Java线程

1. 线程的创建

线程的创建可以分为两种方式:A）继承Tread类；B）实现Runnable接口

两种创建线程的方式区别和联系主要有哪些?

1）、Java单继承机制，限制了Thread类的使用；然后可以通过实现Runnable接口实现多线程，同时也可继承其他类来实现其他功能；

2）、Runable可以使用多线程去处理同一资源，同时也增加了程序的健壮性，相同的代码可以被多个线程共享，这是Thread类所不具备的;

如买票的经典例子:

|  |
| --- |
| **public** **class** Test {  **public** **static** **void** main(String[] args) {  ThreadTest thread1 = **new** Test.ThreadTest();  ThreadTest thread2 = **new** Test.ThreadTest();  thread1.start();;  thread2.start();  RunableTest runable = **new** Test.RunableTest();  **new** Thread(runable).start();;  **new** Thread(runable).start();;  }  **static** **class** ThreadTest **extends** Thread{  **private** **int** ticket = 10;  @Override  **public** **void** run() {  **for** (**int** i = 0; i < 20; i++) {  **if**(**this**.ticket > 0){  System.***out***.println("[" + Thread.*currentThread*().getName() + "] -- " + **this**.ticket--);  }  }  }  }  **static** **class** RunableTest **implements** Runnable {  **private** **int** ticket = 10;  **public** **void** run() {  **for** (**int** i = 0; i < 20; i++) {  **if**(**this**.ticket > 0){  System.***out***.println("[" + Thread.*currentThread*().getName() + "] -- " + **this**.ticket--);  }  }  }  }  } |

PS:

1、JDK中Thread类也是Runnable接口的子类

2、Thread的run()和start()的关系(JDK源码可知)：

start()方法使用native关键字修饰，该关键字表示调用操作系统的底层函数(JNI);

Thread的start()一旦被调用，JVM则会去调用run()方法;

1. 线程的同步