## 实现面包店的生产销售模型。

场景如下：

1. 一个面包2元。
2. 有n个面包厨师在造面包，并放入面包篮子里
3. 有m个客户在同时买面包，并买到自己需要的数量的面包即可付款。
4. 所有厨师和客户是同时进行造面包和生成面包。

请使用python多线程并发模型实现上述场景

import threading  
import time  
import random  
class Cooker(threading.Thread):  
 count = random.randint(0, 200)  
 cooker = random.randint(1, 2)  
 price = 2  
 while True:  
 if count >= 100:  
 break  
 elif count<100:  
 count = count + 1  
 sum = count\*cooker  
 def run(self):  
 for i in range(1,self.sum+1):  
 if self.sum>=0:  
 print(**"面包已经卖出了"**, i, **"个，赚了"**,i\*self.price,**"元！！！"**)  
 time.sleep(0.5)  
 if self.sum==i:  
 print(**"面包卖完了！！！欢迎下次光临！！！！"**)  
 break  
 def make\_bread(self):  
 print(**"厨师有"**,self.cooker,**"个，每个人要做面包"**,self.count,**"个，所以要做"**,self.count\*self.cooker,**"个面包"**)  
  
class Customer(threading.Thread):  
 customer = random.randint(0, 20)  
 money = random.randint(0, 5000)  
 sum1 = customer\*money  
  
 def run(self):  
 for i in range(1,self.sum1+1):  
  
 if self.sum1>0:  
 print(**"客户已经买了"**,i,**"个面包，客户余额为： "**,self.sum1-(i\*2),**"元！！！"**)  
 time.sleep(0.5)  
 if self.sum1-(i\*2)<=0:  
 print(**"您的余额不足！！！"**)  
  
 break  
  
  
 def introduce(self):  
 print(**"用户数量为： "**,self.customer,**"每个人有"**,self.money,**"元"**)  
c = Cooker()  
c1 = Customer()  
c.make\_bread()  
c1.introduce()  
c.start()  
time.sleep(0.1)  
c1.start()