

1 RelativeLayout

1.1 Sumario

- 1 Introducción
- 2 Casos prácticos
 - ♦ 2.1 Vistas sen referencias de posición
 - ♦ 2.2 Vistas posicionadas en relación a outras vistas
 - ♦ 2.3 Vistas posicionadas en relación ao pai e a outras vistas

1.2 Introducción

- Tamén é outro dos layouts máis comúns.
- As vistas (views) poden **posicionarse** en relación:
 - ♦ a **outras views** (o que está arriba, abaixo, a dereita,...). Para iso vai facer falla usar **IDs**.
 - ◊ android:layout_above
 - ◊ android:layout_below
 - ◊ android:layout_toLeftOf
 - ◊ android:layout_toRightOf
 - ◊ android:layout_alignLeft
 - ◊ android:layout_alignRight
 - ◊ android:layout_alignTop
 - ◊ android:layout_alignBottom
 - ◊ android:layout_alignBaseline
 - ♦ ao seu **contedor ou pai**, que é un layout (arriba, centrado, abaixo, ...).
 - ◊ android:layout_alignParentLeft
 - ◊ android:layout_alignParentRight
 - ◊ android:layout_alignParentTop
 - ◊ android:layout_alignParentBottom
 - ◊ android:layout_centerHorizontal
 - ◊ android:layout_centerVertical
 - ◊ android:layout_centerInParent



- É un layout moi potente, que facilita o deseño e fai sinxelo entender o arquivo xml.
- Como norma, se un layout ten varios LinearLayout anidados, é necesario substituílos por un RelativeLayout.
- Como se dixo, o posicionamento dun View faise en base a o seu contedor pai ou en base a outro View referenciado polo seu id.

- Se non se referencia a posición, por defecto, todos as views fillas son postas na parte esquerda superior do seu contedor.

Pódense consultar todas as opcións de posicionamento en:

- Referencias:
 - ♦ RelativeLayout: <http://developer.android.com/guide/topics/ui/layout/relative.html>
 - ♦ Opcións de posicionamento: <http://developer.android.com/reference/android/widget/RelativeLayout.LayoutParams.html>

1.3 Casos prácticos

- Comezamos creando un novo proxecto: **U2_05_Relative**

1.3.1 Vistas sen referencias de posición

- No seguinte xml as vistas (botóns neste caso) non teñen unha referencia de posición, polo tanto, todas van estar ubicadas na posición esquerda-superior.

```
<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" >

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Preme 1"/>

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Preme 2" />

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Preme 3" />

</RelativeLayout>
```



1.3.2 Vistas posicionadas en relación a outras vistas

- Para iso precisamos crear un identificador (ID) para a vista/s que se van usar como referencia.



- E logo na vista que se desexa ubicar en relación a outra vista facer uso dese ID.



- A continuación o contido XML do layout:

```

<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" >

    <Button
        android:id="@+id/btBoton1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Preme 1"/>

    <Button
        android:layout_toRightOf="@id/btBoton1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Preme 2" />

    <Button
        android:layout_below="@id/btBoton1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Preme 3" />

</RelativeLayout>

```

- A imaxe amosa a representación gráfica do layout.



1.3.3 Vistas posicionadas en relación ao pai e a outras vistas

```
<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" >

    <Button
        android:id="@+id/btBoton1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Preme 1" />

    <Button
        android:layout_toRightOf="@id/btBoton1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Preme 2" />

    <Button
        android:layout_below="@id/btBoton1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Preme 3" />

    <Button
        android:id="@+id/btBoton4"
        android:layout_centerInParent="true"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Preme 4" />
```

```
<Button
    android:layout_alignBaseline="@id/btBoton4"
    android:layout_alignParentRight="true"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Preme 5" />

</RelativeLayout>
```

- Observar os botóns **Preme 4** e **Preme 5** como teñen algunha referencia ao Pai para ser ubicados.

