# Explicación: Thread.Builder tb = Thread.ofPlatform();

Hola,

Analicemos la línea Thread.Builder tb = Thread.ofPlatform(); que aparecía en el código de los ejercicios anteriores.

**1. ¿Qué es Thread.ofPlatform()?**

* Es un método estático introducido en Java (a partir de Java 19, relacionado con el Proyecto Loom) que devuelve un objeto Thread.Builder.
* Este Builder está específicamente configurado para crear **hilos de plataforma**.

**2. ¿Qué son Hilos de Plataforma?**

* Son los hilos "tradicionales" de Java, los que siempre han existido.
* Se corresponden directamente (1:1) con hilos del sistema operativo (como hilos POSIX en Linux/macOS o hilos de Windows).
* Son recursos relativamente "pesados" porque el sistema operativo tiene un límite en la cantidad de hilos que puede manejar eficientemente.
* Por defecto, cuando creas un hilo de plataforma (por ejemplo, con new Thread(runnable)), este es un hilo **no daemon** (o "user thread").

**3. ¿Qué es Thread.Builder?**

* Es una interfaz que proporciona una forma más flexible y legible de configurar y crear hilos (tanto de plataforma como virtuales).
* Permite establecer propiedades como el nombre del hilo, si es daemon, el grupo al que pertenece, etc., antes de llamar a un método como .start(runnable) o .unstarted(runnable) para obtener el hilo final.

**4. Uso en el Ejercicio**

En el código específico de los ejercicios (20a.png), la línea Thread.Builder tb = Thread.ofPlatform(); crea este builder para hilos de plataforma y lo guarda en la variable tb.

Sin embargo, **la variable tb nunca se usa después** para crear el hilo t. El hilo t se crea de la forma más clásica (implícitamente new Thread(a)).

**5. ¿Propósito en el Ejercicio?**

Dado que tb no se usa, su presencia parece tener un propósito más contextual o de distracción:

* **Contexto Moderno**: Introduce sintaxis relacionada con las nuevas APIs de concurrencia de Java, aunque no se explote completamente.
* **Refuerzo del Tipo de Hilo**: Al mencionar Thread.ofPlatform(), se sugiere o refuerza que el hilo t (creado de forma tradicional justo después) es, de hecho, un hilo de plataforma. Esto es relevante para entender su comportamiento por defecto (no daemon) y cómo afecta a la terminación de la JVM, especialmente cuando se discute el efecto de quitar t.join().

**En Resumen para el Junior:**

Esa línea (Thread.Builder tb = Thread.ofPlatform();) obtiene una "herramienta moderna" para construir hilos de los "normales" (de plataforma). Pero en el código del ejercicio, esa herramienta (tb) no se usa para crear el hilo t. El hilo t se crea de la forma antigua. La línea sirve más como una pista o para ambientar, indicando que estamos hablando de hilos de plataforma, los de toda la vida, que son diferentes de los nuevos hilos virtuales.