```
import threading

sum_list = []
thread_list = []

class MyThread(threading.Thread):
    def __init__(self, name, *args, **kwargs):
        threading.Thread.__init__(self)
        self.name = name
        self.args = args
        self.kwargs = kwargs

    def run(self):
        global sum_list
        print("开始运行线程:%s(id:%d)" % (self.name, threading.get_ident()))
        ret = self.mysum(self.args, self.kwargs)
        sum_list.append(ret)

    def mysum(self, args, kwargs):
        _sum = 0
        for i in args:
            _sum += i
        for k, v in kwargs:
            if isinstance(v, int):
                _sum += v
        return _sum

for i in range(10):
    start = i * 10000
    t = MyThread('thread-%d' % (i), *[i for i in range(start + 1, start + 10000 + 1)])
    t.start()
    thread_list.append(t)

for i in thread_list:
    i.join()

ret = sum(sum_list)
print("退出主线程,1+2+3+.....+100000=%s" % (ret))
```