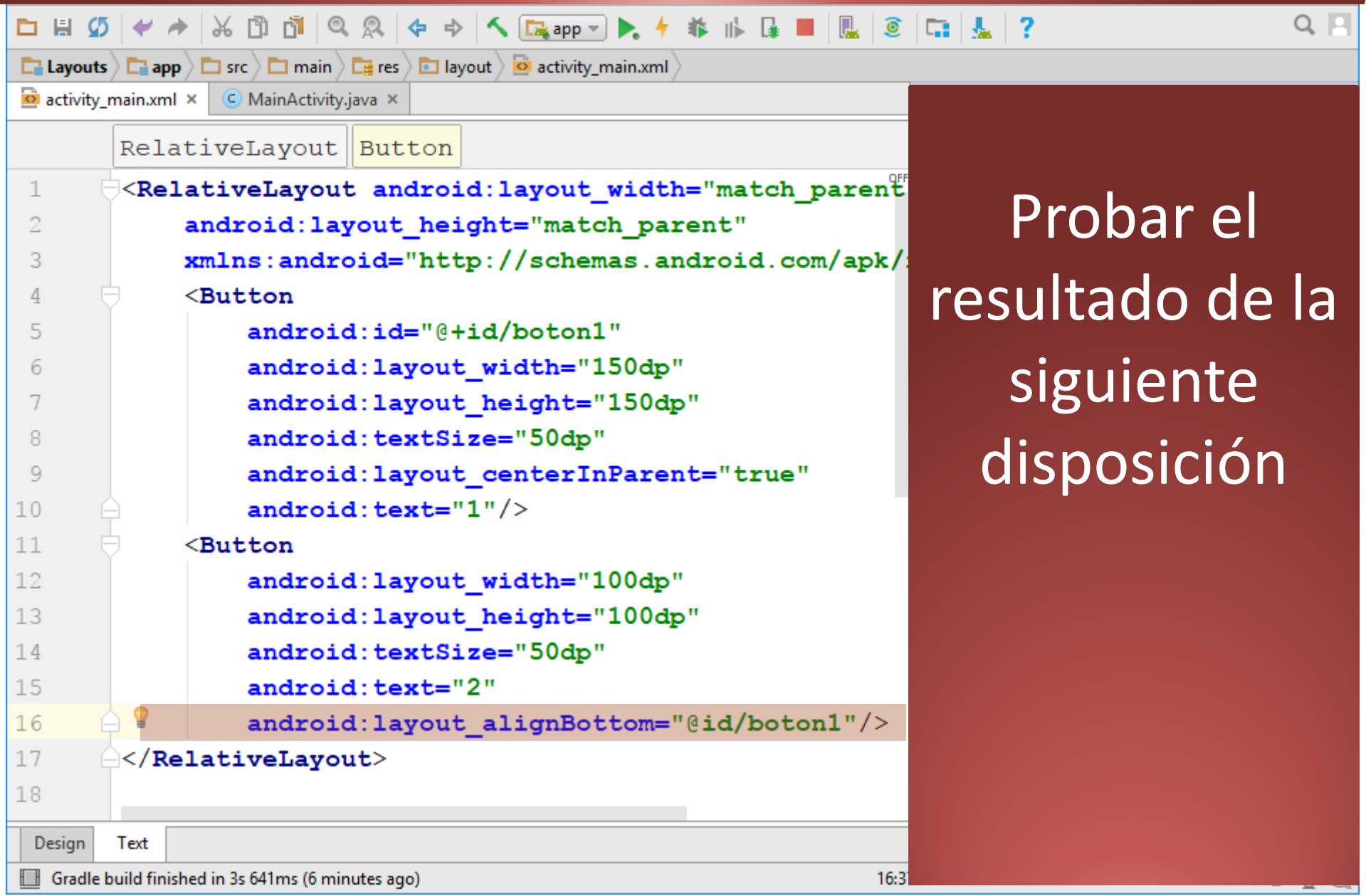
```
<RelativeLayout android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/
    <Button
        android:id="@+id/boton1"
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:textSize="50dp"
        android:layout_centerInParent="true"
        android:text="1"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:text="2"
        android:layout_alignBaseline="@id/boton1"/
</RelativeLayout>
```

Design Text

Gradle build finished in 3s 641ms (4 minutes ago)

1:1 CRLF UTF-8 Context: <no context>

RelativeLayout



Probar el resultado de la siguiente disposición

RelativeLayout

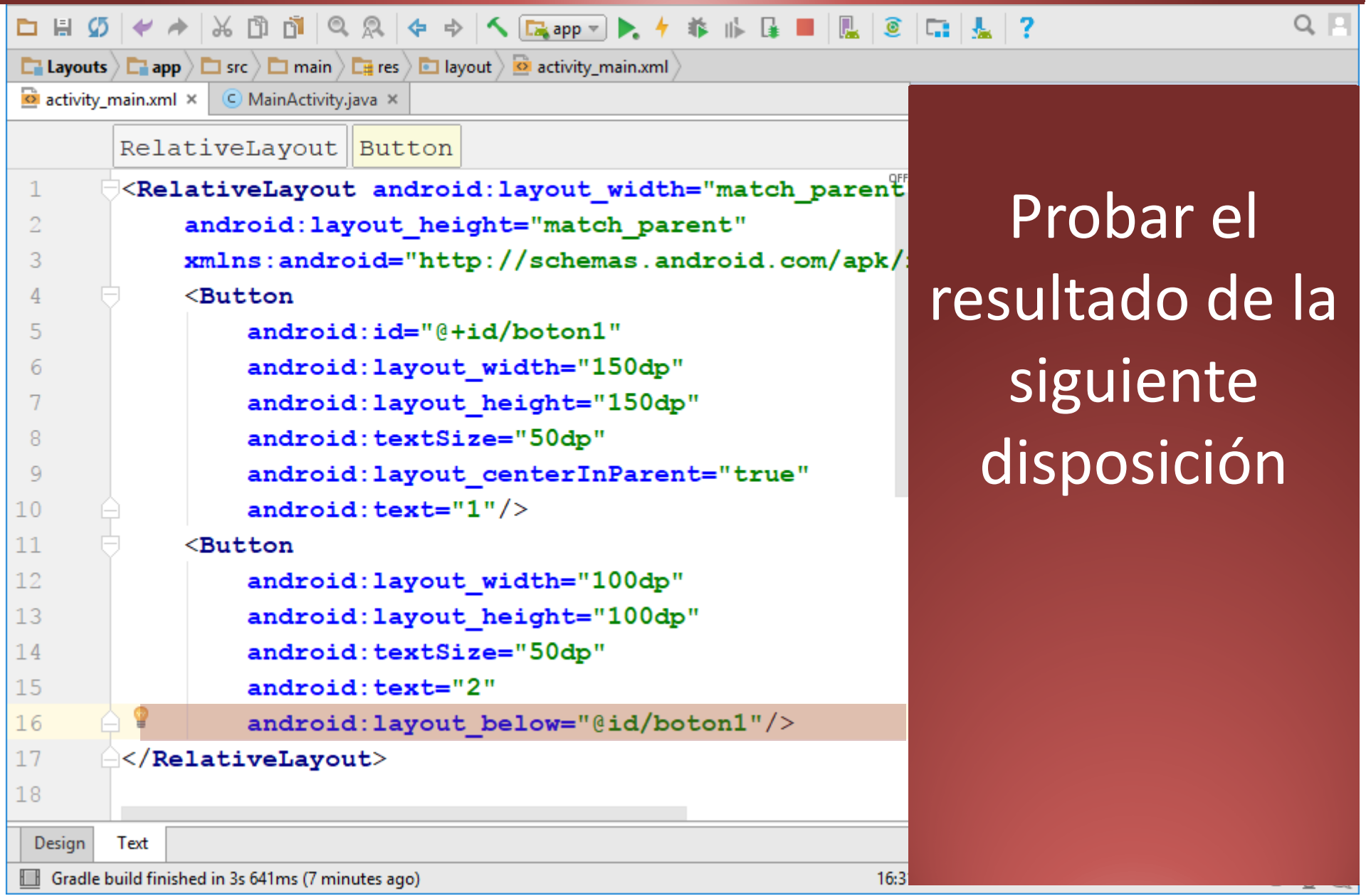
layout_alignBottom: Alinea el borde inferior de un View con el borde inferior de otro View

```
<RelativeLayout android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/
    <Button
        android:id="@+id/boton1"
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:textSize="50dp"
        android:layout_centerInParent="true"
        android:text="1"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:text="2"
        android:layout_alignBottom="@id/boton1"/>
</RelativeLayout>
```

Design Text

Gradle build finished in 3s 641ms (6 minutes ago) 16:37 CRLF UTF-8 Context: <no context>

RelativeLayout

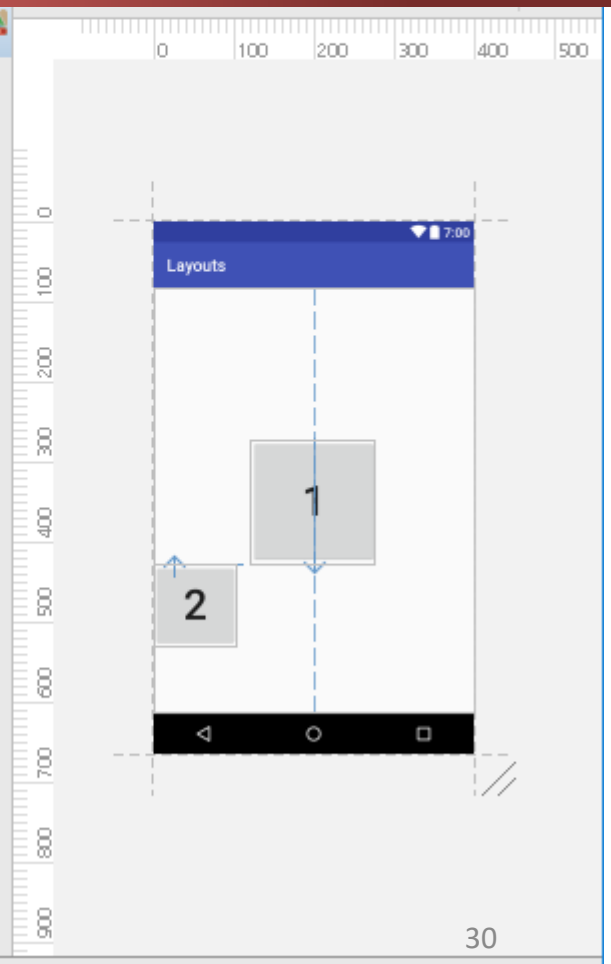


Probar el resultado de la siguiente disposición

RelativeLayout

layout_below: Alinea el borde superior de un View con el borde inferior de otro View

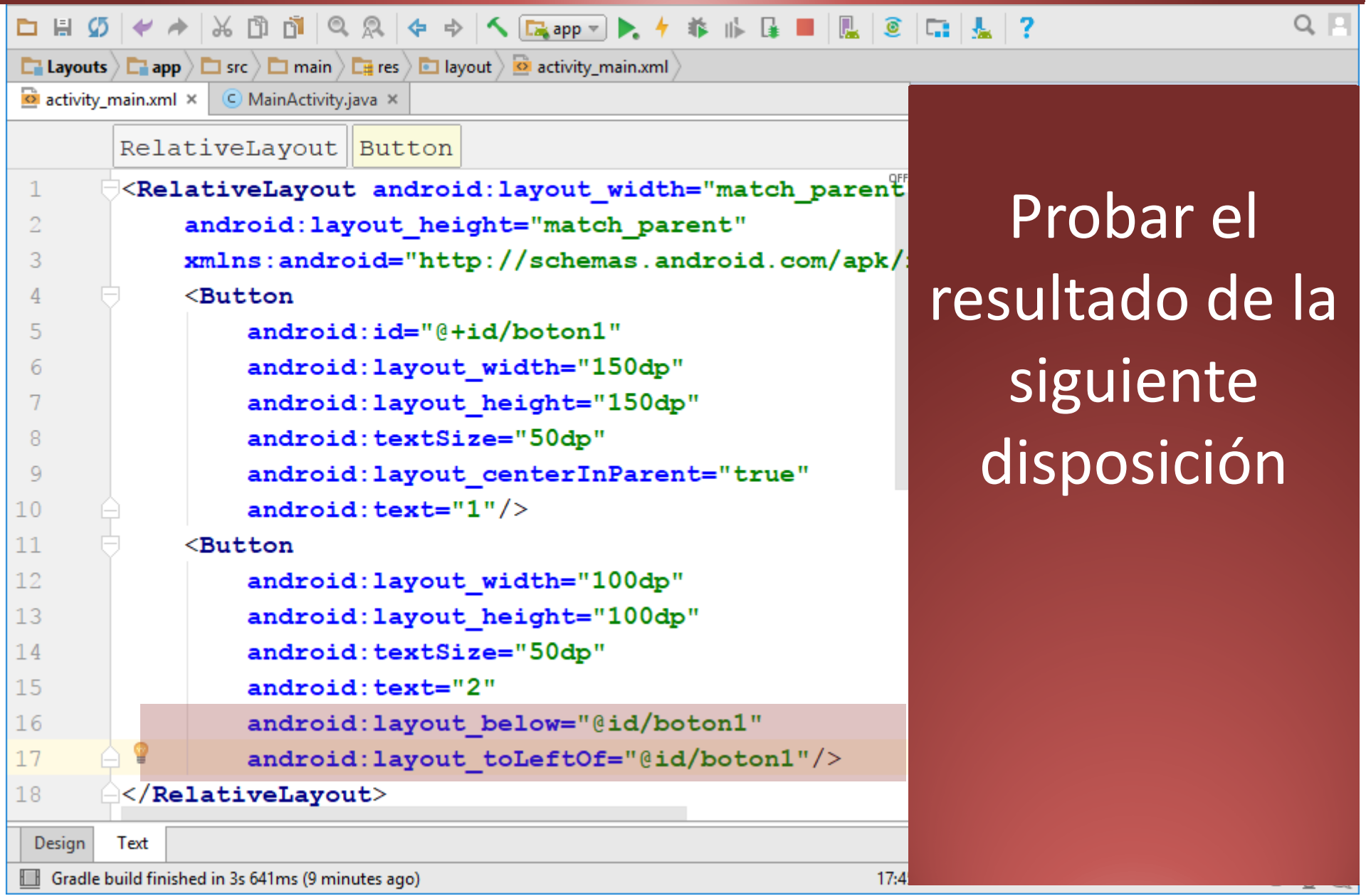
```
<RelativeLayout android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/
    <Button
        android:id="@+id/boton1"
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:textSize="50dp"
        android:layout_centerInParent="true"
        android:text="1"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:text="2"
        android:layout_below="@id/boton1"/>
</RelativeLayout>
```



Design Text

Gradle build finished in 3s 641ms (7 minutes ago) 16:31 CRLF UTF-8 Context: <no context>

RelativeLayout

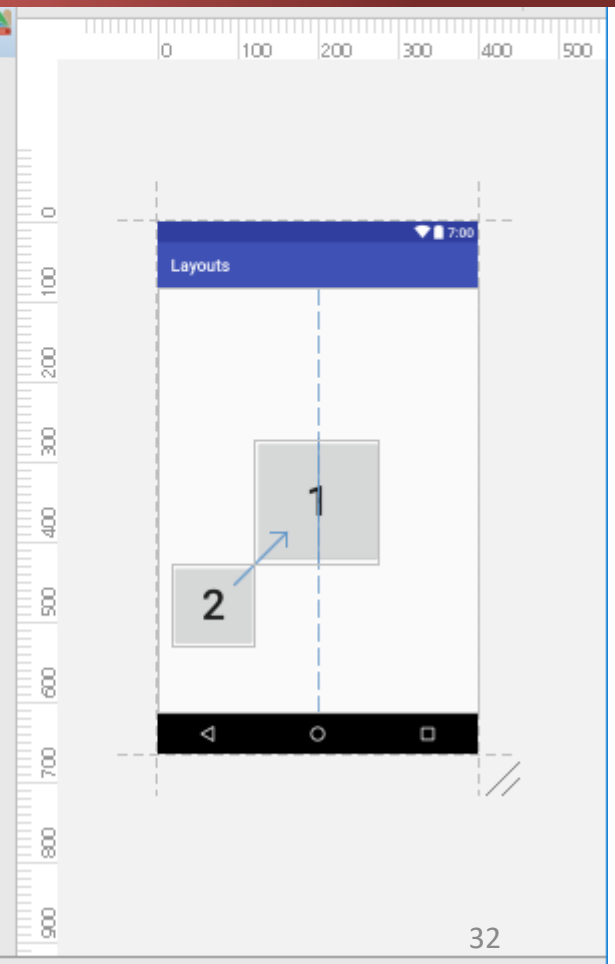


Probar el resultado de la siguiente disposición

RelativeLayout

layout_toLeftOf: Alinea el borde derecho de un View con el borde izquierdo de otro View

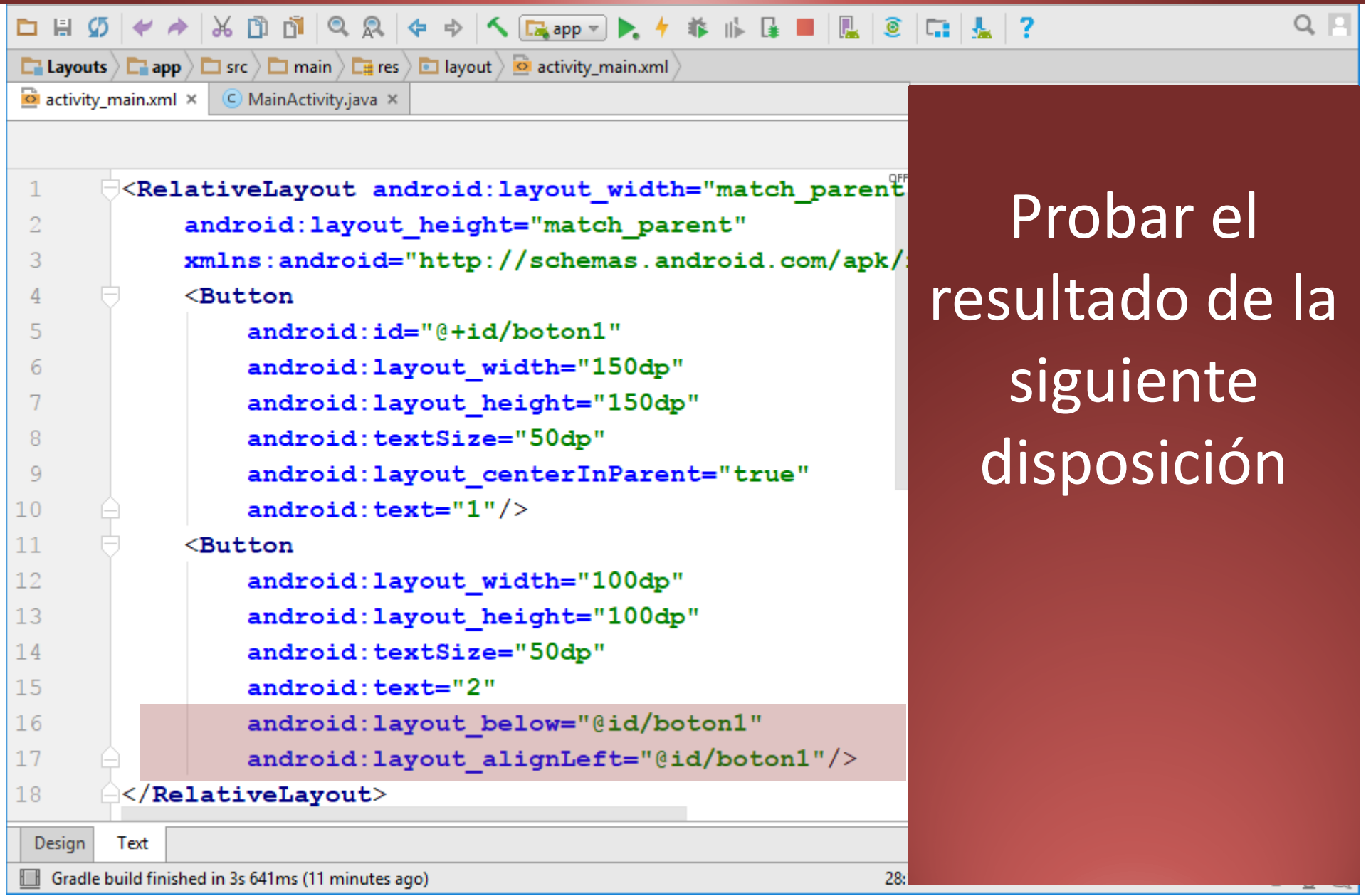
```
<RelativeLayout android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/
    <Button
        android:id="@+id/boton1"
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:textSize="50dp"
        android:layout_centerInParent="true"
        android:text="1"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:text="2"
        android:layout_below="@id/boton1"
        android:layout_toLeftOf="@id/boton1"/>
</RelativeLayout>
```



Design Text

Gradle build finished in 3s 641ms (9 minutes ago) 17:45 CRLF UTF-8 Context: <no context>

RelativeLayout

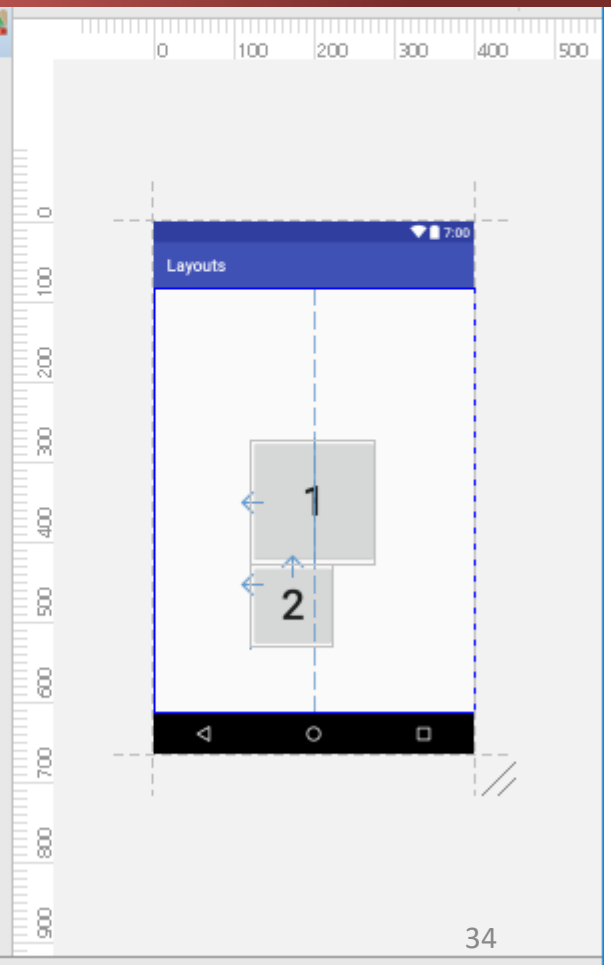


Probar el resultado de la siguiente disposición

RelativeLayout

layout_alignLeft: Alinea el borde izquierdo de un View con el borde izquierdo de otro View.

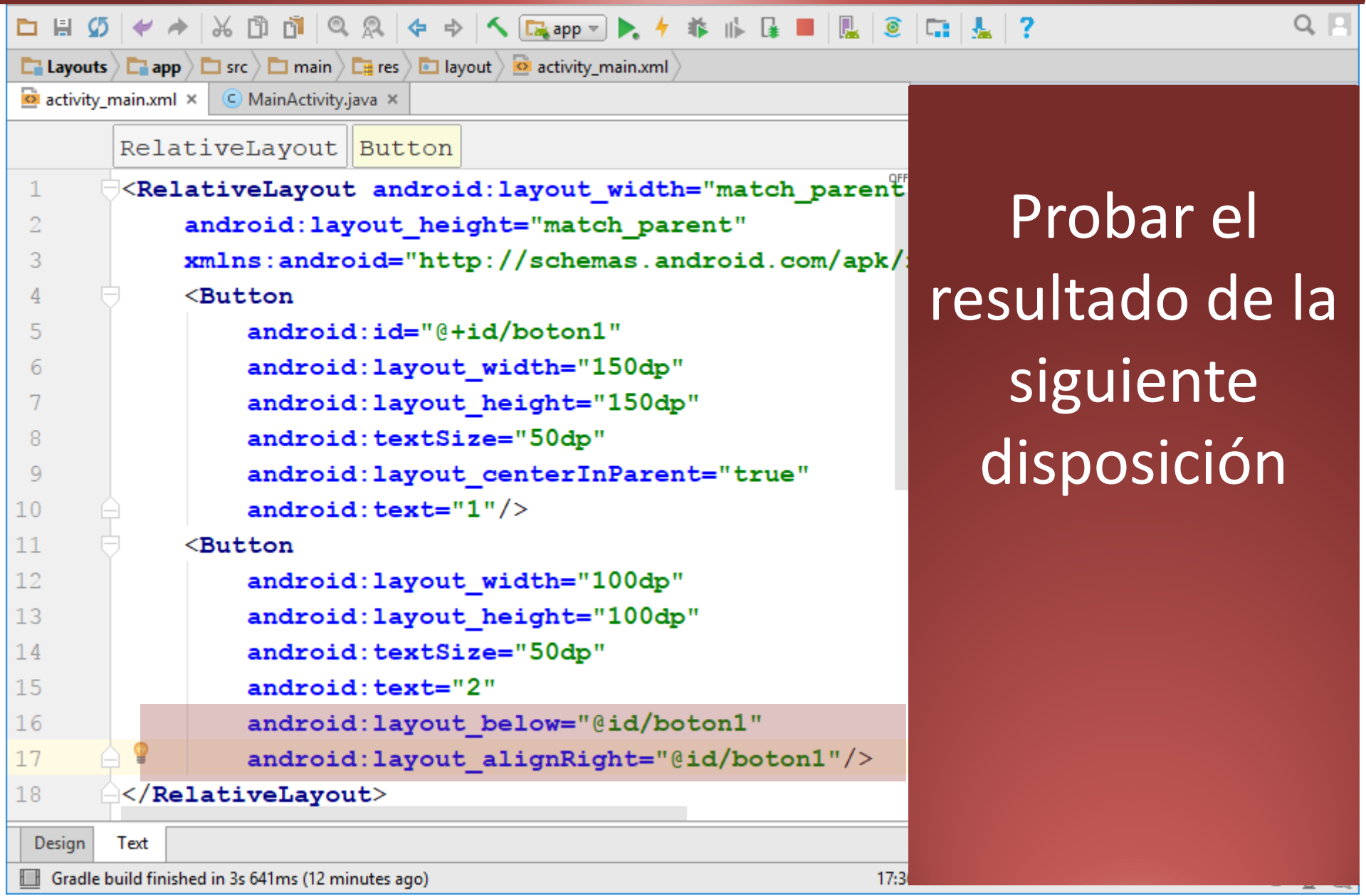
```
<RelativeLayout android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/
    <Button
        android:id="@+id/boton1"
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:textSize="50dp"
        android:layout_centerInParent="true"
        android:text="1"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:text="2"
        android:layout_below="@id/boton1"
        android:layout_alignLeft="@id/boton1"/>
</RelativeLayout>
```



Design Text

Gradle build finished in 3s 641ms (11 minutes ago) 28:1 CRLF UTF-8 Context: <no context>

RelativeLayout



Probar el resultado de la siguiente disposición

RelativeLayout

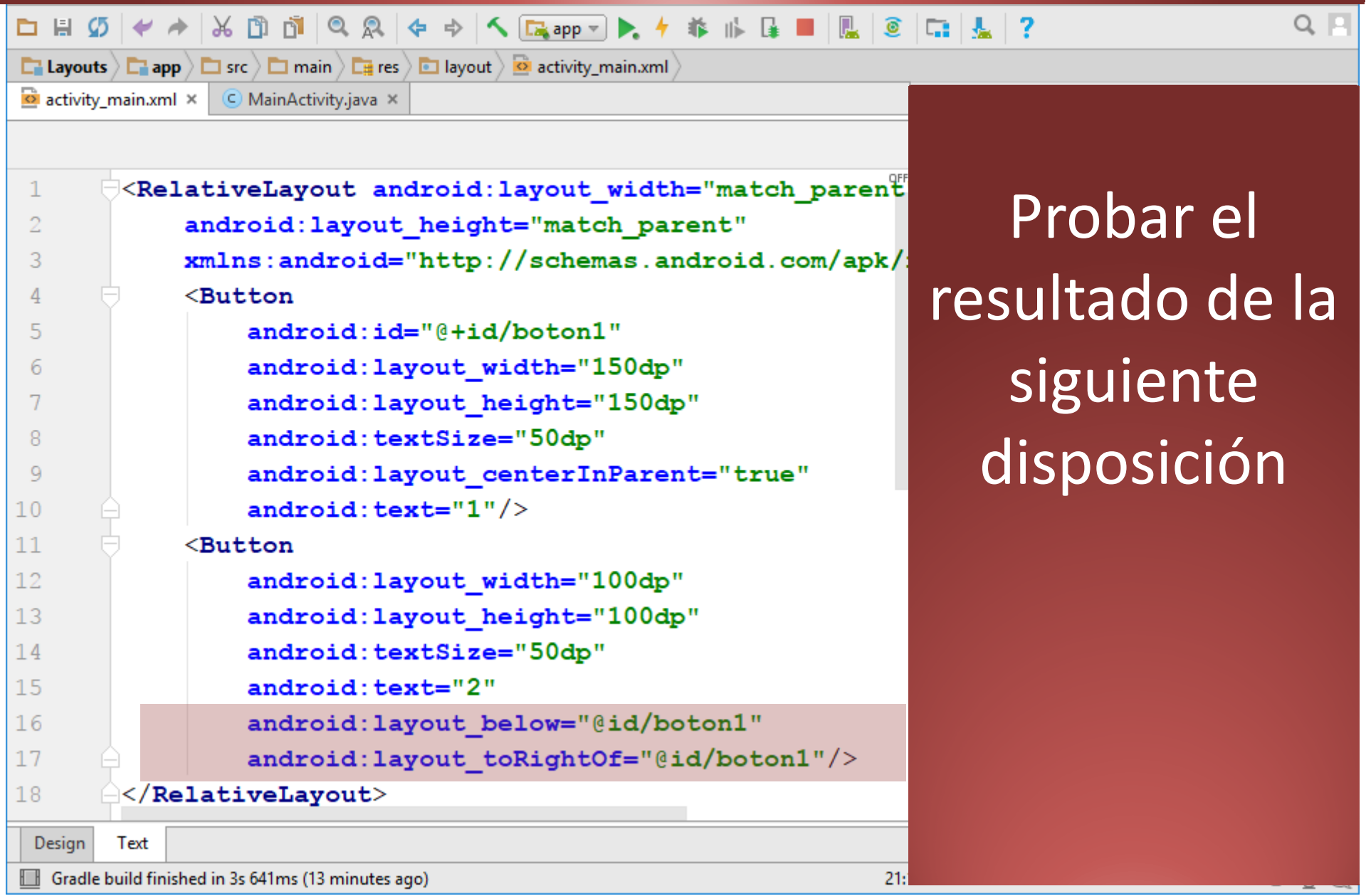
layout_alignRight: Alinea el borde derecho de un View con el borde derecho de otro View

```
<RelativeLayout android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/
    <Button
        android:id="@+id/boton1"
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:textSize="50dp"
        android:layout_centerInParent="true"
        android:text="1"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:text="2"
        android:layout_below="@id/boton1"
        android:layout_alignRight="@id/boton1"/>
</RelativeLayout>
```

Design Text

Gradle build finished in 3s 641ms (12 minutes ago) 17:36 CRLF UTF-8 Context: <no context>

RelativeLayout

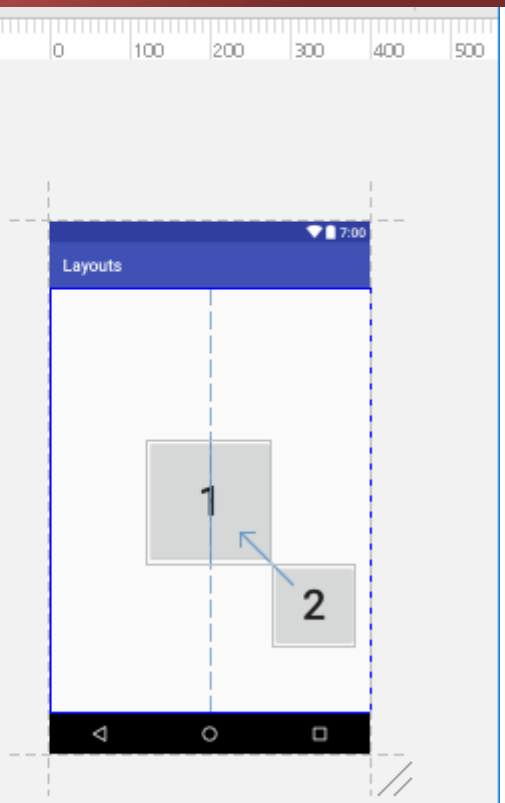


Probar el resultado de la siguiente disposición

RelativeLayout

layout_toRightOf: Alinea el borde izquierdo de un View con el borde derecho de otro View

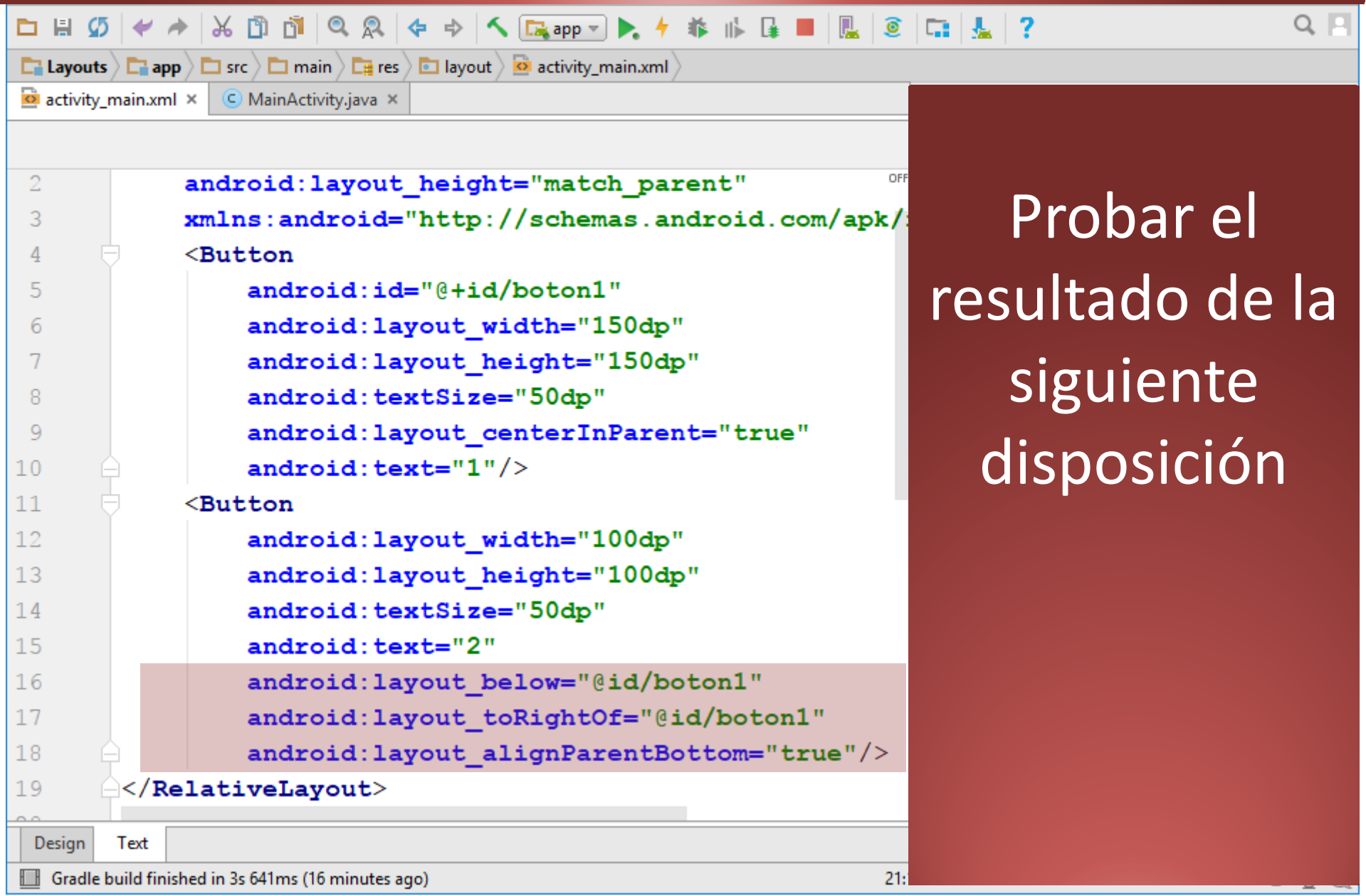
```
<RelativeLayout android:layout_width="match_parent"
    android:layout_height="match_parent"
    xmlns:android="http://schemas.android.com/apk/
    <Button
        android:id="@+id/boton1"
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:textSize="50dp"
        android:layout_centerInParent="true"
        android:text="1"/>
    <Button
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:textSize="50dp"
        android:text="2"
        android:layout_below="@id/boton1"
        android:layout_toRightOf="@id/boton1"/>
</RelativeLayout>
```



Design Text

Gradle build finished in 3s 641ms (13 minutes ago) 21:1 CRLF UTF-8 Context: <no context>

RelativeLayout

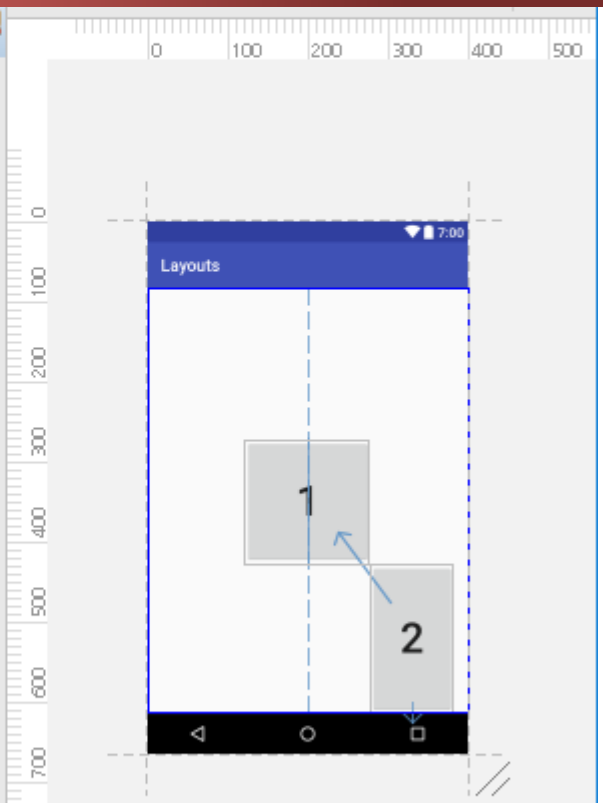


Probar el resultado de la siguiente disposición

RelativeLayout

Se pueden combinar distintas disposiciones para obtener resultados muy variados

```
3   xmlns:android="http://schemas.android.com/apk/
4   <Button
5       android:id="@+id/boton1"
6       android:layout_width="150dp"
7       android:layout_height="150dp"
8       android:textSize="50dp"
9       android:layout_centerInParent="true"
10      android:text="1"/>
11  <Button
12      android:layout_width="100dp"
13      android:layout_height="100dp"
14      android:textSize="50dp"
15      android:text="2"
16      android:layout_below="@id/boton1"
17      android:layout_toRightOf="@id/boton1"
18      android:layout_alignParentBottom="true"/>
19  </RelativeLayout>
```



Design Text

Gradle build finished in 3s 641ms (16 minutes ago)

21:1 CRLF UTF-8 Context: <no context>

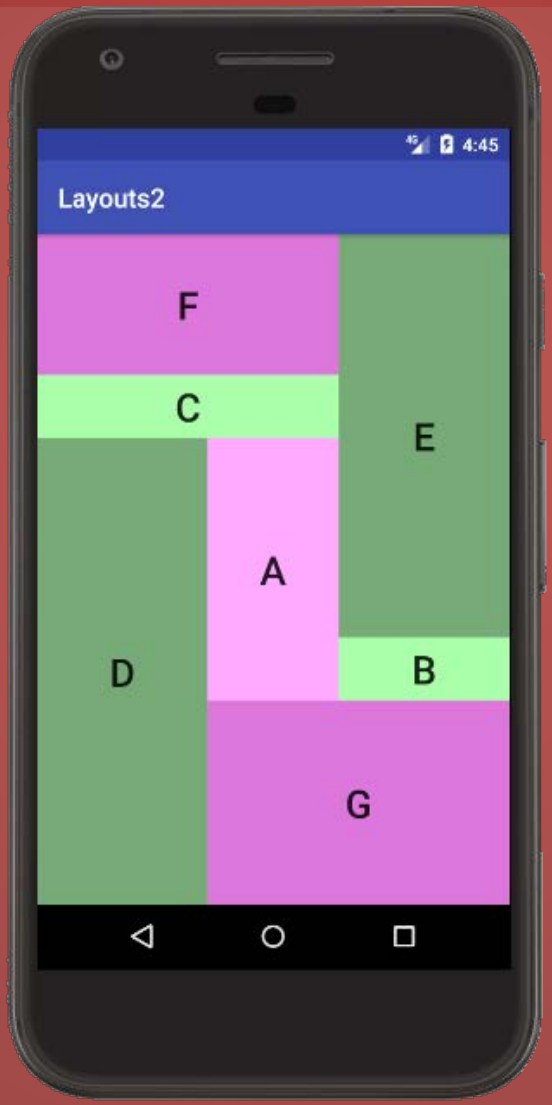
Relacionado con las alineaciones horizontales y teniendo en cuenta los idiomas RTL también existen:

- layout_alignParentStart
- layout_alignParentEnd
- layout_alignStart
- layout_alignEnd
- layout_toStartOf
- layout_toEndOf

Para el idioma español Start funciona como Left y End funciona como Right. Para los idiomas RTL justo lo contrario

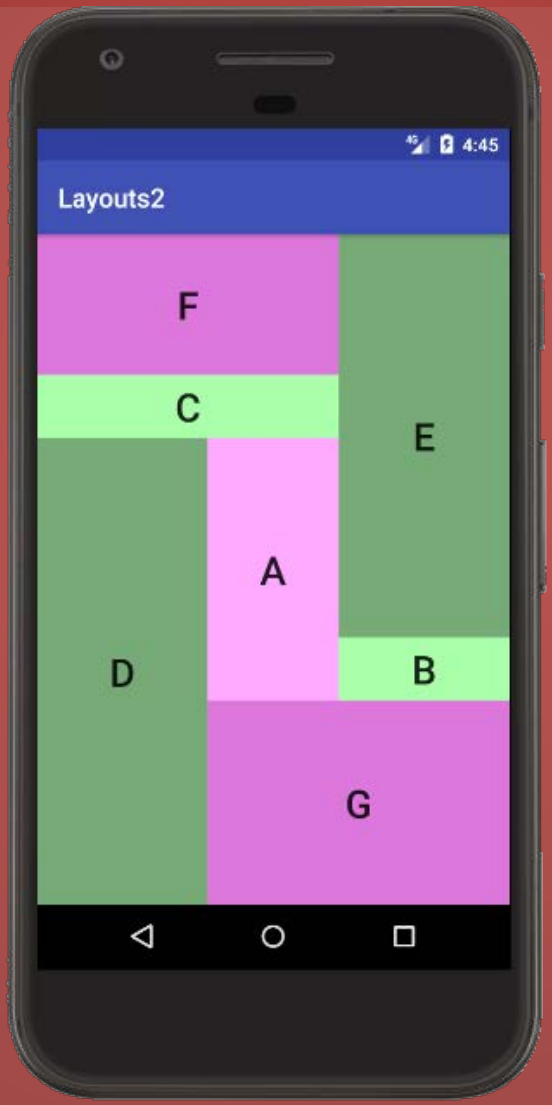
Ejercicio

Diseñar la siguiente interface (los `views` son `Buttons` a los que se les ha cambiado el color del `background`)

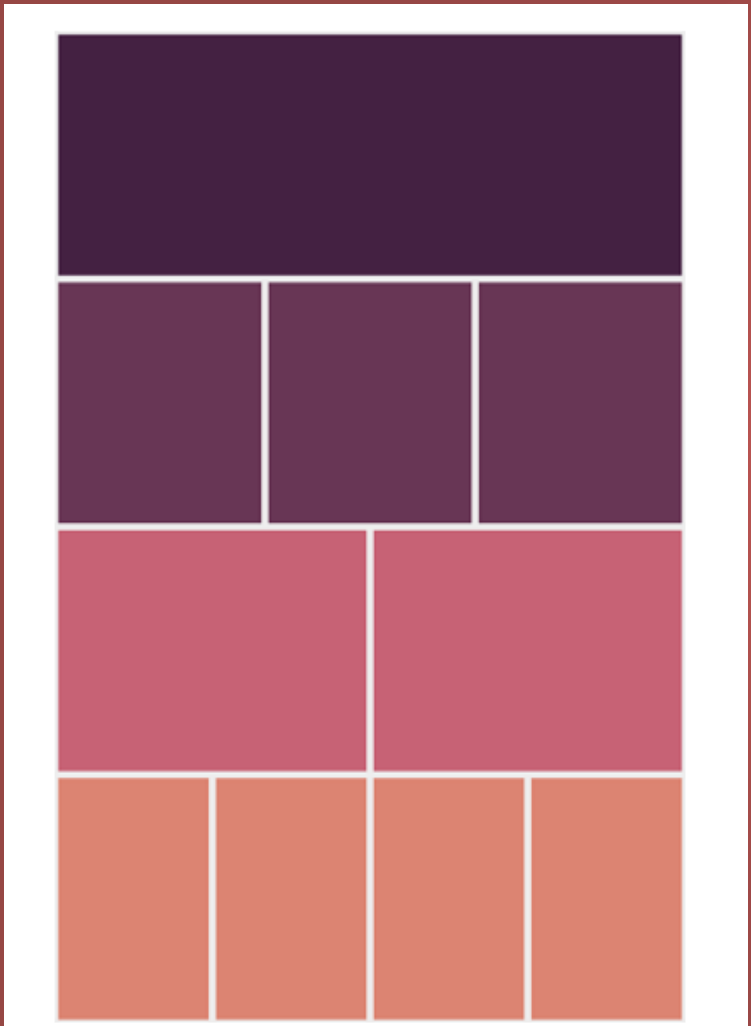


Ejercicio

Diseñar la siguiente interface (los `views` son `Buttons` a los que se les ha cambiado el color del `background`)



TableLayout

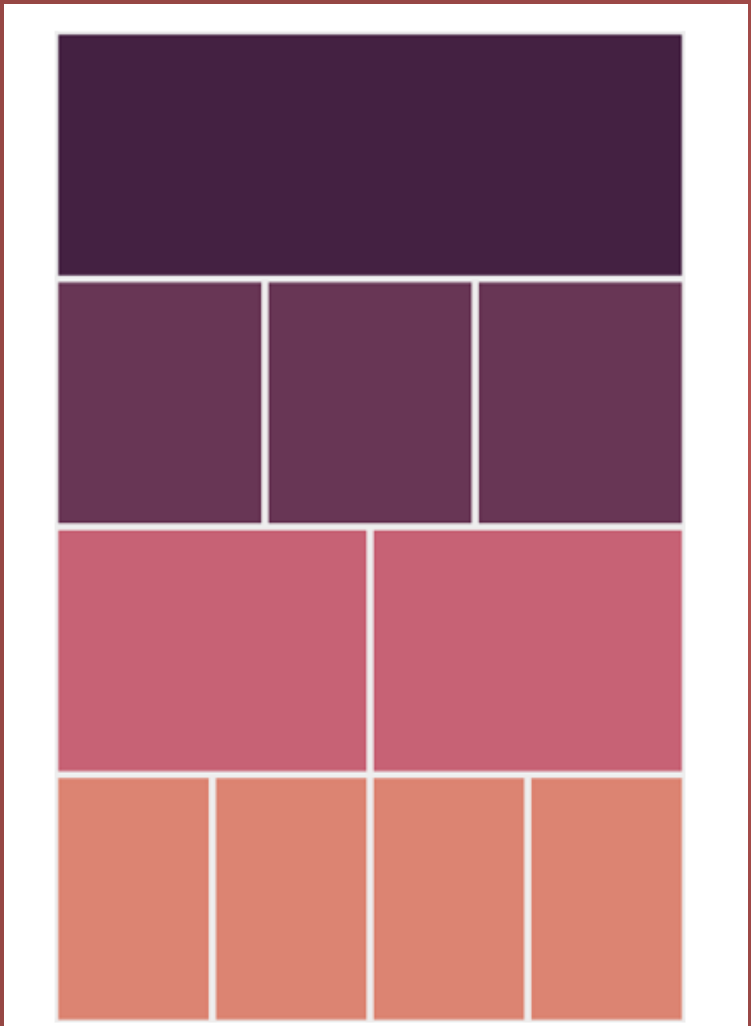


Es un ViewGroup que ubica a sus hijos en filas y columnas

Generalmente está compuesto de objetos TableRow.

Cada TableRow puede tener 0, 1 o más celdas.

TableLayout

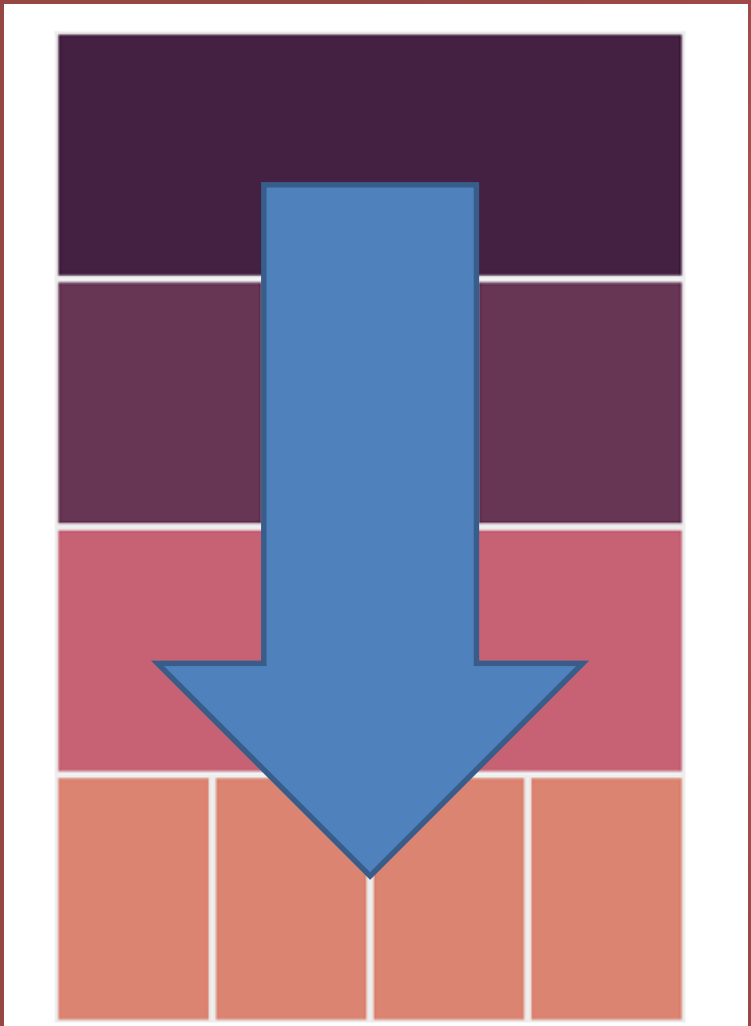


La tabla tendrá tantas columnas como aquel TableRow con más celdas.

Una celda podría expandirse ocupando más de una columna.

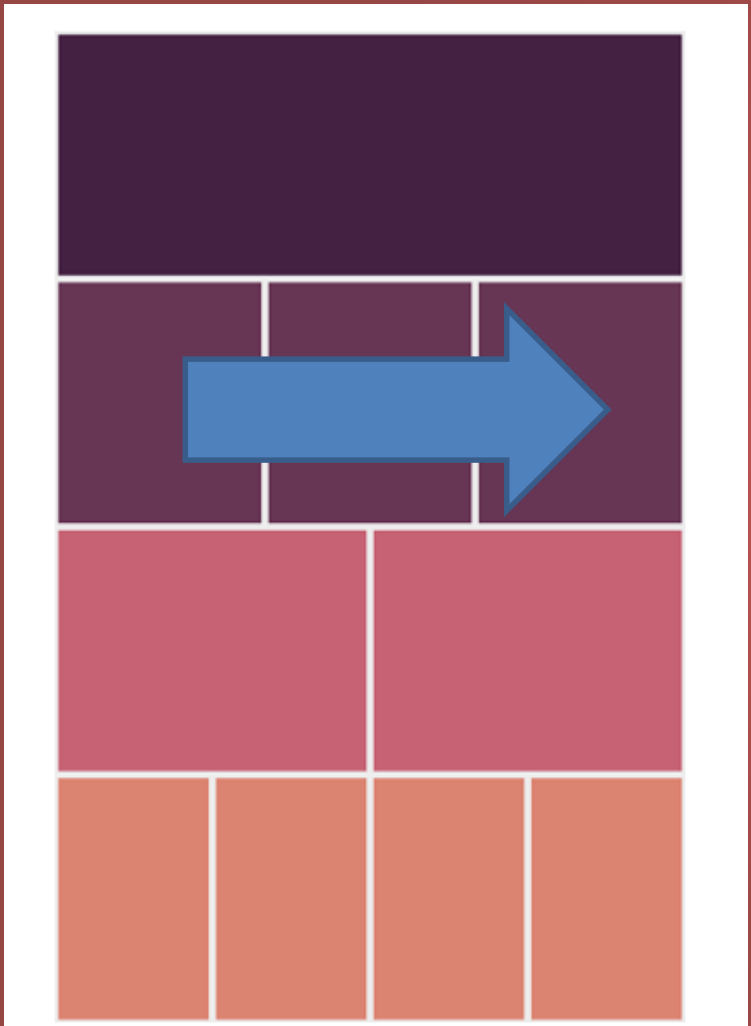
Un TableLayout no contempla bordes. Simplemente organiza elementos visuales

TableLayout



Un TableLayout es una especialización de un LinearLayout con orientación vertical.

TableLayout



Asimismo, cada TableRow es una especialización de un LinearLayout con orientación horizontal.

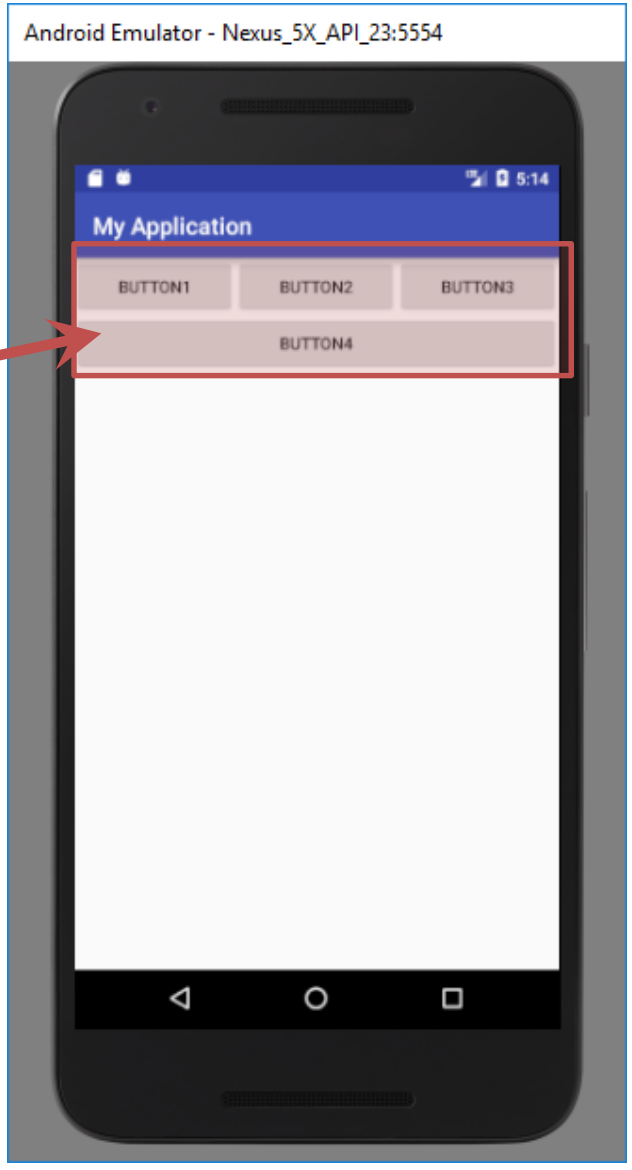
Ejemplo de TableLayout

```
<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:stretchColumns="*">
    <TableRow >
        <Button
            android:id="@+id/button1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="button1" />
        <Button
            android:id="@+id/button2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="button2" />
        <Button
            android:id="@+id/button3"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="button3" />
    </TableRow>
    <TableRow>
        <Button
            android:id="@+id/button4"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_span="3"
            android:text="button4" />
    </TableRow>
</TableLayout>
```

Ejemplo de TableLayout

```
<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:stretchColumns="*">
    <TableRow >
        <Button
            android:id="@+id/button1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android
        <Button
            android
            android
            android
            android
        <Button
            android:id="@+id/button3"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="button3" />
    </TableRow>
    <TableRow>
        <Button
            android:id="@+id/button4"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_span="3"
            android:text="button4" />
    </TableRow>
</TableLayout>
```

Estira las columnas para ocupar todo el ancho del TableLayout

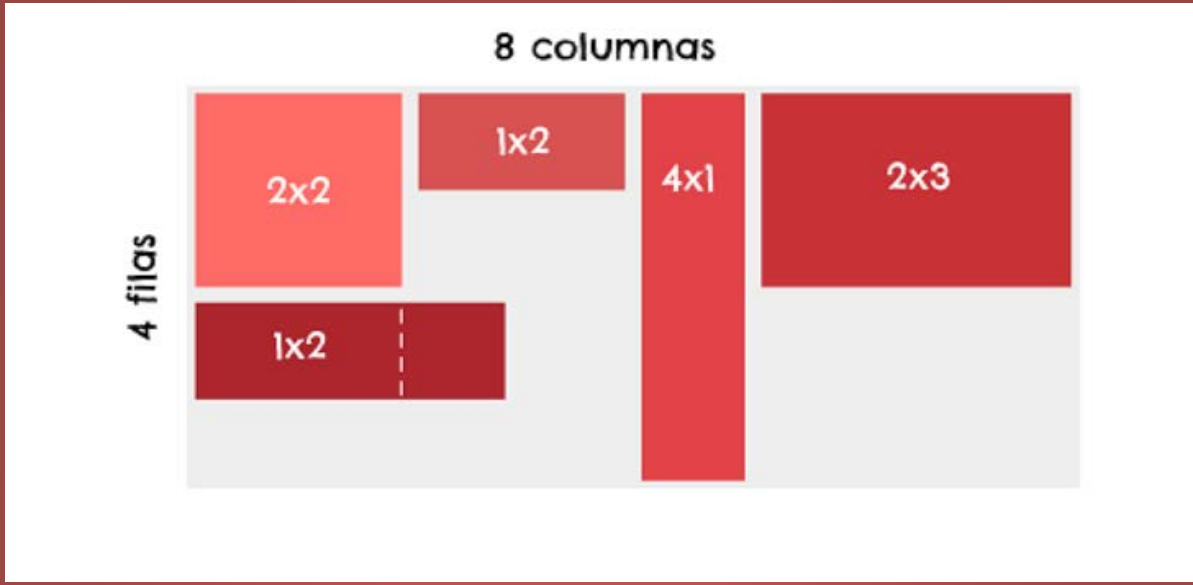


Ejemplo de TableLayout

```
<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:stretchColumns="*">
    <TableRow >
        <Button
            android:id="@+id/button1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="button1" />
        <Button
            android:id="@+id/button2"
            android:layout_width="wrap_content"
            android:layout_hei
            android:text="butt
        <Button
            android:id="@+id/b
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="button3" />
    </TableRow>
    <TableRow>
        <Button
            android:id="@+id/button4"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_span="3"
            android:text="button4" />
    </TableRow>
</TableLayout>
```

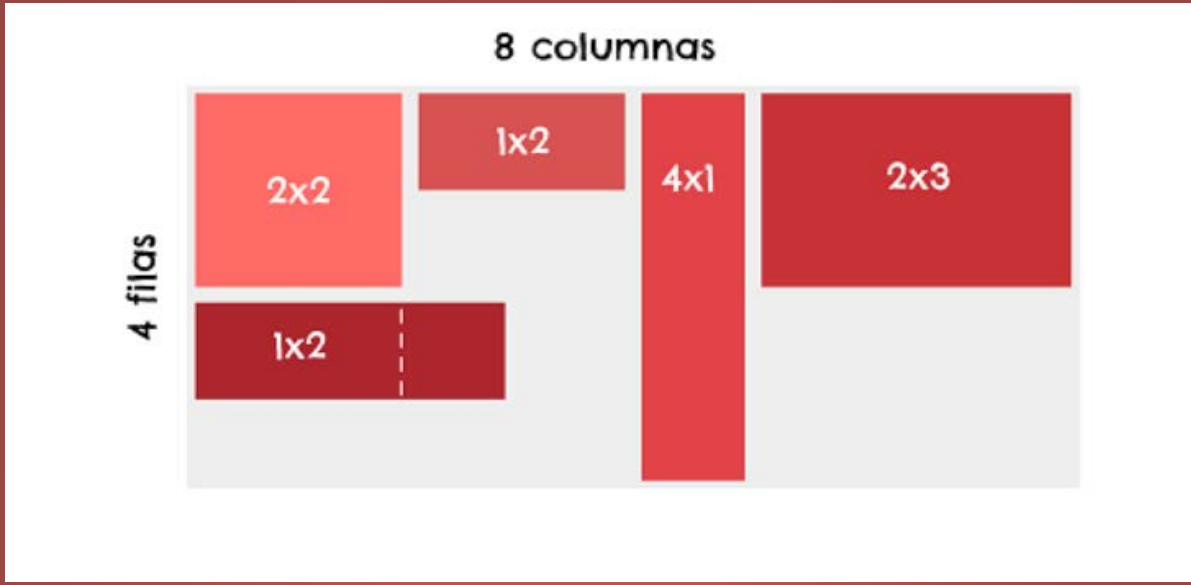
La vista ocupa 3 columnas

GridLayout



- Incluido a partir de Andriod 4.0 (API 14)
- Es un ViewGroup que ubica a sus hijos en una grilla rectangular.
- Es posible especificar cantidad de filas y columnas.
- A diferencia del TableLayout, es posible expandir una celda de forma horizontal o vertical.

GridLayout



- No existe un concepto análogo a TableRow. Los elementos hijos se irán colocando ordenadamente por filas o columnas (dependiendo de la propiedad android:orientation) hasta completar el número de filas o columnas.

Ejemplo de GridLayout

```
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:columnCount="4"
    android:orientation="horizontal" >

    <Button android:text="Botón 1.1" />
    <Button android:text="Botón 1.2" />
    <Button android:text="Botón 1.3"
            android:layout_rowSpan="2"/>
    <Button android:text="Botón 1.4" />

    <Button android:text="Botón 2.1"
            android:layout_columnSpan="2" />
    <Button android:text="Botón 2.4" />

</GridLayout>
```

Ejemplo de GridLayout

```
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:columnCount="4"
    android:orientation="horizontal" >

    <Button android:text="Botón 1.1" />
    <Button android:text="Botón 1.2" />
    <Button android:text="Botón 1.3"
            android:layout_rowSpan="2"/>
    <Button android:text="Botón 1.4" />

    <Button android:text="Botón 2.1"
            android:layout_columnSpan="2" />
    <Button android:text="Botón 2.4" />

</GridLayout>
```

La vista ocupa 2 filas

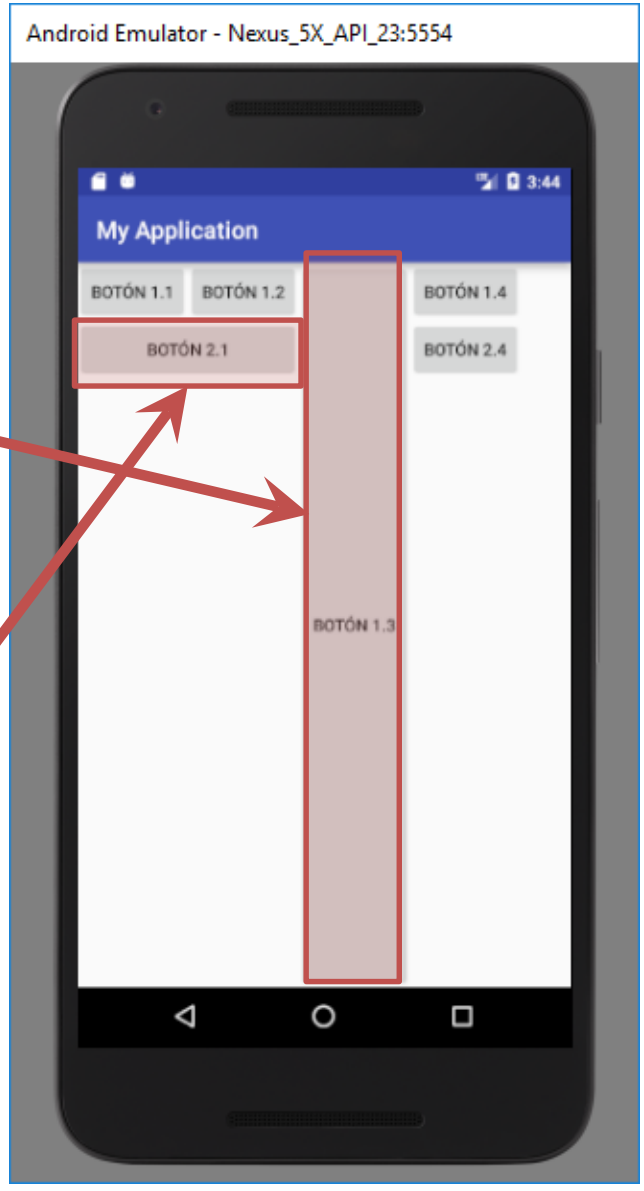
Ejemplo de GridLayout

```
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:columnCount="4"
    android:orientation="horizontal" >

    <Button android:text="Botón 1.1" />
    <Button android:text="Botón 1.2" />
    <Button android:text="Botón 1.3"
            android:layout_rowSpan="2"/>
    <Button android:text="Botón 1.4" />

    <Button android:text="Botón 2.1"
            android:layout_columnSpan="2" />
    <Button android:text="Botón 2.4" />

</GridLayout>
```

La vista ocupa 2 columnas

Ejemplo de GridLayout

```
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:columnCount="4"
    android:orientation="horizontal" >

    <Button android:text="Botón 1.1" />
    <Button android:text="Botón 1.2" />
    <Button android:text="Botón 1.3"
            android:layout_rowSpan="2"
            android:layout_gravity="fill_vertical"/>
    <Button android:text="Botón 1.4" />

    <Button android:text="Botón 2.1"
            android:layout_columnSpan="2“
            android:layout_gravity="fill_horizontal"/>
    <Button android:text="Botón 2.4" />

</GridLayout>
```



Es posible expandir las vistas para que ocupen toda la celda

Ejemplo de GridLayout

```
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:columnCount="4"
    android:orientation="horizontal" >

    <Button android:text="Botón 1.1" />
    <Button android:text="Botón 1.2" />
    <Button android:text="Botón 1.3"
            android:layout_rowSpan="2"
            android:layout_gravity="fill_vertical"/>
    <Button android:text="Botón 1.4" />

    <Button android:text="Botón 2.1"
            android:layout_columnSpan="2“
            android:layout_gravity="fill_horizontal"/>
    <Button android:text="Botón 2.4" />

</GridLayout>
```

¿Cuál es la razón para que el botón 1.3 se expanda al total del dispositivo?

Ejemplo de GridLayout

```
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:columnCount="4"
    android:orientation="horizontal" >

    <Button android:text="Botón 1.1" />
    <Button android:text="Botón 1.2" />
    <Button android:text="Botón 1.3"
            android:layout_rowSpan="2"
            android:layout_gravity="fill_vertical"/>
    <Button android:text="Botón 1.4" />

    <Button android:text="Botón 2.1"
            android:layout_columnSpan="2“
            android:layout_gravity="fill_horizontal"/>
    <Button android:text="Botón 2.4" />

</GridLayout>
```

Ejemplo de GridLayout

```
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height=“wrap_content"
    android:columnCount="4"
    android:orientation="horizontal" >

    <Button android:text="Botón 1.1" />
    <Button android:text="Botón 1.2" />
    <Button android:text="Botón 1.3"
            android:layout_rowSpan="2"
            android:layout_gravity="fill_vertical"/>
    <Button android:text="Botón 1.4" />

    <Button android:text="Botón 2.1"
            android:layout_columnSpan="2“
            android:layout_gravity="fill_horizontal"/>
    <Button android:text="Botón 2.4" />

</GridLayout>
```

Cambiar el valor a wrap_content y probar en emulador