



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

Sistema de Universidad Abierta

Licenciatura en informática

Alumno: Alfonso Muciño Espitia

Materia: Programación de Dispositivos Móviles

Asesor: Cristian Cardoso Arellano

23/08/2025

1. Imprimir un *Hello world*.
2. Ingresar el código de programación. para imprimir en el log "Hello World"
3. Buscar la impresión del "hello world" en la ventana "logcat" dentro de Android Studio.
4. Incluir una captura de pantalla en el entregable.
5. Incluir portada, nombre, actividad, fecha, evidencia (capturas de pantalla) y una conclusión.

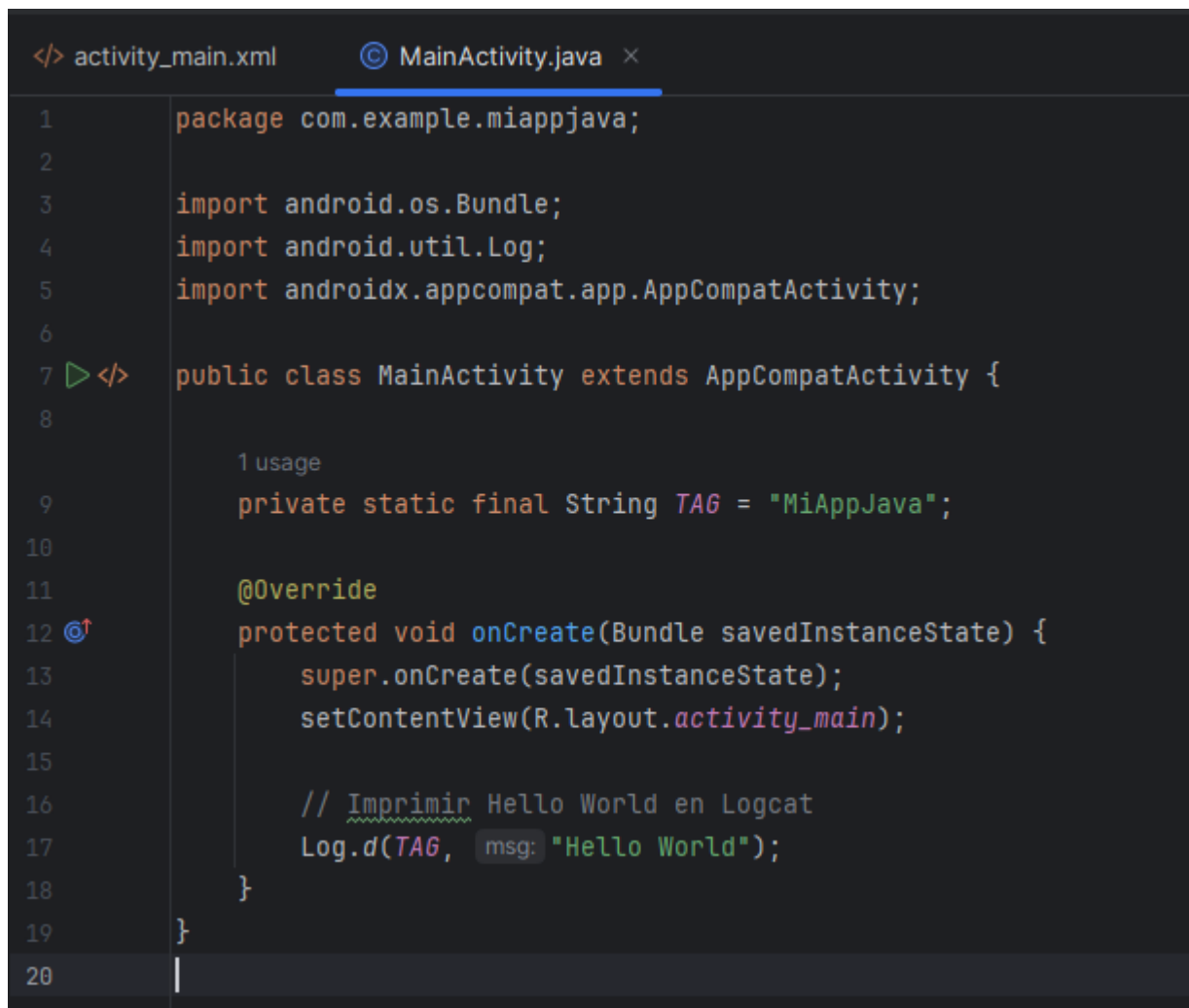
Subir un sólo archivo en formato PDF en el repositorio de GitHub creado por el alumno.

Subir actividad en el repositorio de GitHub creado por el alumno.

1. android_2026_m1_02.pdf
2. código fuente.

Objetivo:

El objetivo de esta actividad es imprimir un mensaje en la consola Logcat utilizando el entorno de desarrollo Android Studio y el lenguaje de programación Java con el fin de familiarizarse con la estructura básica de una aplicación Android, el ciclo de vida de una actividad y el uso de herramientas de depuración esenciales para el desarrollo móvil.



```
</> activity_main.xml    © MainActivity.java x
1      package com.example.miappjava;
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
9      1 usage
10     private static final String TAG = "MiAppJava";
11
12     @Override
13     protected void onCreate(Bundle savedInstanceState) {
14         super.onCreate(savedInstanceState);
15         setContentView(R.layout.activity_main);
16
17         // Imprimir Hello World en Logcat
18         Log.d(TAG, msg: "Hello World");
19     }
20 }
```

Esta imagen muestra el código fuente en Java dentro de la clase MainActivity, donde se utiliza el método Log.d() para imprimir el mensaje "Hello World" en la consola Logcat.

```
</> activity_main.xml x MainActivity.java
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:layout_width="match_parent"
4      android:layout_height="match_parent"
5      android:orientation="vertical"
6      android:gravity="center">
7
8      <TextView
9          android:id="@+id/textHello"
10         android:layout_width="wrap_content"
11         android:layout_height="wrap_content"
12         android:text="Hello World"
13         android:textSize="24sp"
14         android:textColor="#000000"/>
15 </LinearLayout>
16
```

Esta imagen muestra el archivo `activity_main.xml` que define la interfaz gráfica de la aplicación en Android Studio. En este archivo se configura el diseño visual que se verá en el emulador o dispositivo físico. En este caso se ha agregado un `TextView` con el mensaje “Hello World”, lo que permite que dicho texto aparezca directamente en la pantalla del emulador al ejecutar la aplicación.

```
API 16, API 36.0  MiAppJava
6840-6840 CompatChangeReporter com.example.miappjava D Compat change id reported: 377864165; UID 10218; state: ENABLED
6840-6840 DesktopModeFlags com.example.miappjava D Toggle override initialized to: OVERRIDE_UNSET
6840-6840 HWUI com.example.miappjava W Image decoding logging dropped!
6840-6840 ample.miappjava com.example.miappjava I hiddenapi: Accessing hidden method Landroid/view/ViewGroup;->mak
6840-6840 MiAppJava com.example.miappjava D Hello World
```

En esta imagen se muestra la consola Logcat de Android Studio, donde se imprime el mensaje “Hello World” utilizando el método `Log.d()` desde la clase `MainActivity`. Esta herramienta permite visualizar mensajes de depuración, errores y eventos del sistema durante la ejecución de la aplicación.



Esta imagen muestra el emulador de Android ejecutando la aplicación desarrollada en Android Studio. En la pantalla se visualiza el mensaje “Hello World”, el cual fue definido en el archivo `activity_main.xml` mediante un componente `TextView`.

Conclusión:

Esta actividad permitió familiarizarme con el entorno de Android Studio, la estructura de un proyecto en Java y el uso de Logcat para depurar y verificar mensajes. Es un paso fundamental para comenzar a desarrollar aplicaciones móviles.