

Método	Tipo de retorno	Descripción
<code>currentThread()</code>	Thread	Devuelve una referencia del hilo que actualmente está en ejecución.
<code>destroy()</code>	void	Destruye este hilo, sin realizar ninguna operación de limpieza. El hilo pasa a estado terminado.
<code>getName()</code>	String	Devuelve el nombre del hilo
<code>getPriority()</code>	int	Devuelve la prioridad del hilo
<code>interrupt()</code>	void	Envía este hilo al estado de preparado. Obliga al hilo a dejar de hacer lo que está haciendo. Lo usamos, por ejemplo, para “despertarlo”.
<code>isAlive()</code>	boolean	Verifica si este hilo está vivo(no ha terminado)
<code>isDaemon()</code>	boolean	Verifica si este hilo es un hilo demonio.
<code>isInterrupted()</code>	boolean	Verifica si este hilo ha sido interrumpido.
<code>Join([miliseg [,nanoseg])</code>	void	bloquea indefinidamente o el tiempo especificado al hilo. El hilo despierta cuando: a) fin de tiempo pasado como argumento. b) otro hilo ejecuta <code>interrupt()</code> para él. c) todos sus hilos hijos han finalizado.
<code>run()</code>	void	Contiene el código que se ejecutará cuando el hilo pase al estado ejecución. Tarea del hilo.
<code>setDaemon(boolean)</code>	void	Define este hilo como un demonio si se le pasa true o como un hilo de usuario si se le pasa false .
<code>setName(String)</code>	void	Cambia el nombre de este hilo. Por defecto los hilos tienen nombre Thread1, Thread2,...
<code>setPriority(int)</code>	void	Cambia la prioridad de este hilo. Por defecto un hilo hereda la prioridad de su hilo padre . Por omisión es normal(NORM_PRIORITY). Valores [1,10]. MIN_PRIORITY Y MAX_PRIORITY
<code>sleep(long miliseg [,int nanoseg])</code>	void	Envía este hilo a dormir por el tiempo especificado. EL hilo despierta cuando transcurra el tiempo o hasta que se ejecute para él <code>interrupt()</code>
<code>start()</code>	void	Inicia la ejecución de este hilo. Pone al hilo en estado listo (cola de hilos preparados). Lo hace visible al planificador de hilos. La máquina virtual de Java invoca al método run() del hilo.
<code>yield()</code>	void	Detiene temporalmente la ejecución de este hilo para permitir la ejecución de otros. Envía el hilo a la cola de hilos preparados. Planificación cooperativa. Si cola vacía el planificador de hilos no lo ejecuta.
<code>notify()</code>	void	Despierta un hilo de los que están en estado esperando por el monitor de este objeto .(se ejecutó para ellos <code>wait()</code>). Cede el control del monitor del objeto. Se invoca siempre desde dentro de una sección crítica.
<code>notifyAll()</code>	void	Despierta todos los hilos que están en estado esperando por el monitor de este objeto.(se ejecutó para ellos <code>wait()</code>) Cede el control del monitor del objeto. Se invoca siempre desde dentro de una

[illegible]