



# Diálogos



# Diálogos

- Diálogos más habituales:
  - Diálogo de Alerta
  - Diálogo de Confirmación
  - Diálogo de Selección
  - Diálogo Personalizado

# Diálogo de Alerta

- Este tipo de diálogo se limita a mostrar un mensaje sencillo al usuario y un único botón de OK para confirmar su lectura.

```
AlertDialog.Builder builder = new AlertDialog.Builder(MainActivity.this);  
builder.setMessage("Esto es un mensaje de alerta.")  
    .setTitle("Información")  
    .setPositiveButton("OK", (dialog, i) -> {  
        dialog.cancel();  
    });  
builder.create().show();
```

Información

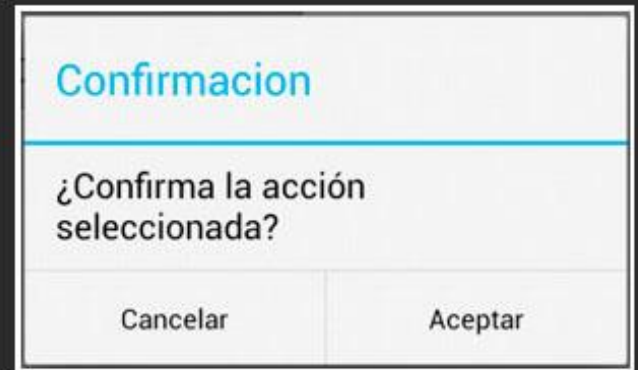
Esto es un mensaje de alerta.

OK

# Diálogo de Confirmación

- Un diálogo de confirmación es muy similar al de alerta, con la diferencia de que lo utilizaremos para solicitar al usuario que nos confirme una determinada acción.

```
AlertDialog.Builder builder = new AlertDialog.Builder(MainActivity.this);
builder.setMessage("¿Confirma la acción seleccionada?")
    .setTitle("Confirmación")
    .setPositiveButton("Aceptar", (dialog, i) → {
        Log.i("Dialogos", "Confirmacion Aceptada");
        dialog.cancel();
    })
    .setNegativeButton("Cancelar", (dialogInterface, i) → {
        Log.i("Dialogos", "Confirmacion Cancelada");
    });
builder.create().show();
```



# Diálogo de Selección

- Cuando las opciones a seleccionar por el usuario no son sólo dos, como en los diálogos de confirmación, sino que el conjunto es mayor, podemos utilizar los diálogos de selección para mostrar una lista de opciones entre las que el usuario pueda elegir.

```
AlertDialog.Builder builder = new AlertDialog.Builder(DialogActivity.this);
builder.setTitle("Selección")
    .setItems(items, new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int position) {
            Log.i("Dialogos", "Opción elegida: " + items[position]);
        }
    });
builder.create().show();
```

## Selección

Español

Inglés

Francés

# Diálogo de Selección(RadioButton)

- El diálogo de selección con radioButtons permite recordar la opción seleccionada por el usuario para que aparezcan marcadas al visualizar de nuevo el cuadro de diálogo

```
AlertDialog.Builder builder = new AlertDialog.Builder(DialogActivity.this);
builder.setTitle("Selección")
    .setSingleChoiceItems(items, -1, new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int position) {
            Log.i("Dialogos", "Opción elegida: " + items[position]);
        }
    });
builder.create().show();
```

## Selección

Español

☐

Inglés

☒

Francés

☐



# Diálogo de Selección(CheckBox)

- El diálogo de selección con checkBox permite es similar al diálogo con radioButtons permitiendo selección múltiple.

```
final String[] items = {"Español", "Inglés", "Francés"};
final boolean[] answers = {false, false, false};

AlertDialog.Builder builder = new AlertDialog.Builder(DialogActivity.this);
builder.setTitle("Selección")
    .setMultiChoiceItems(items, answers, onClick(dialogInterface, position, b) → {
        Log.i("Dialogos", "Opción elegida: " + items[position]);
    });
builder.create().show();
```

Selección	
Español	<input checked="" type="checkbox"/>
Inglés	<input checked="" type="checkbox"/>
Francés	<input type="checkbox"/>

# Diálogos Personalizados

- Los diálogos personalizados permiten definir un layout xml para la vista del diálogo.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal"
    android:padding="3dp">

    <ImageView
        android:id="@+id/imageView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@mipmap/ic_launcher" />

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:padding="3dp">

        <TextView
            android:id="@+id/textView1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Dialogo Linea 1" />

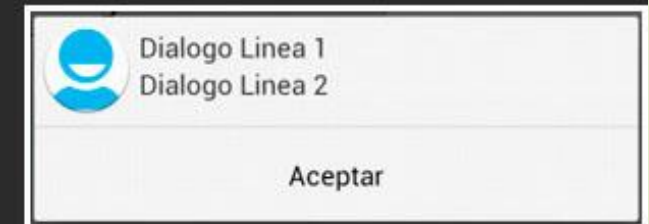
        <TextView
            android:id="@+id/textView2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Dialogo Linea 2" />

    </LinearLayout>
</LinearLayout>
```



# Diálogos Personalizados

```
AlertDialog.Builder builder = new AlertDialog.Builder(MainActivity.this);
LayoutInflater inflater = getLayoutInflater();
builder.setView(inflater.inflate(R.layout.custom_dialog, null))
    .setPositiveButton("Aceptar", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int i) {
            dialog.cancel();
        }
    });
builder.create().show();
```





➡ <https://github.com/rononunsys/Dialogs.git>