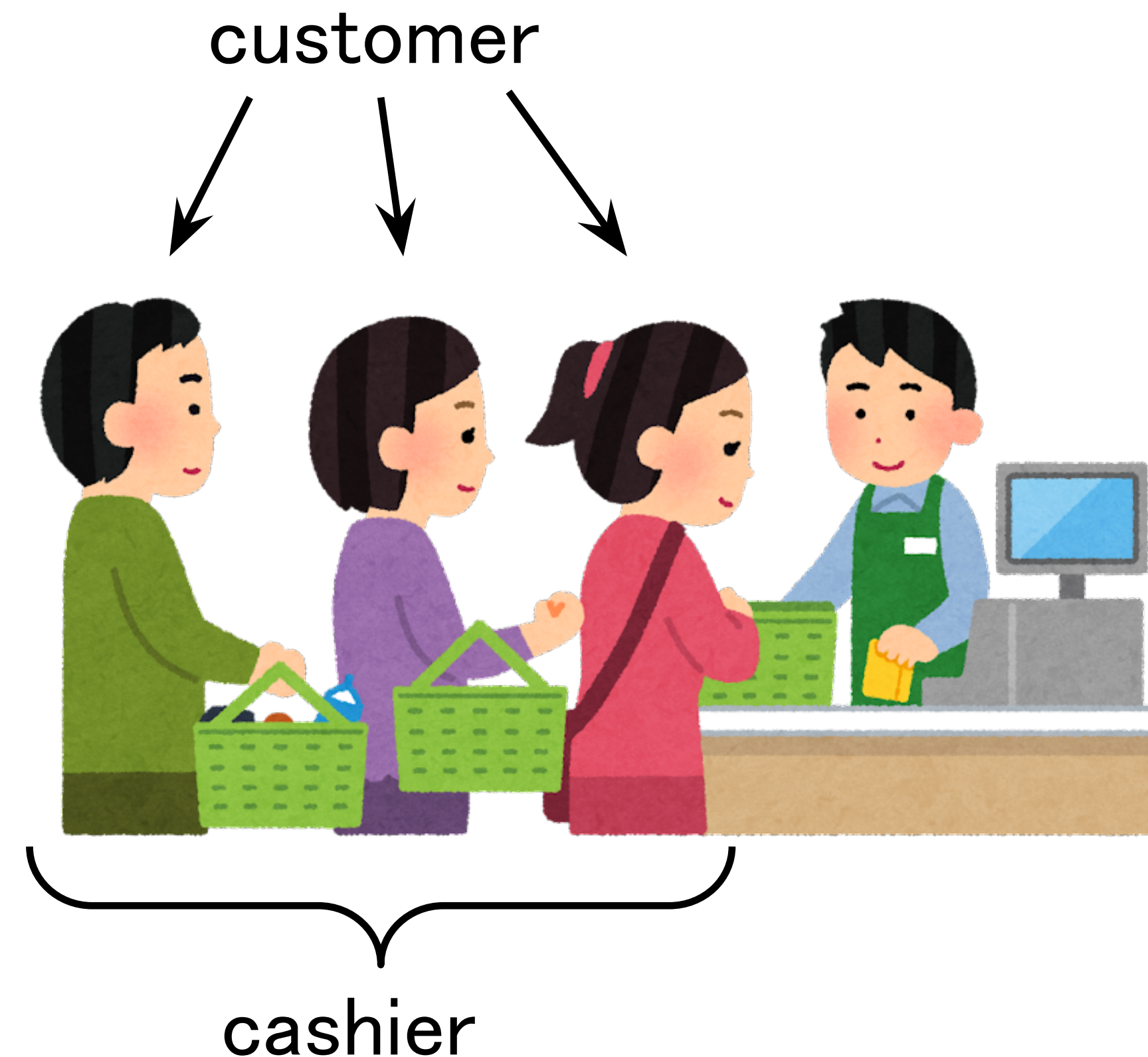


3. 簡単な待ち行列シミュレーションを試みよう



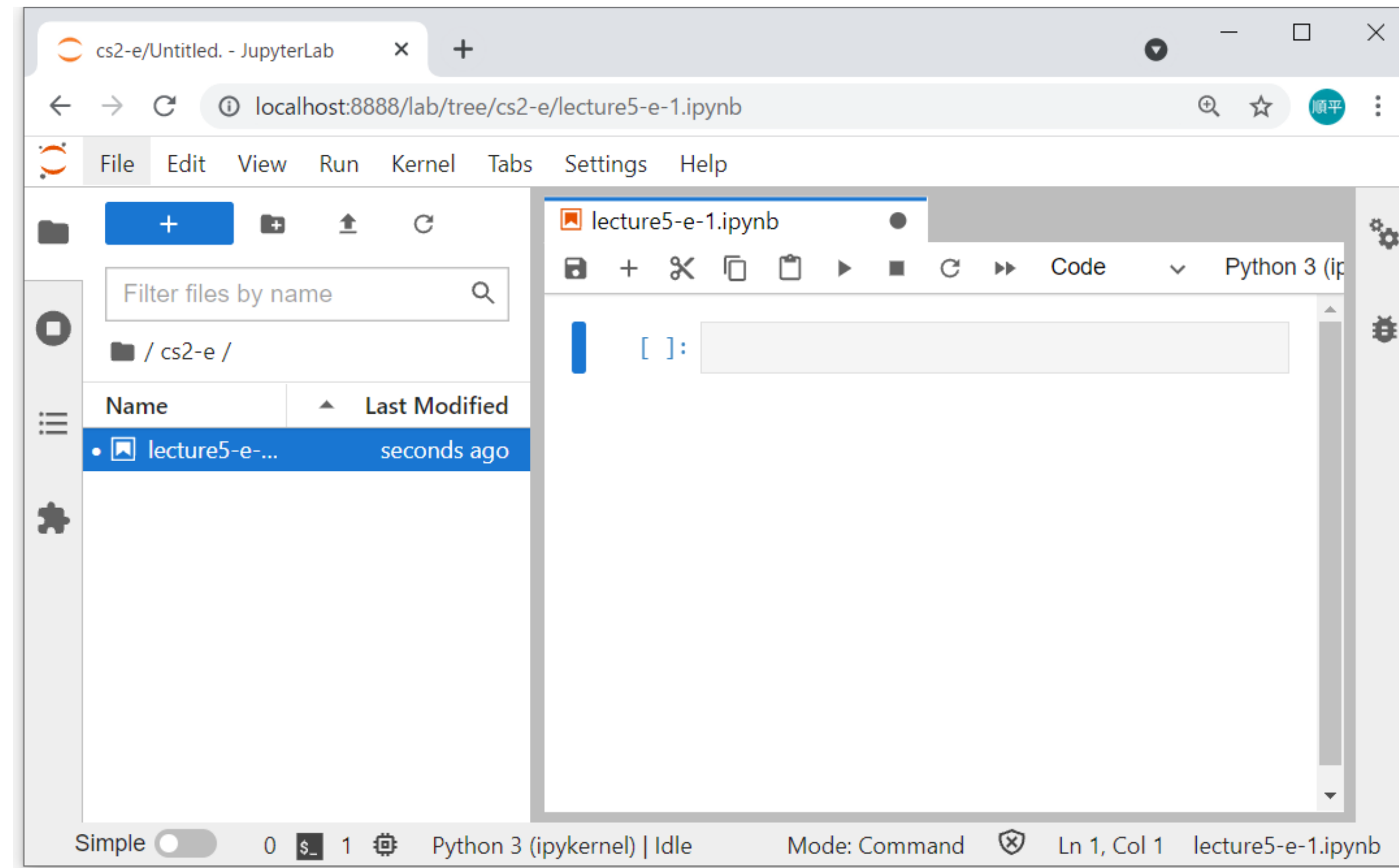
簡単な待ち行列シミュレーションをしてみよう

- スーパーのレジの待ち行列 (cashier) に複数のお客さん (customer) が訪れて、お会計をして帰る状況をシミュレーションしてみましょう。



jupyter labを開いてください

- cs2-e フォルダの中に、**lecture5-e-1.ipynb** を作成してください。

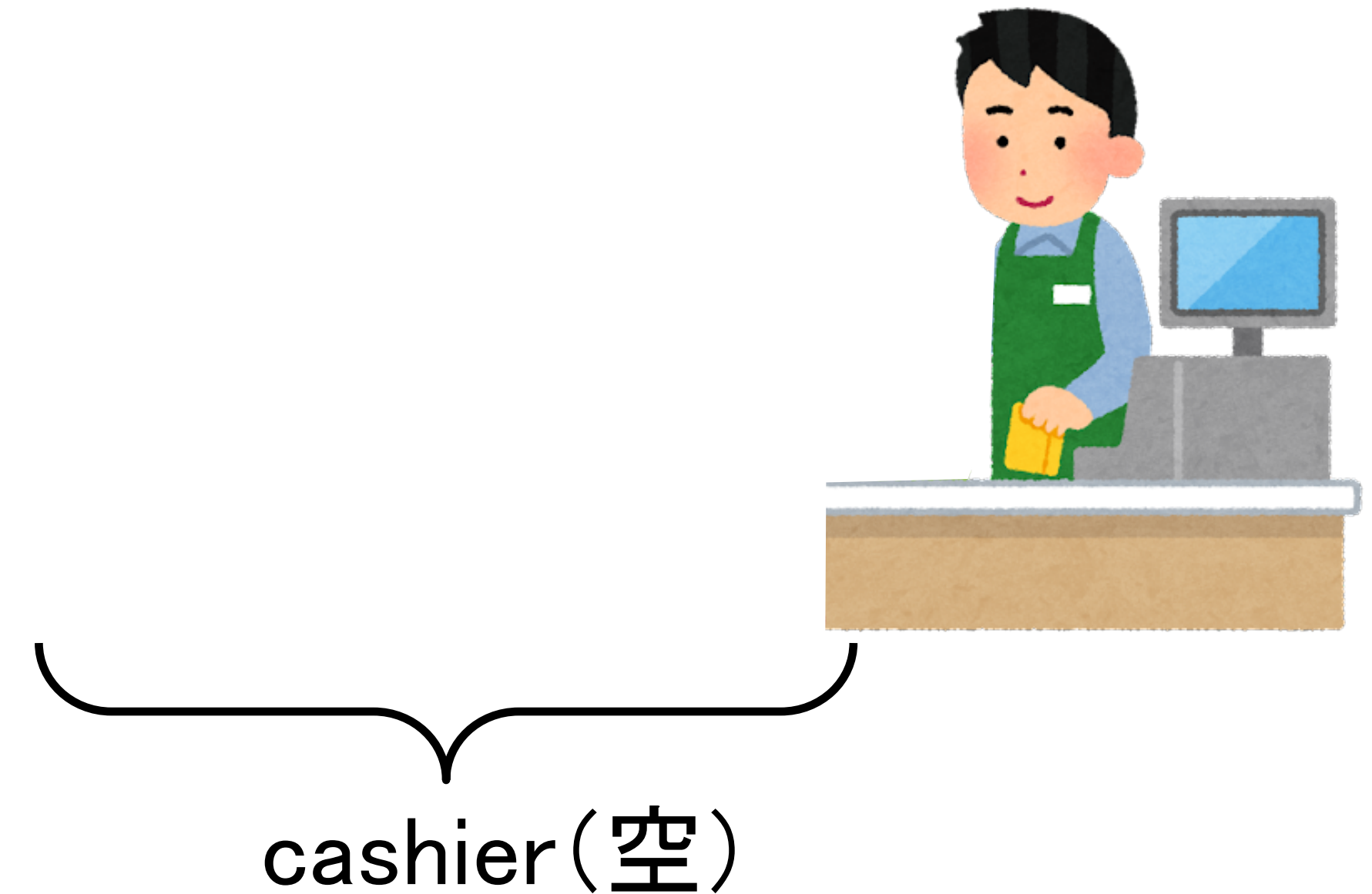


- 以下、出てくるコードは各セルに打ち込んで実行してみてください。

レジの待ち行列シミュレーション(1/4)

- 手順1: レジを作成する

```
from collections import deque  
  
# レジの待ち行列(空)を作成  
cashier = deque()
```



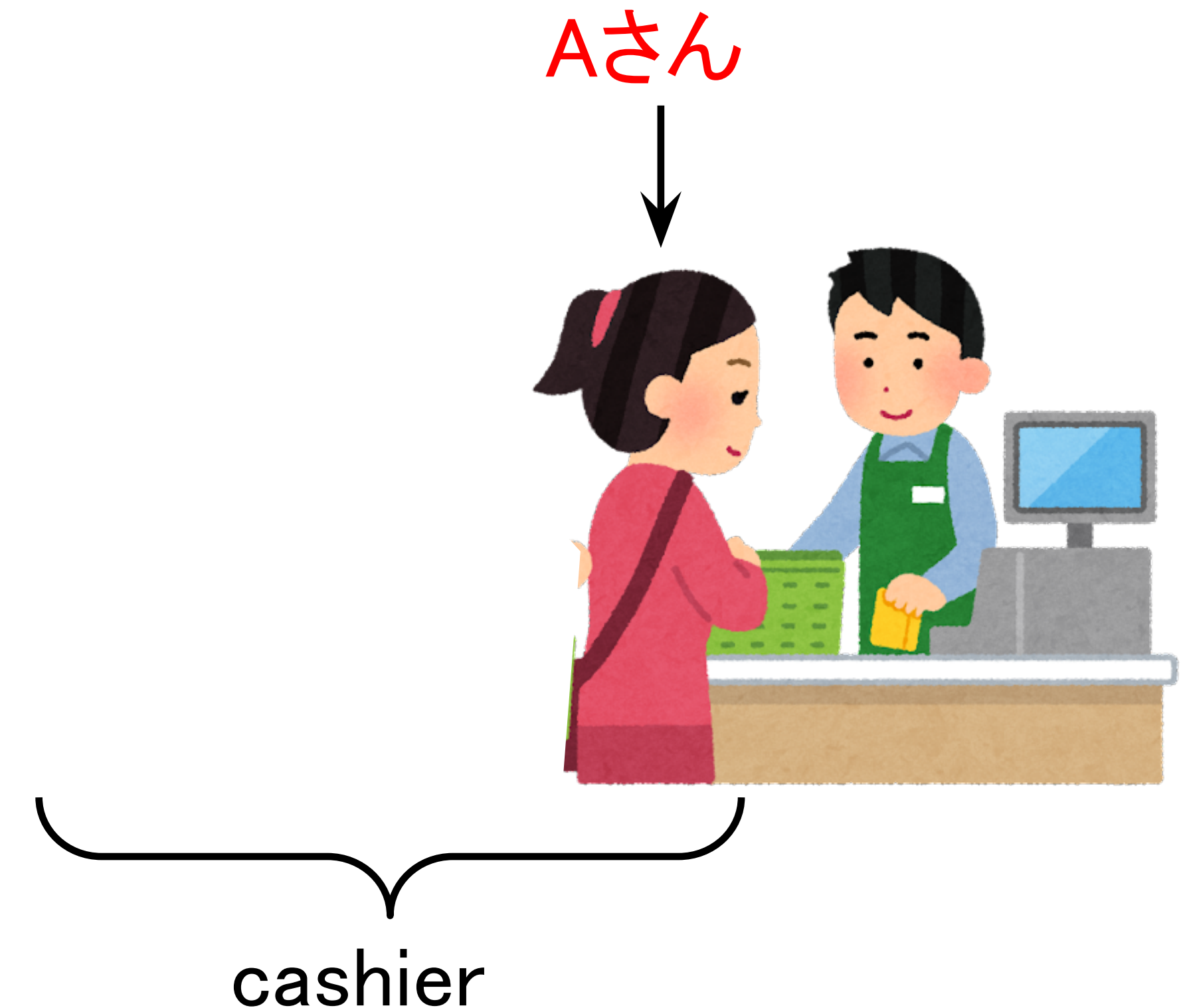
レジの待ち行列シミュレーション (2/4)

- 手順2: レジにお客さんを追加する

```
from collections import deque

# レジの待ち行列(空)を作成
cashier = deque()

# お客さんをレジに並ばせる
customer = "A"    # Aさん
print(customer, "さんがレジに並びました。")
cashier.append( customer )
```



レジの待ち行列シミュレーション (3/4)

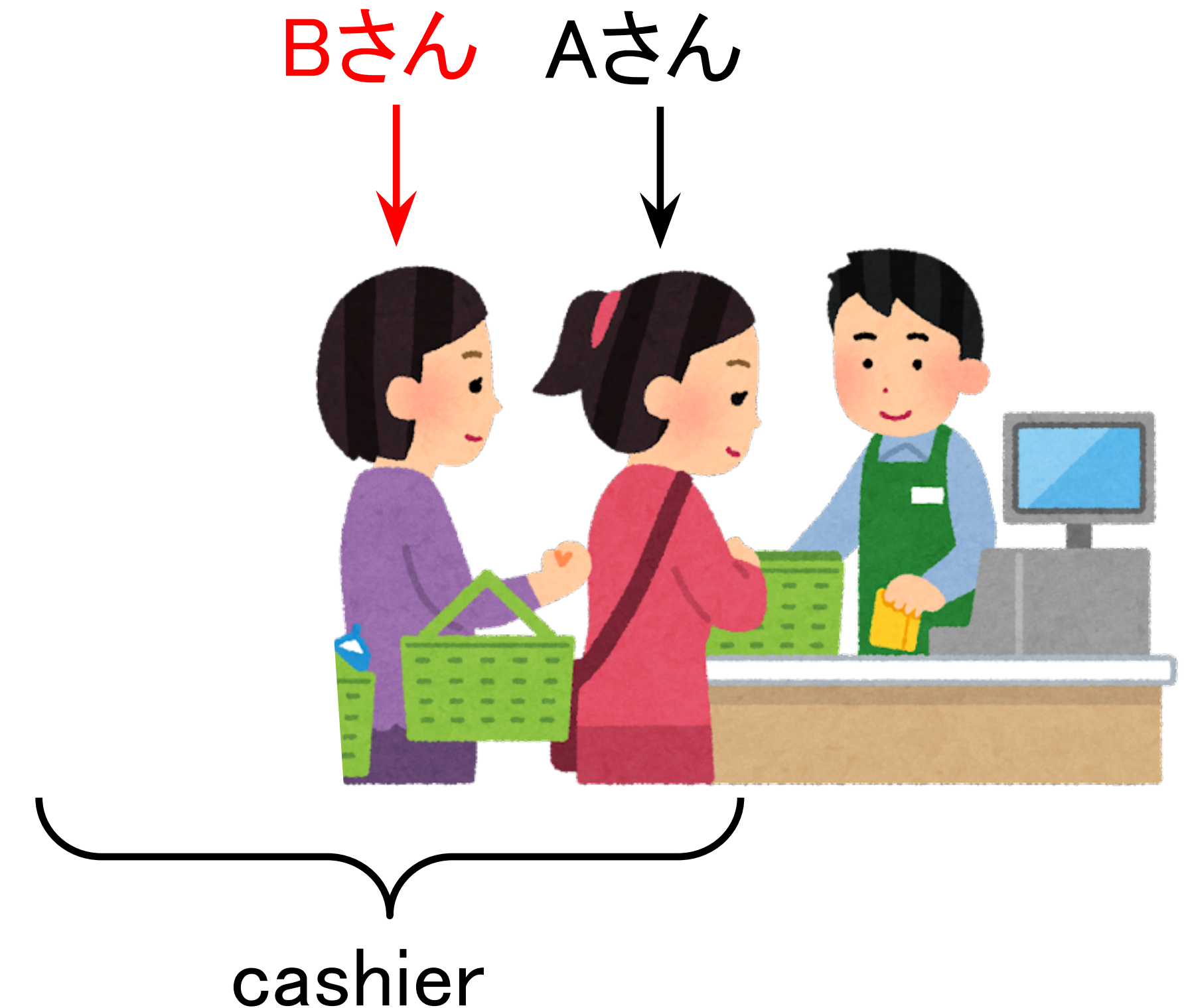
- 手順3: レジにお客さんをもう一人追加する

```
from collections import deque

# レジの待ち行列(空)を作成
cashier = deque()

# お客さんをレジに並ばせる
customer = "A"    # Aさん
print(customer, "さんがレジに並びました。")
cashier.append( customer )

customer = "B"    # Bさん
print(customer, "さんがレジに並びました。")
cashier.append( customer )
```



レジの待ち行列シミュレーション(4/4)

- 手順4: お客さんが会計を済ませる

(続き)

```
# 先頭のお客さん(Aさん)がお会計を済ませる
if len(cashier) > 0:
    customer = cashier.popleft()
    print(customer, "さんが会計を済ませました。")

# 先頭のお客さん(Bさん)がお会計を済ませる
if len(cashier) > 0:
    customer = cashier.popleft()
    print(customer, "さんが会計を済ませました。")
```

