

Programación de Servicios y Procesos

Tarea evaluable 2: Crea un programa que simule el funcionamiento de una biblioteca. En la biblioteca hay dos tipos de acciones que los usuarios pueden realizar: **tomar un libro prestado** o **devolver un libro**. Hay 10 libros disponibles en la biblioteca al principio del día, y los usuarios pueden hacer cualquiera de estas dos acciones de manera aleatoria.

1. El programa debe tener dos variables globales: **libros_disponibles** y **usuarios_prestamos**, donde:
 - **libros_disponibles** guarda cuántos libros quedan en la biblioteca.
 - **usuarios_prestamos** guarda cuántos libros han sido prestados en total durante el día.
2. Se debe crear una clase base común (Biblioteca) y dos clases específicas que heredarán de ella, la clase **ClientePrestamo** y **ClienteDevolucion** que realizarán un cálculo diferente. Ambas clases simulan la interacción de un usuario con la biblioteca.
 - **Si un usuario toma un libro (ClientePrestamo):** La cantidad de libros disponibles disminuye en 1 (si hay libros disponibles).
 - **Si un usuario devuelve un libro (ClienteDevolucion):** La cantidad de libros disponibles aumenta en 1 (si el usuario tiene libros prestados).
3. Se deben crear 15 hilos. Cada hilo representará un usuario que realiza una acción aleatoria (tomar o devolver un libro).
 - Si el hilo representa a un usuario que toma un libro, ejecutará la clase **ClientePrestamo** y el nombre del hilo será Hilo_tomar_k (k empezará en 1 y se incrementará por cada Hilo_tomar_k).
 - Si el hilo representa a un usuario que devuelve un libro, ejecutará la función **ClienteDevolucion** y el nombre del hilo será Hilo_devolver_k (k empezará en 1 y se incrementará por cada Hilo_devolver_k).
4. Las operaciones críticas deben estar protegidas con un **Lock** para evitar problemas de concurrencia.
5. Al final del programa, una vez que todos los usuarios han realizado sus acciones, se debe mostrar la cantidad de libros disponibles en la biblioteca y el número total de préstamos realizados durante el día.
6. Implementa una captura de excepciones dentro de los hilos utilizando el bloque **with**.
7. **Importante:** Comentar el código con explicaciones completas sobre que se pretende en cada línea de código.

Entrega: archivo .py con la solución.