



Actividad M1-02

Hola Mundo

Objetivo: Ejecutar una sentencia básica de programación en lenguaje Java

**Carlos Alberto Ramírez
Sánchez**

No cuenta: 303044651

PROGRAMACIÓN DE DISPOSITIVOS MÓVILES

Agosto 2025



Objetivo del aprendizaje

Ejecutar en lenguaje Java una sentencia básica y comprobar su impresión en Logcat ("Hello World"), así como mostrar un texto en la interfaz usando vistas XML (sin Jetpack Compose).

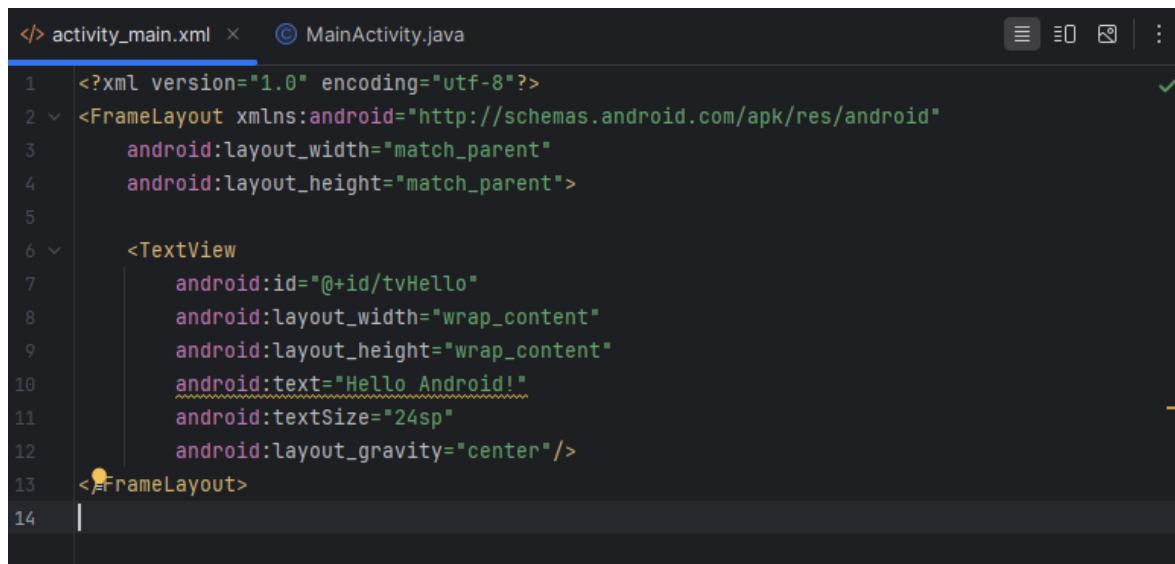
Ambiente y configuración

- **Tipo de proyecto:** Empty Views Activity (no Compose).
- **Lenguaje:** Java.
- **Mínimo SDK:** API 24 (Android 7.0) o superior.
- **Diseño (XML):** activity_main.xml con un TextView centrado.
- **Actividad principal:** MainActivity.java.

Desarrollo

Estructura del layout (activity_main.xml)

Se definió un diseño minimalista con un FrameLayout y un TextView centrado que muestra el texto de prueba en pantalla.



```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:id="@+id/tvHello"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello Android!"
        android:textSize="24sp"
        android:layout_gravity="center"/>
</FrameLayout>
```

Activity_main.xml (interfaz)

- Se declara un **FrameLayout** como contenedor raíz que ocupa todo el ancho y alto (match_parent).
- Dentro, se coloca un **TextView** con:

- **@+id/tvHello**: identificador del control.
- **layout_width/layout_height="wrap_content"**: el tamaño se ajusta al texto.
- **android:text="Hello Android!"**: texto que se muestra en pantalla.
- **android:textSize="24sp"**: tamaño de letra accesible (en sp).
- **android:layout_gravity="center"**: se centra el TextView dentro del FrameLayout.

Resultado: al ejecutar la app, en la pantalla del emulador se visualiza el texto “Hello Android!”.

Lógica de la actividad (MainActivity.java)

Se implementó la actividad principal derivando de AppCompatActivity, se infló el layout y se agregó una impresión en Logcat usando Log.d(...) para la evidencia requerida.

Código de referencia

```

</> activity_main.xml      © MainActivity.java ×
1 package com.example.holamundologjava;
2
3 import android.os.Bundle;
4 import android.util.Log;
5 import androidx.appcompat.app.AppCompatActivity;
6
7 public class MainActivity extends AppCompatActivity {
8     private static final String TAG = "MiPrimerAppCarlos";
9
10    @Override
11    protected void onCreate(Bundle savedInstanceState) {
12        super.onCreate(savedInstanceState);
13        setContentView(R.layout.activity_main); // Carga el layout XML
14
15        // Imprime en Logcat (evidencia)
16        Log.d(TAG, msg: "Hello World");
17    }
18}

```

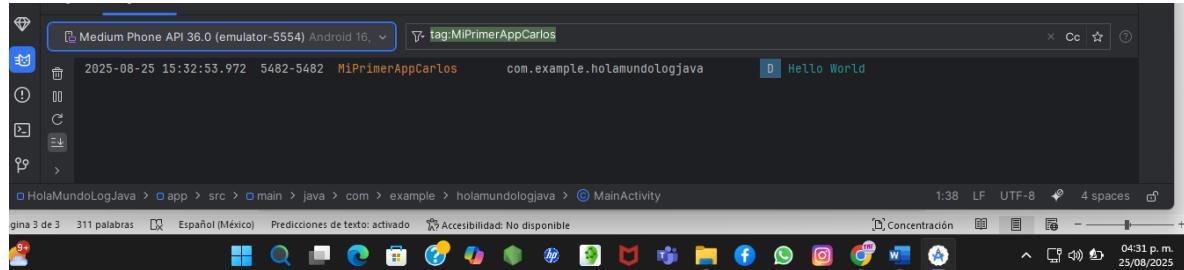
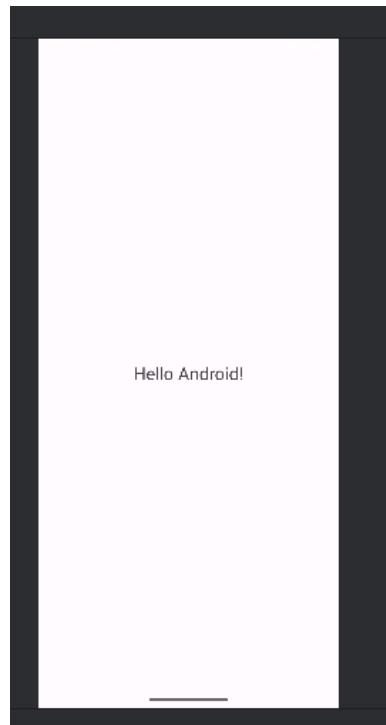
MainActivity.java (lógica)

- Se ubica en el paquete com.example.holamundologjava y extiende AppCompatActivity, por lo que funciona como la actividad principal con compatibilidad de soporte.
- Se importan clases básicas de Android (ciclo de vida, logs y compat).

- Se define una constante TAG con el valor "MiPrimerAppCarlos" para etiquetar mensajes en Logcat.
- En onCreate(Bundle savedInstanceState) (método llamado al crear la Activity):
 - Se invoca setContentView(R.layout.activity_main) para inflar y mostrar el layout XML definido en activity_main.xml.
 - Se ejecuta Log.d(TAG, "Hello World"); para imprimir en Logcat (nivel Debug) el mensaje "Hello World" con la etiqueta definida.

Resultado: la Activity carga la interfaz XML y, además, deja evidencia en Logcat con la línea D/MiPrimerAppCarlos: Hello World.

Resultados



Conclusión

Se creó un proyecto Android en Java utilizando vistas XML (sin Jetpack Compose). Se comprobó la ejecución de una sentencia básica de salida mediante Logcat con el mensaje "Hello World" y se mostró texto en pantalla en el emulador. Con ello se validó el flujo básico del ciclo de vida de una Activity (método `onCreate`), la relación entre código Java y layout XML, y el uso de Logcat como herramienta de depuración.

Fuentes:

- Android Developers. (s. f.). *Log (android.util.Log) — API reference*.
<https://developer.android.com/reference/android/util/Log> Android Developers
- Android Developers. (2024, 3 de enero). *View logs with Logcat*.
<https://developer.android.com/studio/debug/logcat> Android Developers
- Android Developers. (2024, 19 de septiembre). *Run apps on the Android Emulator*.
<https://developer.android.com/studio/run/emulator> Android Developers
- Android Developers. (s. f.). *Jetpack Compose Tutorial*.
<https://developer.android.com/develop/ui/compose/tutorial> Android Developers
- Android Developers. (2024, 3 de enero). *Logcat command-line tool*.
<https://developer.android.com/tools/logcat> Android Developers