

# JAVA DAY 16

## 1. 概念

1) 并发: 指两个或多个事件在同一段时间内发生。

2) 并行: 指两个或多个事件在同一时刻发生 (同时发生)

3) 进程: 是指一个内存中运行的应用程序, 每个进程都有一个独立的内存空间, 进程也是程序的一次执行过程, 是系统运行程序的基本单位; 系统运行一个程序即是一个进程从创建、运行到消亡的过程。

4) 线程: 线程是进程中的一个执行单元, 负责当前进程中程序的执行, 一个进程中至少有一个线程, 一个进程中是可以有多个线程的, 这个应用程序可以称之为多线程程序。简而言之, 一个程序运行后至少有一个进程, 一个进程中可以包含多个线程

多线程并发执行并不能提高程序的运行速度, 但可以提高程序的效率, 可以同时完成多项工作, 让 CPU 的使用率更高

宏观上并行 微观上串行

JVM运行时是多线程的, 至少启动垃圾回收线程和主线程

## 2. 线程三要素 CPU、代码、数据

每个线程都有自己的线程栈。线程方法中的局部变量  
每个线程都有自己的副本

## 3. Java中实现多线程

### a) 继承 Thread 类 (建议)

好处是：可以直接使用 Thread 类中的方法，代码简洁

弊端是：如果已经存在了一类，就不能再用这种方法

### b) Runnable 接口实现

好处：即使自己定义的线程类有父类也没关系。

因为有了接口也可以实现接口，且接口是可以多实现的。

弊端：不能直接使用 Thread，要在类里实现 Runnable 接口，不能继承 Thread

## 4. ~~多线程共享变量~~ 2

~~线程继承 Thread~~

## 4. 多线程共享变量

继承 Thread，用 static 修饰的变量，在所有线程间共享

实现 Runnable，成员变量在所有线程间共享

## 5. 线程方法

getName()  
currentThread()  
join()  
sleep()  
isAlive()

守护线程的优先级比较低,用于为系统中的其他线程提供服务。例如垃圾回收线程就是一个经典的守护线程,当我们的程序中不再有任何运行的 Thread,程序就不会再产生垃圾,垃圾回收器也就无事可做,所以当垃圾回收线程是 JVM 上仅剩的线程时,垃圾回收线程会自动退出。它始终在低级别的状态中运行,用于实时监控和管理系统中的可回收资源。

## 6. 同步代码块

a) 语法:

```
synchronized (同步锁) {  
    // 需要同步的代码  
}
```

同步锁:

对象的同步锁只是一个概念,可以想象为在对象上标记了一个锁

1) 锁对象可以是任意类类型

2) 多个线程对象要使用同一把锁

注意: 在任何时候, 最多允许一个线程拥有同步锁, 谁拿到锁就进入代码块, 其他的线程只能在外等着。

5) 什么情况下需要同步

当多线程并发, 有多段代码同时执行时, 我们希望某一段代码执行的过程中CPU不要切换到其他线程工作, 这时就需要同步。

如果两段代码是同步的, 那么同一时间只能执行一段。在一段代码没执行结束之前, 不会执行另外一段代码。