

Android Developer Fundamentals

Hola Mundo

Lección 1



1.3 Texto y vistas de desplazamiento

Contenidos

- TextView
- ScrollView

TextView

TextView para texto

- TextView es una vista para mostrar texto de una sola línea o de varias líneas
- EditText es una subclase de TextView con texto editable
- Controlado con atributos de diseño
- Establezca el texto de forma estática desde un recurso de cadena en XML, o dinámicamente desde código Java

Formato de texto en el recurso de cadena

- Use las etiquetas HTML `` y `<i>` para negrita y cursiva
- Todas las demás etiquetas HTML son ignoradas
- Recursos de cadena: una línea continua = un párrafo
- `\n` comienza una nueva línea o párrafo
- Escape los apóstrofes y comillas con barra invertida (`\'`, `\"`)
- Escape cualquier carácter no ASCII con barra invertida (`\`)

Creando TextView en XML

```
<TextView android:id="@+id/textview"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="@string/my_story"/>
```

Atributos comunes de TextView

[android:text](#)—texto para mostrar

[android:textColor](#)—color del texto

[android:textAppearance](#)—estilo o tema predefinido

[android:textSize](#)—tamaño del texto en sp

[android:textStyle](#)—normal, negrita, cursiva o negrita|cursiva

[android:typeface](#)—normal, sans, serif o monospace

[android:lineSpacingExtra](#)—espacio extra entre líneas en sp

Formato de enlaces web activos

```
<string name="article_text">... www.rockument.com
...</string>
```

```
<TextView
    android:id="@+id/article"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:autoLink="web"
    android:text="@string/article_text"/>
```

autoLink values:"web", "email", "phone", "map",
"all"

No utilice HTML
para un enlace
web en texto sin
formato

Creación de TextView en código Java

```
TextView myTextview = new TextView(this);  
myTextView.setWidth(LayoutParams.MATCH_PARENT);  
myTextView.setHeight(LayoutParams.WRAP_CONTENT);  
myTextView.setMinLines(3);  
myTextView.setText(R.string.my_story);  
myTextView.append(userComment);
```

ScrollView

¿Qué pasa con grandes cantidades de texto?

- El usuario puede necesitar desplazarse.
 - Noticias, artículos, ...
- Para permitir a los usuarios desplazar un TextView, incrustarlo en un ScrollView
- Otras vistas se pueden incrustar en un ScrollView
 - LinearLayout, TextView, Button, ...

ScrollView para el contenido de desplazamiento

- ScrollView es una subclase de FrameLayout
- Solo puede mantener una vista (que puede ser un ViewGroup)
- Mantiene todo el contenido en la memoria
- No es bueno para textos largos, diseños complejos
- No anidar múltiples vistas de desplazamiento
- Utilice HorizontalScrollView para desplazamiento horizontal
- Use un RecyclerView para las listas

Diseño ScrollView con un TextView

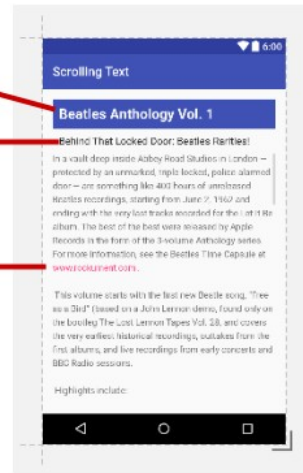
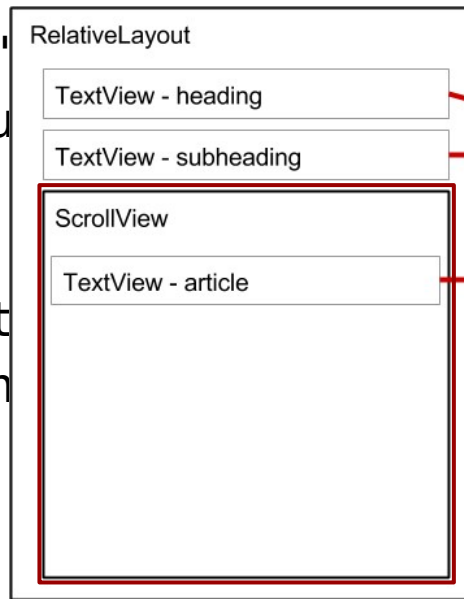
<ScrollView

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/article_su
```

<TextView

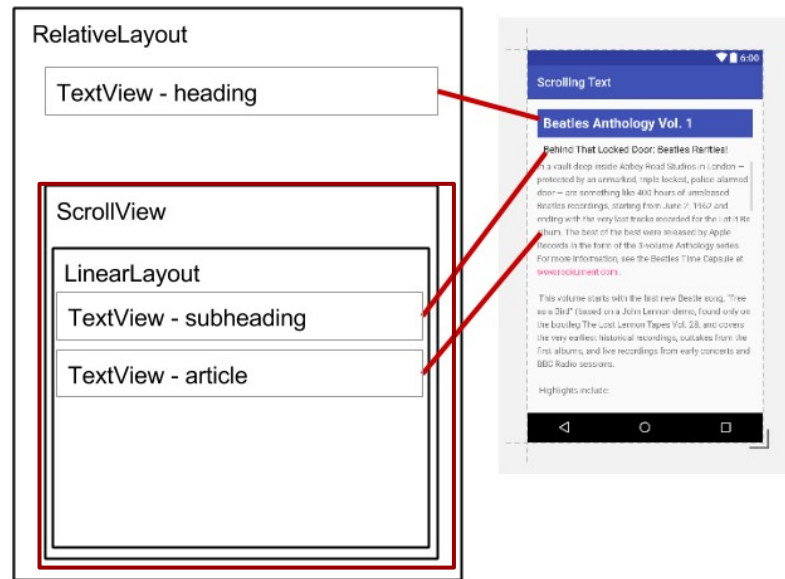
```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    .../>
```

</ScrollView>



Diseño ScrollView con un grupo de vista

```
<ScrollView ...  
  <LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:orientation="vertical">  
  
    <TextView  
      android:id="@+id/article_subheading"  
      .../>  
  
    <TextView  
      android:id="@+id/article" ... />  
  </LinearLayout>  
</ScrollView>
```



ScrollView con imagen y botón

```
<ScrollView...>
```

```
  <LinearLayout...>
```

Un hijo de ScrollView que
puede ser un diseño

```
    <ImageView.../>
```

```
    <Button.../>
```

```
    <TextView.../>
```

Hijos de LinearLayout

```
  </LinearLayout>
```

```
</ScrollView>
```


FIN