线 程

目标：

1. 理解线程的基本概念
2. 掌握编写线程的两种方法
3. 搞清楚线程的运行机制
4. 基本概念

要理解线程首先必须了解什么是进程，线程一般都是在进程中的

a、什么是进程呢：

进程是指运行中的应用程序，每个进程都有自己独立的地址空间(内存)，比如我们

打开一个火狐浏览器，就启动了一个进程，再打开一个火狐浏览器，就又启动了一个进程

b、什么是线程呢：

线程是进程中的一个实体，是被系统独立调度和分派的基本单位，线程自己不拥有系统资源,只拥有一点在运行中必不可少的资源，但它可与同属性一个进程的其它线程共享进程所拥有的全部资源，一个线程可以创建和撤消另一个线程，同一进程的多个线程之间可以并发执行，线程有就绪、阻塞和运行三种基本状态

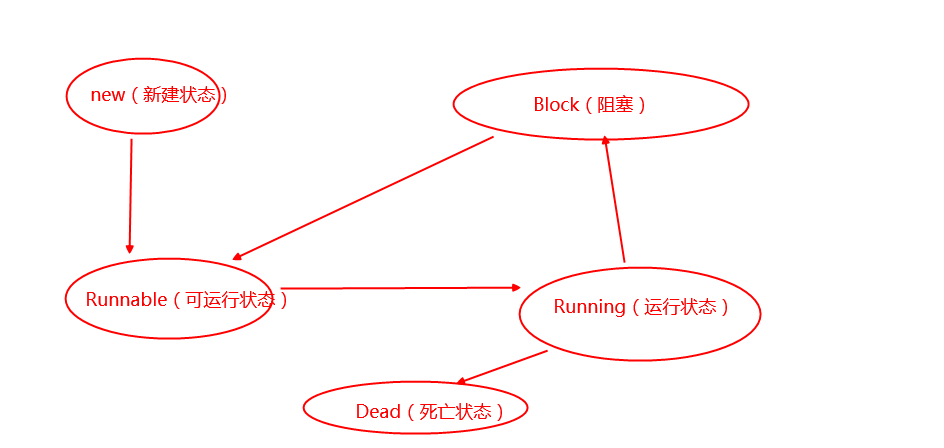


我们换一种说法，给同学们解释一下：

什么是线程：

1. 线程是轻量级的进程
2. 线程没有独立的地址空间(内存空间)
3. 线程是由进程创建的(寄生在进程)
4. 一个进程可以有多个线程----🡪也就是我们常说的多线程，线程增加进程地址空间(内存)随之扩大(32位和64位寻址范围)
5. 线程有几种状态
6. 新建状态(new)
7. 就绪状态(Runnable)
8. 运行状态(Running)
9. 阻塞状态(Blocked)
10. 死亡状态(Dead)

画图说明



线程----如何使用

在java中一个类要当成线程使用有两种方法

1. 继承Thread类，并重写run函数
2. 实现Runnable接口，并重写run函数

案例1：

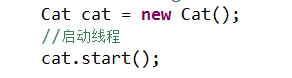
1、继承Thread类来实现一个线程，使程序可以每隔一秒，在控制台输出”Hello World”

Thread类在java.lang包下，系统会自动引入lang包

2、，使程序可以每隔一秒，在控制台输出”Hello World”，打出10次后结束

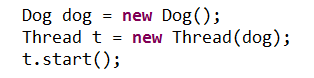


启动线程：



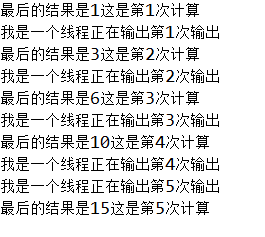
案例2：使用Runnable接口来实现

启动方式



多线程

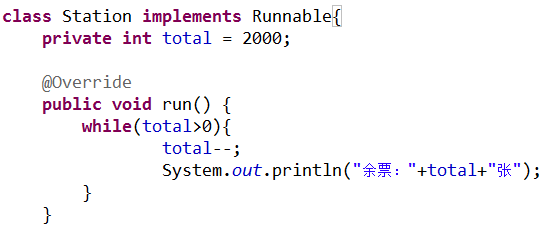
Eg:同时启动两个线程来查看输出效果

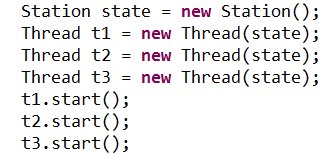


注意：线程对象只能启动一个线程，不管是继承Thread还是实现Runnalbe接口，每一个线程对象只能启动一个线程

我们看下面这个案例：

模拟一个站点有三个窗口售票(三个线程)，一天共卖出2000张票，出票都是从总部出的。





Java线程同步

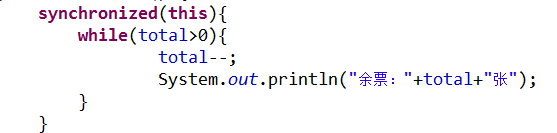
多线程并发给我们编程带来很多好处，可以同时完成更多任务，但也给我们带来了线程安全的问题，像上面的案例，就可能出现一票多卖的情况，

那我们怎么解决这个问题呢，那就要保证我们代码的原子性，也就是每次只有一个线程执行，该线程执行完了，其它线程再执行

解决方法：线程同步

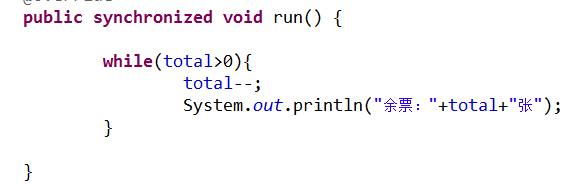
Java中处理线程同步的方法非常简单，只需把需要同步的代码段放在synchronized(对象锁){ }中即可，就好像给这段程序加了个锁，一个线程出来，其它线程才能进去

注意：不要用当前对象来锁定当前对象（即不要用this）



使用方法二、

直接在方法前加synchronized关键字



总结：继承Thread和实现Runnable接口的区别：

最主要的区别在于启动方式不一样，如果没有强制要求，我们一般使用Runnable接

口，扩展性好，可同时继承其它类。