

# 编程入门基础

—— 线程与进程的区别

主讲教师：耿宇航

# 什么是进程？

- 进程就是一个在内存中独立运行的程序，有自己的地址空间。如正在运行的写字板程序就是一个进程
- “多任务”：指操作系统能同时运行多个进程（程序）。如WIN2K系统可以同时运行写字板程序、画图程序、WORD、EXCEL等

# 什么是线程？

- 线程：是进程内部单一的一个顺序控制流
- 线程和进程
  - 每个进程都有独立的代码和数据空间(进程上下文)，进程切换的开销大。
  - 线程：轻量的进程，同一类的线程共享代码和数据空间，每个线程有独立的运行栈和程序计数器(PC)，线程切换的开销小。
  - 多线程：在同一个进程中有很多个顺序流同时执行

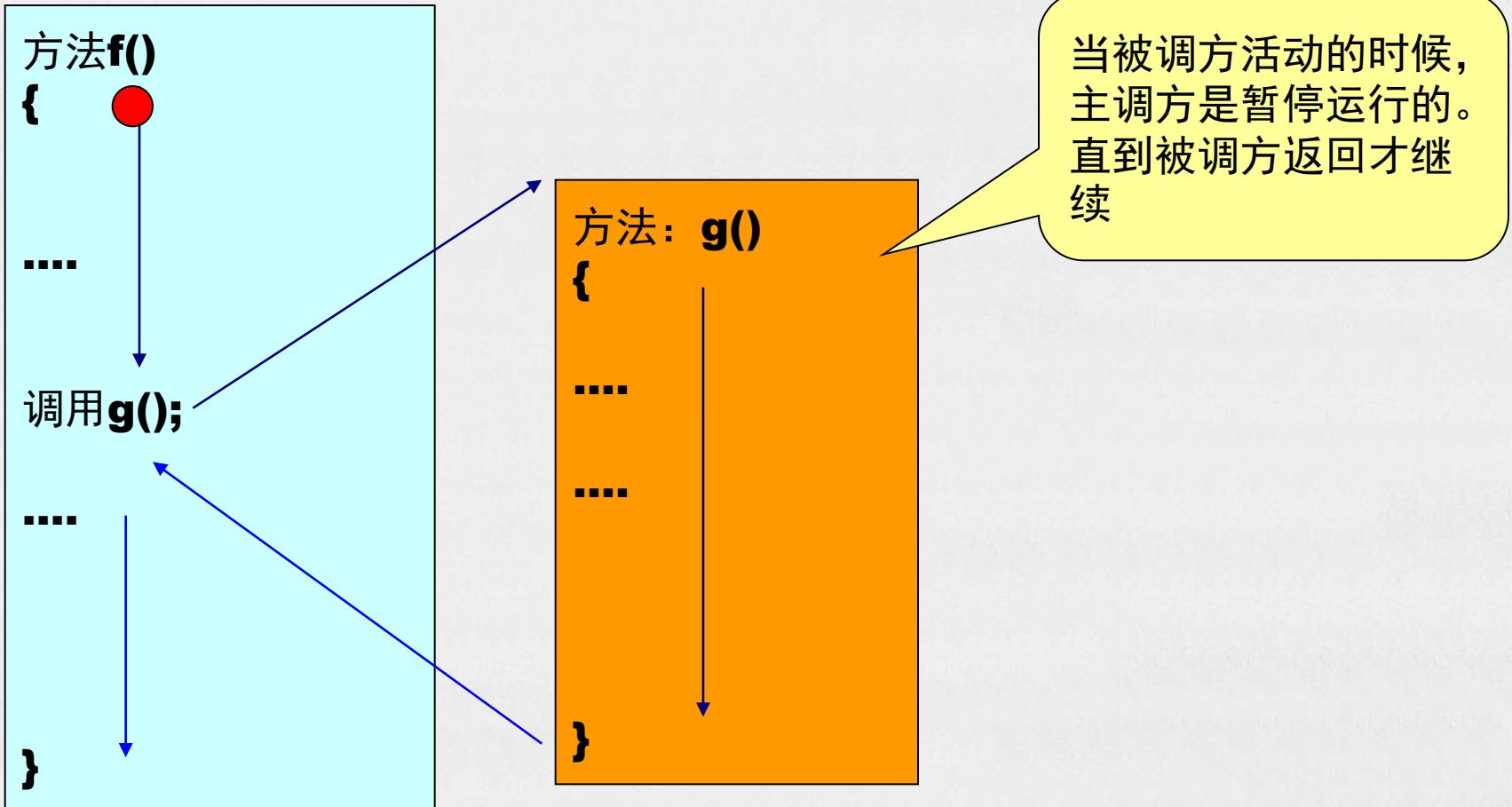
# 名词的比较

- **共同点**
  - 目的都是为了并发
  - 都存在“排斥与同步”的问题
- **不同点**
  - 线程共享内存，而进程是内存隔离的
  - 线程更节省资源
  - 线程又叫“轻进程”

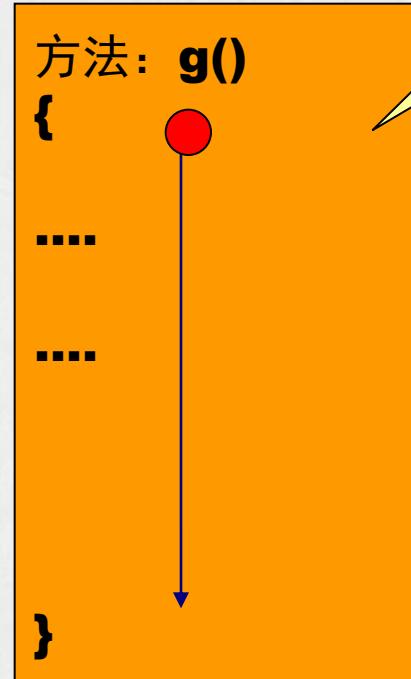
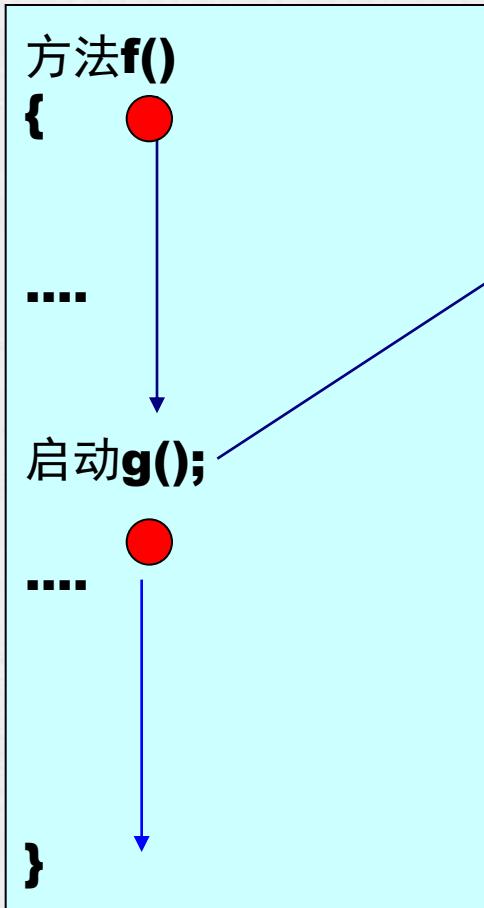
# 怎样并发？

- 调用是不行的
  - 调用永远是排队执行
- 需要java提供的特殊服务
  - 去启动一个方法，而不是直接去调用一个方法。

# 调用的效果



# 启动的效果



启动方与被启动方，并列运行，谁也不用等谁。

# 主线程

- 在任何Java程序启动时，一个线程立刻运行（即main方法对应的线程），该线程通常称为程序的主线程
- 主线程的特点：
  - 它是产生其他子线程的线程
  - 它不一定是最后完成执行的线程，子线程可能在它结束之后还在运行

# 伪装耗时操作

- **Thread.sleep(long x)**
  - x 的单位是毫秒
  - Thread.sleep(1000\*3) 就是把3秒钟消耗掉。
  - 这个方法需要截获异常

# 创建线程的方法

- **继承Thread类**
  - 完成 public void run() 方法体
  - 调用 对象的 start() 启动线程执行
- **实现Runnable接口**
  - 也是完成 public void run() 方法
  - 为什么要使用接口?
    - java的单继承机制

# 谢 谢 !