|  |  |  |
| --- | --- | --- |
| **Título componente** | **Acordeón** | |
| **Título** | **Eventos *listeners*** | |
| **Texto descriptivo** | Para fijar un *listener,* se debe implementar y agregar los métodos del evento correspondiente a cada *View* mediante el método **setOn**. Para realizar el ejemplo del evento *listener* de un *Button*, se deben seguir algunos pasos: | |
| **Título** | **Texto** | **Imagen de referencia (opcional)** |
| **Crear el *Button* en el *Layout*** | Crear el *Button* en el *Layout* con su propiedad ID, ya que es la que nos va a permitir referenciarlo desde el código Java. Se muestra la creación de un botón en e*l layout* llamado **activity\_main.xml**. El código XML tiene un contenedor de tipo **LinearLayout** con su propiedad vertical y e*l Button* es creado con sus propiedades básicas. La propiedad que se necesita para referenciar es la del **id**. El nombre del identificador que se colocó para el ejemplo fue **btnclic**, y en la vista de diseño se puede ver cómo queda el botón. Estas son las diferentes vistas en las que se puede trabajar en el *layout:*   * ***Code:***permite visualizar solo el código XML. * ***Split:*** permite visualizar el código y el diseño como se ve en la figura. * ***Design:***permite visualizar solo el editor de diseño.  |  | | --- | | <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:orientation="vertical"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:layout\_marginTop="80dp"  tools:context=".MainActivity">  <Button  android:id="@+id/btnclic"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="@string/click" />  </LinearLayout> | | No aplica |
| **Declarar el tipo de *view*** | En la clase Java, declarar el tipo de *view* y referenciarlo. La clase en Java está compuesta por un método principal que se llama **onCreate**. Todo el código que esté dentro de este método se ejecuta inmediatamente se corra la aplicación. Cada clase tiene relacionado su *layout* en el método **setContentView**, por lo tanto, se debe referenciar los *view* en cada clase que lo contenga. En el ejemplo, la clase Java **MainActivity** contiene el **layout activity\_main** que es donde se creó el *Button.*   |  | | --- | | import android.os.Bundle;  import android.widget.Button;  import androidx.appcompat.app.AppCompatActivity;  public class MainActivity extends AppCompatActivity {  Button btnclic; // Declarar el botón  @Override  protected void onCreate(Bundle savedInstanceState) { // Método principal  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);  /\* Para referenciar el Button que está en el layout  se hace por medio del Id y con el método findViewByID \*/  btnclic = findViewById(R.id.btnclic);  /\* En la variable btnclic quedan almacenadas las propiedades del botón  para poder manejar el evento listener \*/  }  } | |  |
| **Implementar el evento *listener*** | Implementar el evento *listener:* En la figura se ve la implementación del evento **setOnClickListener** para que cuando el usuario dé clic en el botón se ejecute una instrucción. El código debe escribirse en el método **onClick**.  Los *View* tiene diferentes métodos escuchadores, en la tabla que se encuentra a continuación se pueden observar los más utilizados:   * ***ListView.***SetOnItemClickListener * ***Spinner.*** SetOnItemSelectedListener. * ***ImageView.*** SetOnClickListener * ***RadioButton.*** SetOnClickListener  |  | | --- | | public class MainActivity extends AppCompatActivity {  Button btnclic; // Declarar el botón  @Override  protected void onCreate(Bundle savedInstanceState) { // Método principal  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);  /\* Para referenciar el Button que está en el layout  se hace por medio del Id y con el método findViewByID \*/  btnclic = findViewById(R.id.btnclic);  /\* En la variable btnclic quedan almacenadas las propiedades del botón  para poder manejar el evento listener \*/  btnclic.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  // Aquí se colocan las instrucciones que se quieran ejecutar al dar clic en el botón  }  });  }  } | |  |