

## THREAD

MÉTODOS	MISIÓN
<b>start()</b>	Hace que el hilo comience la ejecución; la máquina virtual de Java llama al método <b>run()</b> de este hilo.
<b>boolean isAlive()</b>	Comprueba si el hilo está vivo
<b>sleep(long mills)</b>	Hace que el hilo actualmente en ejecución pase a dormir temporalmente durante el número de milisegundos especificado. Puede lanzar la excepción <i>InterruptedException</i> .
<b>run()</b>	Constituye el cuerpo del hilo. Es llamado por el método <b>start()</b> después de que el hilo apropiado del sistema se haya inicializado. Si el método <b>run()</b> devuelve el control, el hilo se detiene. Es el único método de la interfaz <b>Runnable</b> .
<b>String toString()</b>	Devuelve una representación en formato cadena de este hilo, incluyendo el nombre del hilo, la prioridad, y el grupo de hilos. Ejemplo: Thread[HILO1,2,main]
<b>long getId()</b>	Devuelve el identificador del hilo.
<b>void yield()</b>	Hace que el hilo actual de ejecución pare temporalmente y permita que otros hilos se ejecuten.
<b>String getName()</b>	Devuelve el nombre del hilo.
<b>setName(String name)</b>	Cambia el nombre de este hilo, asignándole el especificado como argumento.
<b>int getPriority()</b>	Devuelve la prioridad del hilo.
<b>setPriority(int p)</b>	Cambia la prioridad del hilo al valor entero p.
<b>void interrupt()</b>	Interrumpe la ejecución del hilo
<b>boolean interrupted()</b>	Comprueba si el hilo actual ha sido interrumpido.
<b>Thread currentThread()</b>	Devuelve una referencia al objeto hilo que se está ejecutando actualmente.
<b>boolean isDaemon()</b>	Comprueba si el hilo es un hilo Daemon. Los hilos daemon o demonio son hilos con prioridad baja que normalmente se ejecutan en segundo plano. Un ejemplo de hilo demonio que está ejecutándose continuamente es el recolector de basura ( <i>garbage collector</i> ).
<b>setDaemon(boolean on)</b>	Establece este hilo como hilo Daemon, asignando el valor <i>true</i> , o como hilo de usuario, pasando el valor <i>false</i> .
<b>void stop()</b>	Detiene el hilo. Este método está en desuso.
<b>Thread currentThread()</b>	Devuelve una referencia al objeto hilo actualmente en ejecución.
<b>int activeCount()</b>	Este método devuelve el número de hilos activos en el grupo de hilos del hilo actual.
<b>Thread.State getState()</b>	Devuelve el estado del hilo: NEW, RUNNABLE, BLOCKED, WAITING, TIMED_WAITING, TERMINATED