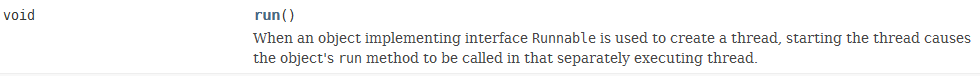
Runnable接口和Thread类

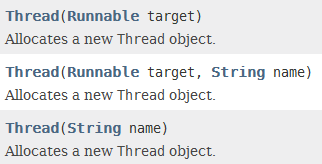
1. **Runnable接口**:存在于**java.lang包**中，是一个**功能接口（Functional Interface），**要想开创一个新的线程，必须直接或间接的实现**此接口且重写其中的run方法**。**run方法**是用来封装线程的任务。
2. **Runnable接口中**只有一个**run方法**：此方法用于封装**新线程要执行的任务**。



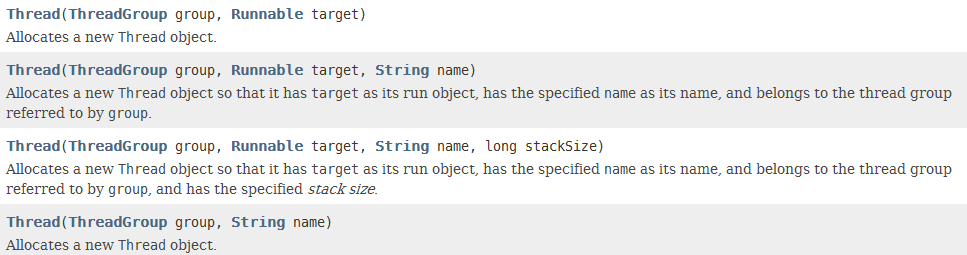
1. **Runnable接口**有许多**直接实现类**，其中最重要的是**Thread类**。
2. **Thread类**：也存在于**java.lang包**中，直接父类就是Object，实现了Runnable接口，并覆写了**Runnable接口**中的**run方法**。要想创建一个新的线程，必须覆盖其中的run方法，因为Thread类中的run方法相当于是空的。
3. **开启一个线程必须是调用start方法，如果是调用run方法，就相当于调用一个普通方法一样，执行完run方法后，代码才会继续进行下面的。利用start方法开启线程后，就会直接和主线程同时随机的执行。**
4. Thread类的构造方法：
5. 空参数：



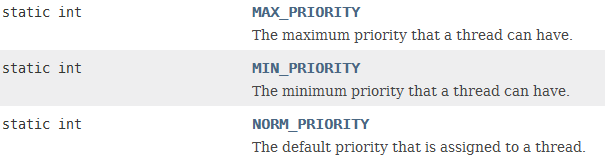
1. **接收一个线程对象**：并可以**指定线程名字**String name。



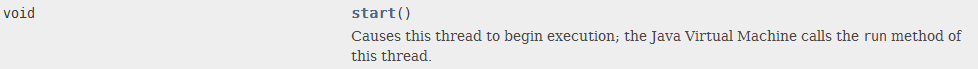
1. **使用了ThreadGroup的构造方法**：



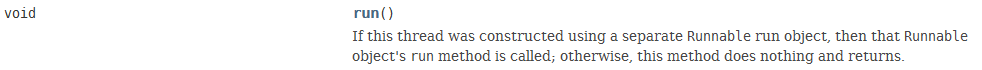
1. **Thread类的字段**：线程的优先级范围是1—10，把**优先级为1、5、10**封装成了**字段值**。



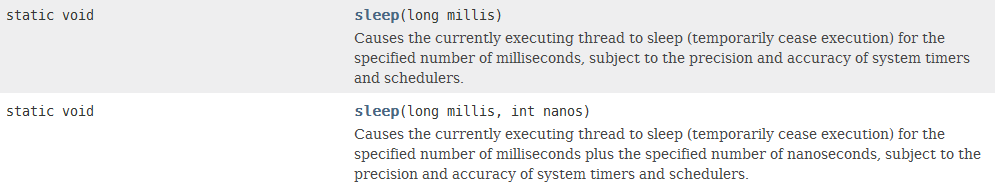
1. **Thread类**的成员方法：
2. **start方法**：**启动此线程**。



1. **run方法**：**封装线程执行的任务**。



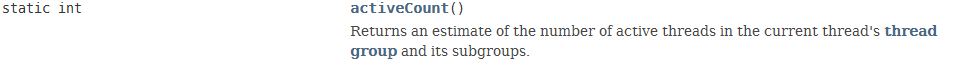
1. **sleep方法**：



1. **currentThread（）静态方法**：获取当前线程对象。



1. **activeCount静态方法**：获取当前**正在进行的线程**的个数。



1. checkAccess方法：



1. **getXxxx方法**：
2. **getId方法**：获取线程的ID。



1. **getName()方法**：获取线程的名字。



1. **getPriority()方法**：获取线程的优先级。



1. **setXxx方法**：
2. **setName()**



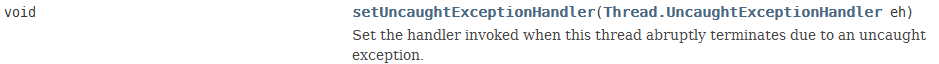
1. **setPriority()方法**: **设置优先级，数值只有1—10；默认的值是5.**

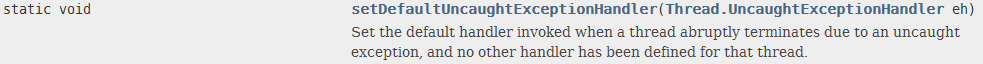


1. **setDaemon(Boolean on)**：**守护线程**。



1. 其他的setXxxx（）



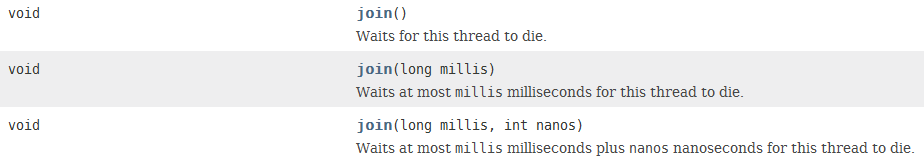




1. **interrupt（）：中断此线程**。



1. **join（）方法**：



1. **判断方法**：
2. interrupted：静态方法，判断此线程是否是被中断了。



1. isInterrupted：



1. isAlive：

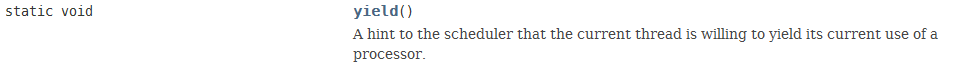


1. isDaemon():是否是后台运行。

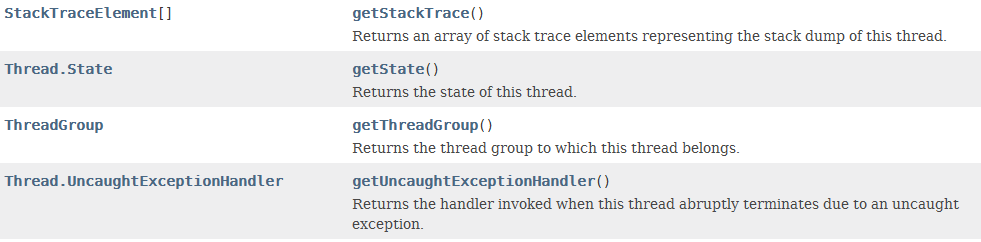
**daemon ['diːmən]n. 守护进程；后台程序**



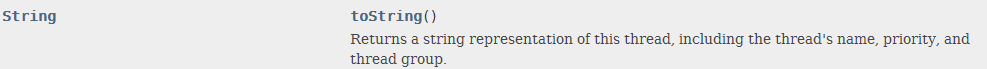
1. yield：**释放cpu执行资格，暂停自己**。



1. getXxxx（）：



1. **toString:返回线程的表现形式，包括线程名、优先级和线程组。**



1. Thread类的静态方法：

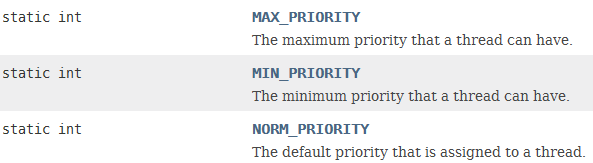
**一个类的静态方法体现了该类的具有的共性**，获取该类的共同性质。

如Thread类的静态方法有：

1. currentThread()获取当前的线程对象。
2. activeCount()获取当前线程池中active的线程个数，返回int值。
3. holdsLock（Object obj）判断**当前线程**是否持有obj锁。
4. interrupted()判断当前线程是否已经中断。

等等。

1. Java虚拟机中通过**线程调度器**来管理线程，调度器把线程分成10个等级，1—10之间。其中java已经把优先级**1、5、10**封装成了静态常量。**MAX\_ PRIORITY、MIN\_ PRIORITY、NORM\_PRIORITY**。



**线程（包括主线程）被创建时，默认的优先级级别都是5。**

通过**setPriority方法**设置优先级。

注意：**优先级并不能完全代表执行的顺序，只是一种概率大小的含义，优先级高的线程先执行的概率较大，但是低优先级的线程也有可能先执行。**

1. **守护线程**： **setDaemon(boolean b)方法**可以将此线程设置为**守护线程或用户线程**。This method must be invoked before the thread is started.该方法必须在调用start方法前调用。守护线程与一般线程**运行、等待**都一样，只不过结束的时候不一样，只要**一般线程消失，守护线程会自动消失**，无论其处于何种状态。当所有的线程都是守护线程时，此时**程序会自动退出**。
2. **join方法**:线程联合。作用：当前运行的线程进入阻塞状态，等待调用join方法的线程运行完毕后，再继续执行，注意主线程只等调用join方法的线程运行完毕，调用**join方法**也会抛出异常。**临时加入一个线程可以使用join方法。**