



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN



DIVISIÓN SISTEMA UNIVERSIDAD ABIERTA Y EDUCACIÓN A DISTANCIA

PROGRAMACION DE DISPOSITIV.MOVILES

UNIDAD 3

ACTIVIDAD 1

NOMBRE DEL ALUMNO:

PAULINA RODRIGUEZ SAMPEDRO

NOMBRE DEL ASESOR:

CRISTIAN CARDOSO ARELLANO

SEXTO SEMESTRE

GRUPO: 9696

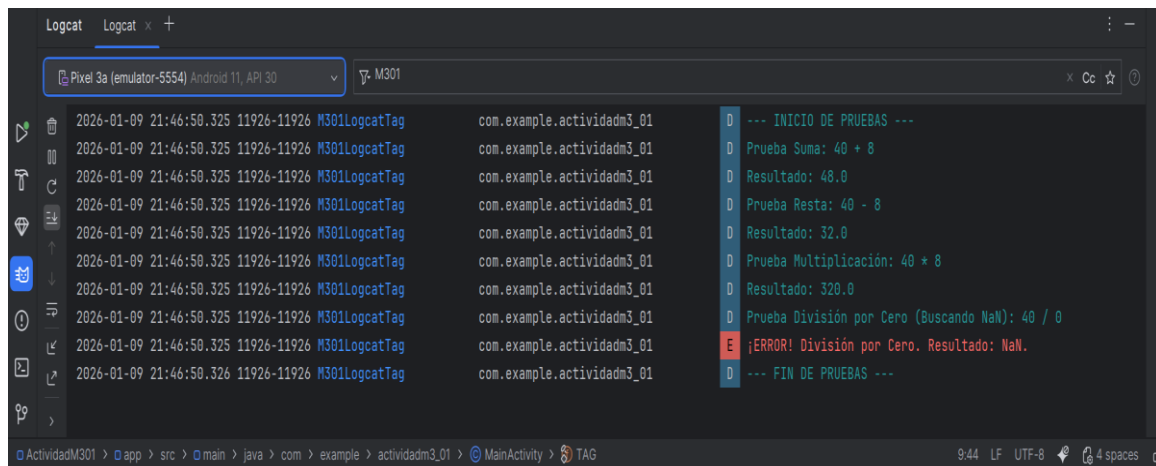
Mi primera calculadora en Android Studio

Objetivo

Comprender y practicar el uso de los operadores básicos del lenguaje Java mediante el desarrollo de una calculadora modularizada, con el fin de fortalecer el dominio de la lógica aritmética y asegurar la robustez del código a través de la validación de estados NaN y el monitoreo de resultados en el Logcat.

Resultados de la actividad

Ilustración 1. Prueba de ejecución de las operaciones básicas y manejo de NaN en el logcat



```
Logcat  Logcat x +
Pixel 3a (emulator-5554) Android 11, API 30  M301
2026-01-09 21:46:50.325 11926-11926 M301LogcatTag com.example.actividadm3_01 D --- INICIO DE PRUEBAS ---
2026-01-09 21:46:50.325 11926-11926 M301LogcatTag com.example.actividadm3_01 D Prueba Suma: 40 + 8
2026-01-09 21:46:50.325 11926-11926 M301LogcatTag com.example.actividadm3_01 D Resultado: 48.0
2026-01-09 21:46:50.325 11926-11926 M301LogcatTag com.example.actividadm3_01 D Prueba Resta: 40 - 8
2026-01-09 21:46:50.325 11926-11926 M301LogcatTag com.example.actividadm3_01 D Resultado: 32.0
2026-01-09 21:46:50.325 11926-11926 M301LogcatTag com.example.actividadm3_01 D Prueba Multiplicación: 40 * 8
2026-01-09 21:46:50.325 11926-11926 M301LogcatTag com.example.actividadm3_01 D Resultado: 320.0
2026-01-09 21:46:50.325 11926-11926 M301LogcatTag com.example.actividadm3_01 D Prueba División por Cero (Buscando NaN): 40 / 0
2026-01-09 21:46:50.325 11926-11926 M301LogcatTag com.example.actividadm3_01 E ¡ERROR! División por Cero. Resultado: NaN.
2026-01-09 21:46:50.326 11926-11926 M301LogcatTag com.example.actividadm3_01 D --- FIN DE PRUEBAS ---
```

En esta captura se observa la ejecución de las operaciones básicas (suma, resta y multiplicación) desplegadas en la salida estándar del Logcat, además al intentar realizar una operación inválida, el programa utiliza la clase `Double.isNaN()` para validar el resultado y desplegar un mensaje de error controlado en la salida estándar.

Conclusión

El desarrollo de esta actividad permitió consolidar el uso de los operadores básicos del lenguaje Java mediante una arquitectura basada en la abstracción y modularización, lo que facilitó la separación de la lógica matemática de la interfaz de salida. Durante el proceso, se validó la correcta implementación de las funciones de suma, resta y multiplicación, logrando un flujo de datos eficiente entre las clases diseñadas. Asimismo, se puso especial énfasis en la función de división, integrando un mecanismo de manejo de error para el valor NaN (Not a Number); esto garantizó que, ante una división por cero, el sistema no presentara comportamientos inesperados, sino que gestionara la excepción de manera controlada. Finalmente, la verificación a través del Logcat confirmó que el código no solo cumple con los cálculos aritméticos requeridos, sino que también presenta una estructura robusta para ser utilizada en otros proyectos.

Referencia

Desarrolladores de Android. (s.f.). *Entorno de desarrollo para aplicaciones Android*.
<https://desarrollador>