

INDICE

<u>1.1.</u>	<u>INTRODUCCIÓN</u>	<u>1</u>
<u>1.2.</u>	<u>CASO PRÁCTICO</u>	<u>2</u>
<u>1.2.1.</u>	<u>CÓDIGO DEL LAYOUT</u>	<u>3</u>
<u>1.2.2.</u>	<u>CÓDIGO JAVA DE LA APLICACIÓN</u>	<u>3</u>

1.1. Introducción

- ✓ Un ListView es una view que le permite al usuario seleccionar un elemento.
- ✓ A diferencia de un Spinner una lista muestra todos sus elementos en la pantalla.



- ✓ La imagen recoge un ejemplo de lista, cada ítem puede contener un solo dato, o múltiples datos. El primer caso es el que vamos a ver.

ListView



La imagen muestra una lista simple



Un ListView más complejo es la lista de contactos del teléfono. Cada ítem tiene como mínimo: Foto, Nombre, Tfno, si tiene whatsapp, etc.

- ✓ Al igual que en el Spinner usaremos una fuente de datos (array estático, array dinámico, recurso xml, etc) para enlazarlo con el adaptador y este con el ListView.
- ✓ También podemos hacer uso de layouts 'predefinidos'.
 - Los layout que se pueden emplear se pueden consultar en:
http://wiki.cifprodolfoucha.es/index.php?title=Spinner#Caso_pr.C3.A1ctico
 - El código fuente de cada layout se puede consultar en https://github.com/aosp-mirror/platform_frameworks_base/tree/master/core/res/res/layout
- ✓ Todo lo visto con Spinner en el uso de adaptadores se puede emplear en este view teniendo en cuenta estas diferencias:
 - Los layout 'predefinidos' son de tipo listview, no spinner.
 - La gestión de eventos sobre un elemento de la lista se hace con la interface OnItemClickListener.
- ✓ Si queremos que al pulsar sobre la lista, visualmente aparezca seleccionado el elemento (**visualmente hablando, no queda seleccionado a nivel de programación**) podemos cambiar estas propiedades en el diseñador de Layout sobre el ListView:

```
android:choiceMode="singleChoice"  
android:listSelector="@color/colorPrimary"
```

Nota: @color/colorPrimary es un recurso de tipo 'color' con un color definido.

Nota Importante: En la actualidad se recomienda emplear el RecyclerView en vez del ListView, ya que consume mucha menos memoria.

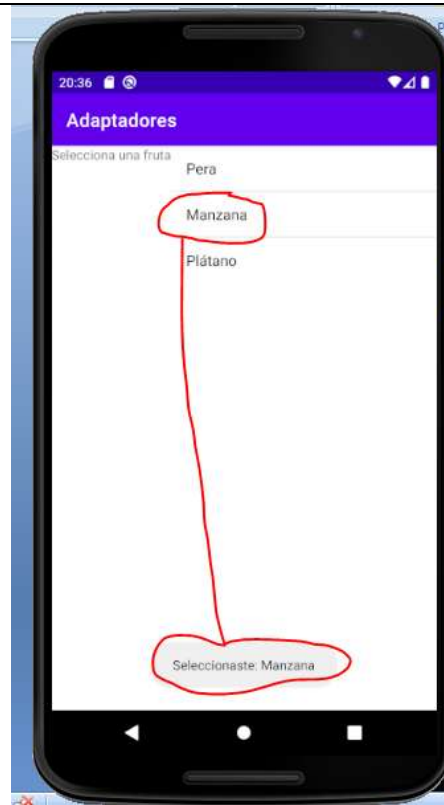
Un ListView va a 'generar' todos los componentes gráficos que conforman cada fila para todos los elementos de la lista, mientras que el RecyclerView solamente genera aquellos componentes que se visualizan.

1.2. Caso Práctico

- ✓ Dentro del paquete **Adaptadores** crear una nueva 'Empty Activity' de nombre: **UD04_01_ListView** de tipo Launcher. Modifica el archivo **AndroidManifest.xml** y añade una label a la activity.



Ejemplo de una lista simple, al pulsar en un ítem...



Nos indica que ítem fue seleccionado con un Toast.

1.2.1.Código del Layout

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".UD04_01_ListView">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Selecciona una fruta" />

    <ListView
        android:id="@+id/lv_frutas"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

</LinearLayout>
```

1.2.2.Código Java de la aplicación

```
package com.example.adaptadores;

import androidx.appcompat.app.AppCompatActivity;
```

```

import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.TextView;
import android.widget.Toast;

public class UD04_01_ListView extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_u_d04_01_list_view);
        ListView lvFrutas = (ListView) findViewById(R.id.lv_frutas);

        //Fuente de datos
        String[] frutas = new String[] { "Pera", "Manzana", "Plátano" };

        //Enlace del adaptador con la fuente de datos
        ArrayAdapter<String> adaptador = new ArrayAdapter<String>(this, android.R.layout.simple_list_item_1, frutas);

        //Enlace del adaptador con ListView
        lvFrutas.setAdapter(adaptador);

        //Escuchador
        lvFrutas.setOnItemClickListener(new AdapterView.OnItemClickListener() {

            @Override
            public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
                // TODO Auto-generated method stub
                //Toast.makeText(getBaseContext(), "Seleccionastes: " + parent.getItemAtPosition(position), Toast.LENGTH_SHORT).show();
                Toast.makeText(getBaseContext(), "Seleccionaste: " + ((TextView) view).getText(), Toast.LENGTH_SHORT).show();
            }
        });
    }
}

```