

Curso de Programación de Aplicaciones para Android con Kotlin

 jotajotavm.com/android

DISEÑO RESPONSIVE

ADVERTENCIA: Aunque el concepto de diseño responsive lo estamos considerando con objetos Views, puede ser aplicado a objetos que hereden de Views o ViewGroups. NO es exclusivo de un único tipo de objetos, pero no son admitidos en todos los tipos de Layouts.

Si queremos que nuestros LinearLayouts se adapten al tamaño de la pantalla ocupando un ancho relativo en lugar de una medida específica, debemos hacer los siguientes pasos:

NOTA: Estos diseños están concebidos cuando la orientación es horizontal.

OPCION 1

1- Asignamos al Layout Padre un valor total del ancho que tendrá, a esto se le llama "peso", o "weight" sin indicar una unidad específica
`android:weightSum="3"`

2- Asignamos a cada hijo la cantidad que ocupará del ancho del padre
`android:layout_weight="1"`
y le ponemos el width a 0, ya que al ser una propiedad obligatoria, tenemos que darle algún valor



En este ejemplo el padre tiene un `weightSum 3` y cada hijo un `weight 1`. Como vemos se adaptan a cada anchura de pantalla



layout.xml

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="300dp"
    android:background="@color/black"
    android:weightSum="3"
    android:orientation="horizontal">

    <View
        android:layout_width="0dp"
        android:layout_height="300dp"
        android:layout_weight="1"
        android:background="@color/blue_dark"/>





    <View
        android:layout_width="0dp"
        android:layout_height="300dp"
        android:layout_weight="1"
        android:background="@color/blue_light"/>

    <View
        android:layout_width="0dp"
        android:layout_height="300dp"
        android:layout_weight="1"
        android:background="@color/blue_dark"/>

</LinearLayout>
```

Código disponible para copiar/pegar desde el curso



 jotajotavm
 @JoseCodFacilito
 jotajotavm
 jotajotavm

OPCION 2

En el elemento en cuestión que queramos darle un ancho adaptable, le asignamos una anchura de 0 y un peso con valor entre 0 y 1



En este ejemplo los elementos azules y naranja tienen `android:layout_width="0dp"`
El elemento azul tiene
`android:layout_weight="0.7"`
o también `android:layout_weight=".70"`

El elemento naranja tiene
`android:layout_weight="0.3"`
Como vemos mantienen esa proporción en las pantallas según su anchura

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="300dp"
    android:background="@color/black"
    android:orientation="horizontal">

    <View
        android:layout_width="0dp"
        android:layout_height="300dp"
        android:layout_weight="0.7"
        android:background="@color/blue_light"/>

    <View
        android:layout_width="0dp"
        android:layout_height="300dp"
        android:layout_weight="0.3"
        android:background="@color/orange"/>

</LinearLayout>
```