操作系统可以同时运行多个任务,现代的操作系统比如 Windows、Mac OSX、Linux、Unix等都是支持多任务的系统

了解多任务原理

单核CPU实现多任务的原理: 找图看

多核CPU实现多任务的原理: 找图

为什么要实现多任务:

想效率高,不卡顿

- 1、多进程模式: 启动多个进程,每个进程虽然只有一个线程,但是多个进程可以一起执行多个任务
- 2、多线程模式: 启动一个进程, 在一个进程的内部启动多个线程, 这样多个线程也可以一起执行多个任务
- 3、多进程+多线程: 启动多个进程, 每个进程再启动多个线程
- 4、协程

实现多任务的方式:

5、多进程+协程

什么是进程: 进程(Process)是计算机中的程序关于某数据集合上的一次运行活动,是系统进行资源分配和调度的基本单位 是操作系统结构的基础。在早期面向进程设计的计算机结构中,进程是程序的基本执行实体; 在当代面向线程设计的计算机结 构中,进程是线程的容器。程序是指令、数据及其组织形式的描述,进程是程序的实体

进程的概念

比如:

对于操作系统来说,一个任务就是一个进程。比方说打开浏览器就是启动一个浏览器的进程,在打开一个记事本就启动一个记 事本进程,如果打开两个记事本就启动两个记事本进程。

进程相关

multiprocessing模块

1、跨平台的多进程模块

2、提供了一个Process类代表一个进程对象

进程中的注意事项:

主进程的结束不能影响子进程,随意可以等待子进程结束在结束主进程 等待子进程结束,才能继续执行主进程 后期主进程主要做的是调度相关的工作,不具体负责业务逻辑

表示可以同时执行的进程数量

由于Pool的默认值为CPU的核心数量,如果有4个核心,则至少需要子进程才能看到等待的效果

Pool进程池

注意:

进程池对象调用join之前必须先调用close,调用close之后就不能向进程池中添加进程了

Pool对象调用join方法,会等待所有子进程结束在执行主进程

有名管道 无名管道 队列 共享内存 信号 信号量

进程间的通信:

强制结束进程:

进程名.terminate()