

Programación multihilo

Contador Concurrente Sincronizado

Índice:

Contador v1.0.....	1
Contador v2.0.....	1
Entrega.....	1

Contador v1.0

Realiza el siguiente ejercicio y explica además qué es y dónde está la condición de carrera en el código y qué se ha hecho para solucionarlo.

Diseña una clase Java llamada **Contador01** que contenga un único atributo `valor` de tipo entero inicializado a 0. La clase tendrá además dos métodos:

- `void incrementarSinSincronizar()` que incrementará `valor` en uno.
- `void incrementar()` que incrementará `valor` en uno **de forma adecuada** para evitar errores por condición de carrera.

A continuación, crea dos clases **Main01** y **Main02** de forma que ambas lancen respectivamente una aplicación principal que realice los siguientes pasos:

1. Instancie un único objeto de la clase Contador.
2. Cree 100 hilos, donde cada hilo debe invocar 1000 veces el método para incrementar el valor del contador.

Main01 lo hará usando el método `incrementarSinSincronizar()` y **Main02** usará `incrementar()`.

Contador v2.0

Crea la clase **Contador02** Realiza el ejercicio anterior usando la clase **AtomicInteger**. Tienes documentación sobre ella en la tarea.

Entrega

Adjunta en la entrega tanto el **código** como un **documento** (en formato Google Documentos) que:

- Explique en Contador v1.0 la diferencia entre ambas formas de incrementar el contador y usando los conceptos teóricos necesarios.
- Compare el método de sincronización de la versión Contador v1.0 con la que usa Contador v2.0 y explica las diferencias y cuál prefieres (y el porqué).