

ソフトウェア演習Ⅲ〔課題 2：Python でクラス作成〕青野雅樹

以下の問題に対する Python プログラムを作成し、プログラム (kadai2.py) と実行結果 (kadai2-1.txt と kadai2-2.txt) をつけ、全体を ZIP にまとめて Moodle にアップロードせよ。締め切りは 10 月 20 日 (火) までとする。

(注) 以下のクラス内の関数で、self 引数は暗黙に含まれるものとして関数の引数には記載していないが、実際は self を第一引数としてクラス内の関数すべてに付与すること。また、メンバー変数(例 x)とは、クラス内で self.x のように定義されるものを指す。

(1) 商品クラス (クラス名=Goods)

メンバー変数

メンバー変数名	概要
name	商品名
price	定価 (消費税は含まない)

コンストラクタ (__init__)

引数の数	引数の順序	概要
2	(name, price)	self.name と self.price に代入

関数：

関数名	概要
printGoods	name と price をプリント

(2) 商品在庫管理クラス (クラス名=GoodsStock)

メンバー変数

メンバー変数名	概要
goods	商品クラスの変数
stock	在庫数

コンストラクタ (__init__)

引数の数	引数の順序	概要
2	(goods, stock)	それぞれ self(例 self.goods)に代入

関数

関数名	引数	戻り値型	概要
buyGoods	num	整数値	num 個購入要求, 購入できる数を返す。関数内では、stock(self.stock) の数を購入要求数に基づき変更する
printStock	なし	なし	goods に対する printGoods 関数を呼び出し、その商品の在庫数(stock) をプリントする

(3) 店のクラス (クラス名=Shop)

メンバー変数

メンバー変数名	型	概要
name	文字列	店の名前
telephone	文字列	電話番号
goodsStockList	GoodsStock のリスト	商品在庫クラスのリスト 初期値は None とする。 (setGoodsStockList 関数でセットする)

コンストラクタ (__init__)

引数の数	引数の順序	概要
2	(name, telephone)	goodsStockList は None で初期化

関数

関数名	引数	戻り値 型	概要
buyRequest	goods, num	整数 値	Goods を num 個、買いたい需要に対して、在庫をチェックし買えた商品数を返す。具体的には goods.name にマッチした商品に対して、goodsStockList にある GoodsStock クラスの関数 buyGoods(num) を返す。マッチするものがなければ 0 を返す。
setGoodsStockList	list	なし	引数の list で指定された GoodsStock のリストを対応するメンバー変数にセットする
printShop	なし	なし	店 (名前と電話番号) をプリント後、goodsStockList をたどり、GoodsStock クラスの printStock を呼び出す

[11] (1), (2), (3) のクラスを (同じ kadai2.py の中で) 作成せよ。

- [2] `__main__`内では、以下の指示に従いプログラムせよ（乱数生成を8回行う）。
- [2-1] 出力の先頭に自分の名前、プログラムを実行した際の日付や、課題名を書き出せ。
- [2-2] "東京店"を電話番号"03-1234-5678"とし、"豊橋店"を電話番号"0532-44-9876"として、2つのShopオブジェクトを作成せよ。
- [2-3] 東京店の商品ならびに在庫数は以下のように決めるとする。
- ```
Goods("海洋深層水", 1200), 在庫数 N1 (N1 は 1~30 の整数 (乱数))
Goods("ビタミンC", 350), 在庫数 N2 (N2 は 1~30 の整数 (乱数))
Goods("アガリクス", 2000), 在庫数 N3 (N3 は 1~30 の整数 (乱数))
```
- ここで N1,N2,N3 を書き出し、商品リストをセットしたのち、東京店の `printShop` 関数を呼び出すこと。
- [2-4] 豊橋店の商品ならびに在庫数は以下のように決めるとする。
- ```
Goods("天然黒酢", 890), 在庫数 N4 (N4 は 1~20 の整数 (乱数))
Goods("ロイヤル葉緑素", 1600), 在庫数 N5 (N5 は 1~20 の整数 (乱数))
Goods("アロエはちみつ", 650), 在庫数 N6 (N6 は 1~20 の整数 (乱数))
```
- ここで N4,N5,N6 を書き出し、商品リストをセットしたのち、豊橋店の `printShop` 関数を呼び出すこと。
- [2-5] 自分が顧客と仮定し、東京店の3商品からランダムに1つ商品を選び、それを N7 個買う。一方、豊橋店の3商品からランダムに商品1つを選び、それを N8 個買うと仮定する。ただし、N7,N8 は 1~10 の間のランダムな個数(整数値)とする。このとき、N7,N8 を書き出し、2つの店での買い物を合計し、必要金額をプリントせよ。ただし、すべての商品の定価に10%の消費税がかかるとする。万一、在庫数より、多く買おうとしたときは、在庫までの数までの販売にとどめ、「在庫は xx だけで、在庫限り販売します。」と書き出すこと (xx は N1~N6 で決められた数)。これは、GoodsStock クラスの `buyGoods` 関数の中で行うこと。なお、出力は、購入希望数が在庫以内のケース(kadai2-1.txt)と、上記の在庫より多くの注文があったケース(kadai2-2.txt)の両方のケースを出力すること。乱数は、コメントとヒントにある方法で初期化し、実行するたびに値が変わるようにすること。

コメントとヒント: [min,max]の間の整数値をランダムに選ぶ関数は、たとえば、以下のようにして実現するとよい。

```
import random
def getRand(min, max): # [min,max]の間の整数乱数を発生
    t = random.random()
    return int(min+(max-min) * t)
```

また、乱数の初期値は、現在時刻を利用し、たとえば、以下のように宣言しておくこと。

```
from datetime import datetime
now = datetime.now()
random.seed(now)
```

【出力例 1】（在庫限り販売が発生しないとき）

青野雅樹, 011620

日付: 2020-10-03 13:20:51

内容: クラスの作成

N1 = 9 , N2 = 28 , N3 = 20 , N4 = 9 , N5 = 19 , N6 = 5

Shop: 東京店 03-1234-5678

Goods (商品): name = 海洋深層水 price = 1200

GoodsStock (在庫数): 9

Goods (商品): name = ビタミンC price = 350

GoodsStock (在庫数): 28

Goods (商品): name = アガリクス price = 2000

GoodsStock (在庫数): 20

Shop: 豊橋店 0532-44-9876

Goods (商品): name = 天然黒酢 price = 890

GoodsStock (在庫数): 9

Goods (商品): name = ロイヤル葉緑素 price = 1600

GoodsStock (在庫数): 19

Goods (商品): name = アロエはちみつ price = 650

GoodsStock (在庫数): 5

N7 = 2

商品: ビタミンC を 東京店 から 2 個、購入希望します

商品: ビタミンC は税込みで1個当たり 385 円です

商品: ビタミンC を 2 個、販売します

N8 = 7

商品: ロイヤル葉緑素 を 豊橋店 から 7 個、購入希望します

商品: ロイヤル葉緑素 は税込みで1個当たり 1760 円です

商品: ロイヤル葉緑素 を 7 個、販売します

合計金額: 13090 円

【出力例 2】（在庫限り販売が含まれるとき）

青野雅樹, 011620

日付: 2020-09-03 13:22:00

内容: クラスの作成

N1 = 4 , N2 = 15 , N3 = 1 , N4 = 17 , N5 = 12 , N6 = 9

Shop: 東京店 03-1234-5678

Goods (商品): name = 海洋深層水 price = 1200

GoodsStock (在庫数): 4

Goods (商品): name = ビタミンC price = 350

GoodsStock (在庫数): 15

Goods (商品): name = アガリクス price = 2000
GoodsStock (在庫数): 1

Shop: 豊橋店 0532-44-9876
Goods (商品): name = 天然黒酢 price = 890
GoodsStock (在庫数): 17
Goods (商品): name = ロイヤル葉緑素 price = 1600
GoodsStock (在庫数): 12
Goods (商品): name = アロエはちみつ price = 650
GoodsStock (在庫数): 9

N7 = 7
商品: 海洋深層水 を 東京店 から 7 個、購入希望します
商品: 海洋深層水 は税込みで1個当たり 1320 円です
GoodsStock: 在庫は 4 だけで、在庫限り販売します
商品: 海洋深層水 を 4 個、販売します

N8 = 3
商品: ロイヤル葉緑素 を 豊橋店 から 3 個、購入希望します
商品: ロイヤル葉緑素 は税込みで1個当たり 1760 円です
商品: ロイヤル葉緑素 を 3 個、販売します
合計金額: 10560 円