

# 在Python中使用协程

扩展：

## gevent模块介绍

在Python当中，我们提供了这样一个模块来实现协程----gevent

当一个程序遇到IO操作时（需要等待），比如访问网络，就自动切换到其他的程序，等到IO操作完成，再在适当的时候切换回来继续执行。由于IO操作非常耗时，经常使程序处于等待状态，有了gevent为我们自动切换协程，就保证总有程序在运行，而不是等待IO。

## gevent模块使用

### 把程序变成协作模式

```
1 monkey.patch_all()
```

### 创建任务

```
1 gevent.spawn()
```

### 开始任务

```
1 gevent.joinall(tasks_list)
```

## 案例

```
1 # 从gevent库里导入monkey模块
2 from gevent import monkey
3 # 把程序变成协作式运行，就是可以帮助程序实现异步
4 monkey.patch_all()
5 # 导入gevent、time、requests
6 import gevent
7 import time
10 # 记录程序开始时间
11 start = time.time()
12 # 定义一个work1()函数
15 def work1(url):
```

```
16     time.sleep(5)
17     print('work1')
18 # 定义一个work1()函数
19 def work2():
20     time.sleep(2)
21     print('work2')
22
23 # 创建空的列表
24 tasks_list = []
25 # 创建任务
26 task1 = gevent.spawn(work1)
27 # 创建任务
28 task2 = gevent.spawn(work2)
29 # 往任务列表添加任务
30 tasks_list.append(task1)
31 tasks_list.append(task2)
32 # 执行任务列表里的所有任务，就是让爬虫开始爬取网站
33 gevent.joinall(tasks_list)
34 # 记录程序结束时间
35 end = time.time()
36 # 打印程序最终所需时间
37 print(end-start)
```

这段代码因为第一个任务延时比较长，所以会先进行任务2，在回头执行任务1