## 同步异步与协程线程进程关系

fqbqrr 2019-06-17 16:09:02 118 收藏

展开

同步异步与协程线程进程关系.

参考:链接

**主要看那个老张烧开水:下面附**

异步执行:老张(调用者),烧开水.是一个调用者-->烧开水.然后烧开水-->(反馈)调用者,调用者-->上水(剩余,直接先写)的过程.

有了协程,是不是可以不要异步了?应该就是!搞一堆协程池,都是干活的.这个事情完了,或者慢得很,就不管了,干其他事情,当反馈后,直接另外一个协程去干活.

应该就是这样.所谓的异步,不过是一件事情分拆成几个动作. 只不过编程的时候都写在一起,执行的时候分开执行而已.

再仔细思考:异步,为什么异步?就是干某件事的时间长,如果瞎等,实在是瞎.所以就必须另起个工作者来专门作这件事.然后你就可以做其他事情了.而不是瞎等.

因而,只要耗时长的事情,都可以异步走起.另开一个工作者.主要其实就是协程.所以,我一直看协程,协程.看了过后,感觉没异步什么事了呢?原来异步就是靠协程或线程来实现.但线程成本太高,因而就用协程了.

当然,做一件事,可能有几件事情都非常消耗时间,这时,用同步手法写异步或者写协程就是非常爽的了.

参考这篇文章,非常好的文章.

其实就是把原来的一根死线程拆解成一堆活协程.这样,你这边慢,我就保存环境,然后干其他事.等你完成后,把要办的活干了.

协程还可以返回值.

Task Add100ByCoroutine(int init, call\_back f)

{

int ret = co\_await Add100AWaitable(init);

ret = co\_await Add100AWaitable(ret);

ret = co\_await Add100AWaitable(ret);

f(ret);//流程化执行.它这里怪,都是相同函数

}

当然,大量计算耗时的任务,还是得交给线程去完成.

————————————————

版权声明：本文为CSDN博主「fqbqrr」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：<https://blog.csdn.net/fqbqrr/java/article/details/92642285>

## 关于异步 同步的一些理解：

同步和异步的区别就在于是否等待IO执行的结果。好比你去麦当劳点餐，你说“来个汉堡”，服务员告诉你，对不起，汉堡要现做，需要等5分钟，于是你站在收银台前面等了5分钟，拿到汉堡再去逛商场，这是同步IO。

你说“来个汉堡”，服务员告诉你，汉堡需要等5分钟，你可以先去逛商场，等做好了，我们再通知你，这样你可以立刻去干别的事情（逛商场），这是异步IO。

老张爱喝茶，废话不说，煮开水。出场人物：老张，水壶两把（普通水壶，简称水壶；会响的水壶，简称响水壶）。

1 老张把水壶放到火上，立等水开。（同步阻塞）老张觉得自己有点傻

2 老张把水壶放到火上，去客厅看电视，时不时去厨房看看水开没有。（同步非阻塞）老张还是觉得自己有点傻，于是变高端了，买了把会响笛的那种水壶。水开之后，能大声发出嘀~~~~的噪音。

3 老张把响水壶放到火上，立等水开。（异步阻塞）老张觉得这样傻等意义不大

4 老张把响水壶放到火上，去客厅看电视，水壶响之前不再去看它了，响了再去拿壶。（异步非阻塞）老张觉得自己聪明了。

**所谓同步异步**，**只是对于水壶而言**。普通水壶，同步；响水壶，异步。虽然都能干活，但响水壶可以在自己完工之后，提示老张水开了。这是普通水壶所不能及的。同步只能让调用者去轮询自己（情况2中），造成老张效率的低下。

**所谓阻塞非阻塞**，**仅仅对于老张而言**。立等的老张，阻塞；看电视的老张，非阻塞。情况1和情况3中老张就是阻塞的，媳妇喊他都不知道。虽然3中响水壶是异步的，可对于立等的老张没有太大的意义。所以一般异步是配合非阻塞使用的，这样才能发挥异步的效用。

————————————————

版权声明：本文为CSDN博主「haeasringnar」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：https://blog.csdn.net/haeasringnar/java/article/details/79978718