


Open Opened 3 days ago by  **Rubén Montero**

Celda para nuestro RecyclerView (Java)

Resumen

- Añadiremos un nuevo archivo `ClipViewHolder.java`

Descripción

Estamos acostumbrados a escribir una clase Java cada vez que añadimos un fichero de interfaz XML.

- Cuando creamos una actividad, el XML lleva asociado una `Activity` en Java
- Cuando creamos un *fragment*, el XML lleva asociado un `Fragment` en Java

En esta ocasión, crearemos un `ViewHolder`. O sea, la clase Java para cada *celda* de nuestro `RecyclerView`.

La tarea

En la jerarquía de archivos, desde `java > com.afundacion.fp.clips` **haz click** derecho > `New > Java Class`. **Dale** el nombre `ClipViewHolder`:

```
package com.afundacion.fp.clips;

public class ClipViewHolder {
}
```

Como primer paso **indica** que `ClipViewHolder` *hereda* de `ViewHolder`. Concretamente, de `RecyclerView.ViewHolder`. **Así**:

```
package com.afundacion.fp.clips;

import androidx.recyclerview.widget.RecyclerView;

public class ClipViewHolder extends RecyclerView.ViewHolder {
}
```

Verás que aparece el error:

```
There is no default constructor available in 'androidx.recyclerview.widget.RecyclerView.ViewHolder'
```

Esto se debe a que no está permitido *instanciar* una subclase de `ViewHolder` *directamente*. Es decir, por ser una subclase de `ViewHolder`, no podríamos hacer `v = new ClipViewHolder();`. ¡El constructor *vacío* ya no existe!

Estamos, por ello, obligados a crear un *constructor* que recibe una *vista* (`View`).

Usa la ayuda de Android Studio (bombillita roja) y **selecciona** *Create constructor matching super* para llegar a:

```
import android.view.View;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

public class ClipViewHolder extends RecyclerView.ViewHolder {
    public ClipViewHolder(@NonNull View itemView) {
        super(itemView);
    }
}
```

Luego, **añade** a `ClipViewHolder.java` un atributo privado de tipo `TextView`, llamado `textView`. **Inicialízalo** desde el método constructor mediante `findViewById`, como se indica a continuación:

```
public class ClipViewHolder extends RecyclerView.ViewHolder {
    private TextView textView;
```

```
public ClipViewHolder(@NonNull View itemView) {  
    super(itemView);  
    textView = itemView.findViewById( /* EL ID que le diste a tu TextView en recycler_view_cell.xml */);  
}  
}
```

(Como puedes ver, estamos invocando `findViewById` sobre `itemView`. Este objeto será creado dentro del adaptador en la siguiente tarea, donde uniremos todas las piezas).

Por último

Sube tus cambios al repositorio en un nuevo *commit*.



[Rubén Montero @ruben.montero](#) changed milestone to [%Sprint 2](#) 3 days ago