1. Este ejercisio ejecuta un bucle infinito de PING PONG a pesar de poner stop en las variables , igual seguiría si no interviniéramos manualmente.

El ‘try-catch’ alrededor del método ‘sleep()’ en el método ‘run()’ no es necesario ya que ‘interruptedException no se lanza desde el método ‘sleep()’ en este caso

1. creación del Thread a través de la interfaz Runnable.

Es el mismo PingPong pero se ejecuta cada 2 segundos y la diferencia que ahora lo hacemos utilizando la interfaz Runnable, aunque no funciona el código por un error en la clase sleep.

1. Clase activa con thread auto lanzado

Donde el código nos devuelve en ejecución cada 2 segundos

Texto

Descripción generada automáticamente

1. El método curren thread() se usa para la clase Recurso para obtener una referencia al hilo actual y luego se imprime el nombre de ese hilo en consola. Entonces al llamar a estos dos nombres luego se imprimirán por consola juan e ines .
2. Y 6 .
3. En el primer ejemplo con isAlive() se pone en espera el código hasta que agente halla terminado , pero de esta forma se consumen muchos recursos del CPU ya que el bucle while se ejecuta constantemente sin descanso. Seria mejor remplazarlo por un join para que lo bloquee hasta que el hilo este terminado
4. Interrupted:

El programa muestra de como funciona el isInterrupted() donde verifica si un hilo fue interrumpido si es así el interrupt() funciona para devolver true o false depende de si se esta ejecutando el hilo o interrumpiendo .