

## RecyclerView - Paso a Paso

### Parte 1 - Dependencias

1. Definir la **dependencia** en el archivo **build del Gradle**:

```
dependencies {  
    implementation 'com.android.support:recyclerview-v7:27.1.1'  
}
```

### Parte 2 - Layout

1. Agregar el **RecyclerView** al **XML**.
2. Crear un nuevo **layout para la celda** que se va a mostrar el ítem en la lista.

### Parte 3 - Clase

1. Implementar una clase que **modele el objeto** que mostraremos en nuestro RecyclerView.

### Parte 4 - Adapter

1. Crear una clase Adapter que **extienda de RecyclerView.Adapter** y **sobreescribir los métodos** necesarios.
2. Dentro del Adapter crear una clase privada para el **ViewHolder** que **extienda de RecyclerView.ViewHolder**.
3. En el ViewHolder **sobreescribir el constructor**.
4. Agregar los **componentes del layout de la celda** como **atributos** del ViewHolder, y en el constructor **inicializarlos con findViewById a través de la vista** que recibe.
5. En el ViewHolder **crear un método** para realizar el **bind** (cargar la información dentro de los componentes del layout de la celda) que **reciba los datos necesarios y los setee**. Ejemplo:

```
public void bindTitular (Titular titular){  
    textViewTitulo.setText(titular.getTitulo());  
}
```

6. En el **Adapter** agregar los atributos necesarios: **lista**. Crear el **constructor** correspondiente.
7. Programar los **métodos** que sobrescribimos:
  - a. **getItemCount()**: Devuelve el tamaño de la lista.
  - b. **onCreateViewHolder(...)**: Esta función es la encargada de inflar la vista del detalle de la lista y además de crear un objeto ViewHolder utilizando esa vista.
    - i. En este método vamos a **buscar el inflador**.
    - ii. **Inflar la vista**.
    - iii. **Instanciar un ViewHolder** del tipo que creamos anteriormente pasándole como parámetro la **vista** que inflamos.
    - iv. Por último, **retornamos el ViewHolder**.
  - c. **onBindViewHolder(..)**: Esta función es la encargada de actualizar los datos reutilizando el ViewHolder que viene por parámetro (holder). Es decir, carga el ViewHolder con los datos del objeto que se encuentra en la posición que me pasan por parámetro (position).
    - i. Buscamos en la lista el **ítem a través de la posición** que recibe por parámetro.
    - ii. **Castear el holder** que recibe por parámetro al tipo ViewHolder que creamos.
    - iii. Pedirle al **ViewHolder que haga el bind** del ítem.

## Parte 5 - Activity

1. Declarar y **referenciar el RecyclerView** que definimos en el XML.
2. **Setear el LayoutManager al RecyclerView** (recordar que podemos definir distintos tipos de LayoutManager según la forma en la cual queremos mostrar nuestra lista.)
3. Setear al RecyclerView setHasFixedSize en true, si sabemos que tanto el ancho como el alto del recyclerView no se modificarán.
4. Crear un **Adapter**, utilizando la clase definida anteriormente.
5. **Setear al RecyclerView el adapter** creado.