

ProgressBars

ProgressBar

SeekBar

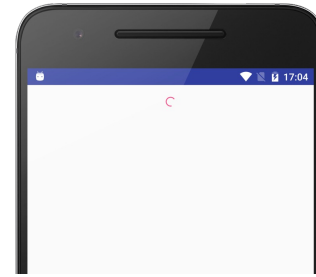
RatingBar

ProgressBar

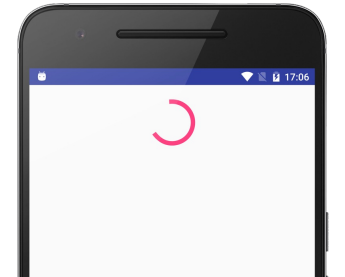
Definición en Layout

```
<ProgressBar android:id="@+id/pb"  
    android:indeterminate="true"  
    style="@android:style/Widget.Material.ProgressBar.Small"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"/>
```

@android:style/Widget.Material.ProgressBar.Small



@android:style/Widget.Material.ProgressBar.Large



```
<ProgressBar android:id="@+id/pb"  
    android:indeterminate="false"  
    android:max="100"  
    android:progress="20"  
    style="@android:style/Widget.Material.ProgressBar.Horizontal"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"/>
```

max valor máximo
progress valor actual

@android:style/Widget.Material.ProgressBar.Horizontal



Métodos

ProgressBar

setMax(max: Int)

getMax() : Int

Establece (o recupera) el valor entero que representa el valor máximo del progressBar.

setProgress(progress: Int)

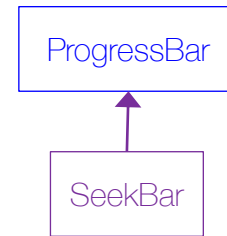
getProgress() : Int

Establece (o recupera) el valor entero que representa el progreso actual del progressBar. Su valor nunca puede ser superior al fijado a través de max

SeekBar

Definición en Layout

```
<SeekBar android:id="@+id/sb"  
    android:max="100"  
    android:progress="20"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"/>
```



SeekBar

Métodos

setMax(max: Int)

getMax() : Int

Establece (o recupera) el valor entero que representa el valor máximo del progressBar.

setProgress(progress: Int)

getProgress() : Int

Establece (o recupera) el valor entero que representa el progreso actual del progressBar. Su valor nunca puede ser superior al fijado a través de max

setOnSeekBarChangeListener(listener : SeekBar.OnSeekBarChangeListener)

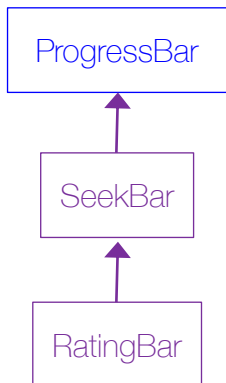
Establece un listener que se ejecutará cuando se modifique el progress del SeekBar

```
val sb : SeekBar = findViewById<SeekBar>(R.id.sb)
sb.setOnSeekBarChangeListener( object : SeekBar.OnSeekBarChangeListener {
    override fun onProgressChanged(seekBar: SeekBar, progress: Int, fromUser: Boolean) : Unit
    {
        seekBar: vista modificada
        progress: nuevo valor actual
        fromUser: indica si la modificación se ha producido por la interacción del usuario con el SeekBar (false indicaría que se ha
                    modificado por programación)
    }
    override fun onStartTrackingTouch(seekBar: SeekBar) : Unit { /* El usuario ha comenzado a interactuar con el SeekBar */ }
    override fun onStopTrackingTouch(seekBar: SeekBar) : Unit { /* El usuario ha finalizado la interacción con el SeekBar*/ }
})
}
```

RatingBar

Definición en Layout

```
<RatingBar android:id="@+id/rating_bar"
    android:isIndicator="false"
    android:numStarts="5"
    android:stepSize="0.5"
    android:rating="3.5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"/>
```



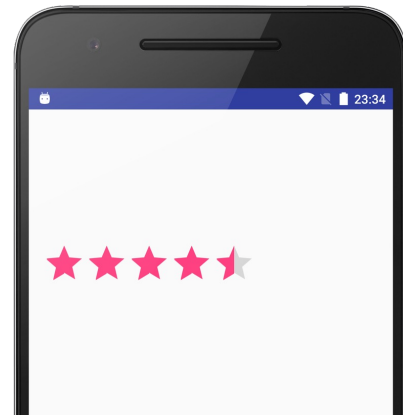
isIndicator Si el valor del RatingBar es sólo un indicador (true) o puede ser modificado por la interacción del usuario (false)

numStarts número de estrellas a mostrar (obligatoriamente layout_width debe ser wrap_content)

stepSize determina la granularidad de cada estrella. Podría decirse que:

$$\text{"max"} = \frac{\text{android:numStarts}}{\text{android:stepSize}}$$

rating número de starts seleccionadas



RatingBar

Métodos

setIsIndicator(isIndicator : boolean)

isIndicator() : Boolean

Establece (o recupera) el valor que determina si el RatingBar debe utilizarse sólo como indicador.

setNumStars(numStars: Int)

getNumStars() : Int

Establece (o recupera) el valor que determina el número de estrellas que aparecerán en la vista

setStepSize(stepSize: Float)

getStepSize() : Float

Establece (o recupera) la porción de estrella que puede ser seleccionable.

setRating(rating: Float)

getRating() : Float

Establece (o recupera) el número de estrellas que hay seleccionadas.

setOnRatingBarChangeListener(listener : RatingBar.OnRatingBarChangeListener)

Establece un listener que se ejecutará cuando se modifique el rating

```
val rb : RatingBar = findViewById<RatingBar>(R.id.rb)
rb.setOnRatingBarChangeListener(object : RatingBar.OnRatingBarChangeListener {
    override fun onRatingChanged (ratingBar: RatingBar, rating: Float, fromUser: Boolean) {
        ratingBar: vista modificada
        rating: nuevo valor de rating
        fromUser: indica si la modificación se ha producido por la interacción del usuario con el RatingBar (false indicaría que se ha modificado por programación)
    }
})
```