


UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN

DIVISIÓN SISTEMA UNIVERSIDAD ABIERTA Y EDUCACIÓN A DISTANCIA

PROGRAMACION DE DISPOSITIV.MOVILES

UNIDAD 5

ACTIVIDAD 2

NOMBRE DEL ALUMNO:

PAULINA RODRIGUEZ SAMPEDRO

NOMBRE DEL ASESOR:

CRISTIAN CARDOSO ARELLANO

SEXTO SEMESTRE

GRUPO: 9696

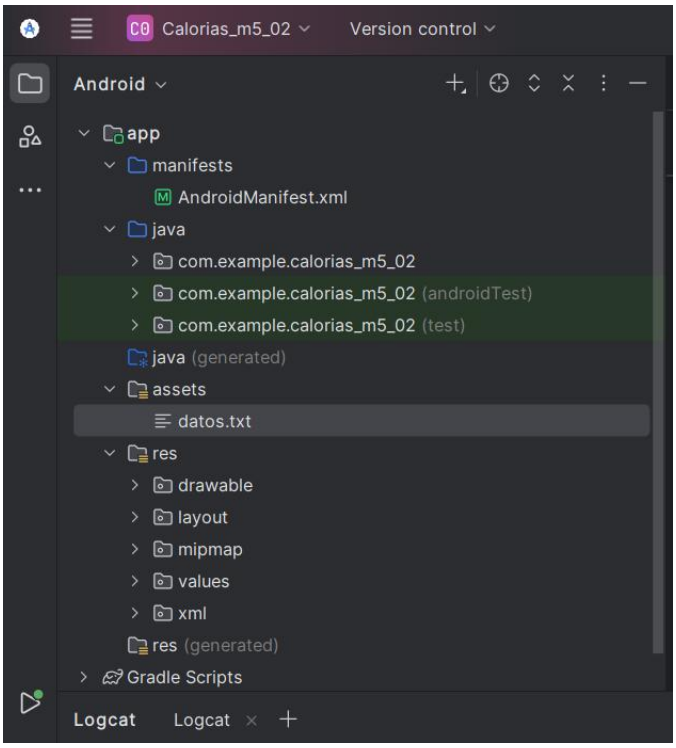
CONTEO DE CALORÍAS

Objetivo

Desarrollar una programa en android studio utilizando el lenguaje Java para resolver un problema de “conteo de calorías", procesando una gran cantidad de datos de entrada, con el fin de comprender y practicar la implementación de las estructuras de control fundamentales.

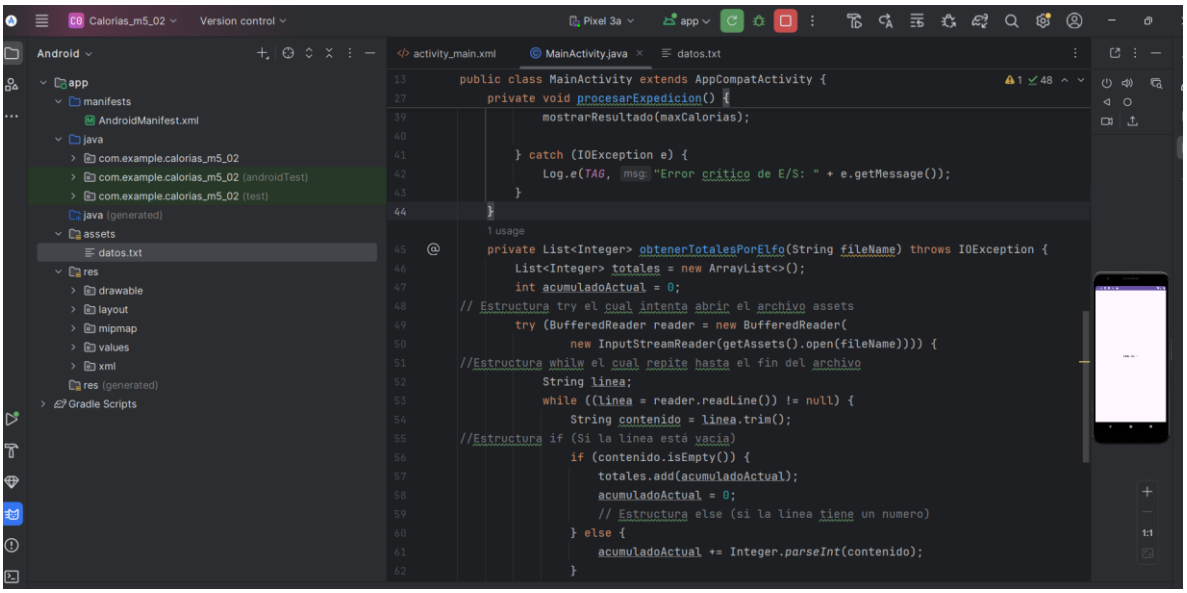
Desarrollo de la actividad

Ilustración 1. Estructura del proyecto



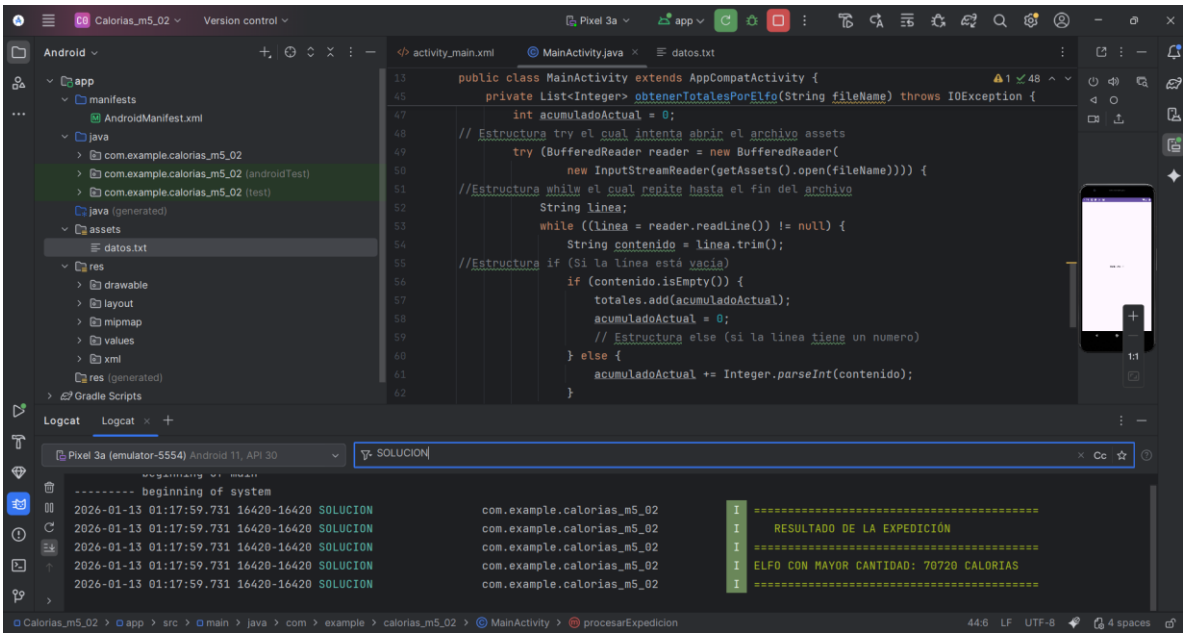
En esta captura de pantalla se observa la organización del proyecto en Android Studio. Se utilizó la carpeta assets para alojar el archivo de texto (datos. txt) con los 2,251 datos de entrada.

Ilustración 2. Código Java



Se implementaron algunas estructuras de control, de las cuales se destaca while para la lectura secuencial del archivo, y la condicional if-else para identificar los saltos de línea (líneas en blanco) que separan el inventario de cada elfo, permitiendo el reinicio del acumulador.

Ilustración 3. Resultado calculado en el logcat



Aquí se muestra la evidencia de la ejecución del programa en el Logcat, tras procesar la totalidad de los datos, el sistema identifica correctamente al elfo con la mayor cantidad de calorías, siendo el resultado obtenido de 70720 calorías.

Conclusión

En el lenguaje Java, las estructuras de control se dividen en tres grandes categorías: las de selección, como if, if-else y switch, que dirigen el flujo según condiciones específicas; las de repetición o bucles, como while, do-while y for, que automatizan tareas iterativas; y las de manejo de excepciones, como try, catch y finally, que garantizan la estabilidad del programa ante fallos inesperados.

Para esta actividad, se utilizaron de forma integrada las tres categorías, en primer lugar, se implementó un bucle while como motor principal para recorrer las más de 2,000 líneas de datos de forma secuencial hasta llegar al final del archivo. Dentro de este bucle, se aplicó una estructura if-else para diferenciar el contenido de cada línea: el if detectaba las líneas en blanco para cerrar el conteo de un elfo y comparar su total, mientras que el else se encargaba de sumar los valores numéricos al acumulador actual. Finalmente, todo este proceso se envolvió en un bloque try-catch, una estructura de control de excepciones indispensable en Android para gestionar de forma segura la lectura de archivos desde la carpeta assets y prevenir que la aplicación se detuviera por errores de formato o de lectura. Gracias a la combinación de estas herramientas, se logró procesar un gran volumen de datos, obteniendo un resultado preciso de 70720 calorías.

Referencias

Android Developers. (2024). *Manage external storage: Assets folder*. https://developer.android.com/guide/topics/resources/providing-resources

Deitel, P., & Deitel, H. (2017). *Java: How to Program* (11° ed.). Pearson Education.

Oracle. (2023). *Java Platform, Standard Edition Documentation: Control Flow Statements*. https://docs.oracle.com/javase/tutorial/java/nutsandbolts/flow.html

Google Developers. (s.f.). *Write and view logs with Logcat*. Android Studio Documentation. https://developer.android.com/studio/debug/am-logcat