

Programación de Hilos en Java

Ejercicios de Bloqueos Explícitos y Avanzados. Colecciones Seguras

Ejercicio 1:

Desarrolla una simulación de un parking con plazas limitadas usando programación multihilo en Java.

Requisitos:

1. **Plazas limitadas:** El parking tiene un número fijo de plazas. Este número debe ser configurable.
2. **Vehículos:** Cada vehículo será representado por un hilo independiente que intentará entrar al parking. Si hay plazas disponibles, el vehículo podrá estacionar. Si no hay plazas disponibles, deberá esperar a que se libere alguna.
3. **Tiempo de estacionamiento:** Cada vehículo que logre estacionar permanecerá dentro del parking durante un tiempo aleatorio (por ejemplo, entre 1 y 5 segundos).
4. **Salida del parking:** Después de haber estacionado el tiempo correspondiente, el vehículo debe liberar su plaza, permitiendo que otros vehículos puedan entrar.

Ejercicio 2:

Extiende el ejercicio anterior de simulación de un parking con plazas limitadas. En esta nueva versión, debes implementar un sistema de facturación que calcule automáticamente el costo de estancia de cada vehículo en función del tiempo que permanezca en el parking.

Requisitos Adicionales:

1. Implementa una variable que mantenga un registro de la facturación total acumulada en el parking.
2. Cada vehículo deberá pagar 10 € por cada segundo que permanezca en el parking. Al salir, se debe calcular el costo total basado en el tiempo de estancia.
3. Protege el acceso a la variable de facturación utilizando un **Lock** para evitar condiciones de carrera. Asegúrate de que solo un hilo pueda actualizar la facturación a la vez.
4. Al finalizar la simulación, muestra el total acumulado en la facturación del parking. Además, cada vez que un vehículo salga, imprime el costo total que debe pagar.

Ejercicio 3:

Implementa un programa en Java que simule el problema del Productor-Consumidor utilizando hilos y una estructura de datos concurrente. En este escenario, varios **productores** generarán números enteros aleatorios y los depositarán en una **cola compartida** de tamaño fijo. Simultáneamente, varios **consumidores** tomarán los números de la cola, calcularán su número de Fibonacci correspondiente e imprimirán el resultado en la consola.

Productores:

- Cada productor debe generar un número entero aleatorio entre 1 y 20.
- El número generado debe colocarse en una **cola compartida concurrente**.

Consumidores:

- Los consumidores deben tomar los números de la cola.
- Por cada número que tomen, deben calcular el **n-ésimo número de Fibonacci** correspondiente.
- Una vez calculado, deben imprimir en la consola el número original y su valor de Fibonacci.

$$F_0 = 0$$

$$F_1 = 1$$

$$F_n = F_{n-1} + F_{n-2}$$