**AN6100 PROGRAMMING ESSENTIALS**

**Python Assignment**

Due: 27 July 2022 (Wednesday) 12 noon

## Worth: 20% to AN6100 final score.

Note: This assignment is an INDIVIDUAL assignment

***Assignment Objective :***

You are required to download the given .py file and provide all the necessary functions in order to support a drug dispensing system.

A template file is given to you. You should rename this file in the format A1\_[MatricNo or Name].py, work on this same file for submission. Please ensure that you include your name and admin number in the comment session as well. You must keep a copy of the final version of your submission and be prepared to provide it on request.

All the codes you provide must be original and MUST make use of what you have learnt in the course so far. The University treats plagiarism, collusion, theft of other students’ work and other forms of dishonesty in assessment seriously. Students shared out their answers or copied answers from others shall receive a zero score.

**Submission instructions:**

1. You should submit your assignment (only one .py file) via the NTU-learn blackboard site.

2. As this is a graded assignment, your instructor is not supposed to debug the program or give you any hint to your approach. Only clarifications on requirements are allowed.

**Please take note :**

Only the last submission before the due date will be graded, we will only provide qualitative feedback (if any) after that.

**Assignment background knowledge.**

Drug dispensing goes digital

<https://pharmafactz.com/pharmacy-calculations-formulations/>

Graphical user interface, website

Description automatically generated

<http://mca.gm/wp-content/uploads/2018/02/Guideline-for-Repackaging-and-Labelling-of-Medicines.pdf>

**Definition of repackaging**

Graphical user interface, text

Description automatically generated

**You are a junior coder in a software development company and the software team you are in is responsible to develop a drug dispensing system.**

**Your team leader knew that you are a python coder, he has prepared a .py start file for you to help to prepare some supporting functions for the entire system.**

**Some basic testing requirements are provided for you to test your codes but you should do more tests than provided to ensure the functions will work for all scenarios.**

**Content of the given .py file**

'''

Todo:

1. Rename this file, change in the format A1\_[TutorialGroup]\_[Your student matric number]\_[YourName], for example A1\_1\_G1418412A\_AiBianCheng.

2. You are required to complete the functions as per question, the instructions of each functions are written

as comment within the function, you are reminded to amend the return values whenever is required by the instructions

3. You are given some test data with command calling the function and the expected results in the main program

for your own testing

4. You will be graded based on the correct output with additional test cases that are prepared by the guider

5. You will also be graded on the quality of code, in terms of readability (so please insert comment when required),

efficiency and maintainability.

'''

'''

Background of this assignment : Drug dispensing goes digital

https://pharmafactz.com/pharmacy-calculations-formulations/

&

http://mca.gm/wp-content/uploads/2018/02/Guideline-for-Repackaging-and-Labelling-of-Medicines.pdf

Some definitions of terms used adopted from the pdf above :

11. DEFINITION OF TERMS

The definitions provided below apply to the words and phrases used in the MCA

guidelines. Although an effort has been made to use standard definitions as far as

possible, minor alterations have been made in some cases.

Batch (or Lot):

A defined quantity of products processed in a one process or series of processes

so that it is expected to be homogeneous.

Batch number (or Lot number):

A distinctive combination of numbers and/or letters which specifically identifies a

batch

Dispensing:

Refers to the process of preparing and giving a medicine to a named person or

animal on the basis of a prescription

Labelling:

Information on the primary or secondary packaging of a medicine

Secondary container/packaging (or Outer packaging):

Container that is not in direct contact with the medicine

Shelf-life:

The period of time during which a medicine or related product, if stored correctly,

is expected to comply with the specifications as determined by stability studies on

a number of batches of the product; the shelf-life is used to establish the expiry

date of each batch

&

via https://medical-dictionary.thefreedictionary.com/repackage

the definition of repackaging is as follows

The transfer of specified doses of a medication from a manufacturer's bulk container to smaller containers used by

patients and/or dispensing institutions.

'''

#Question 1

def balancePart(partlist, totalparts=100):

""" Question 2 instructions

this function accept an integer list call partlist and an

optional parameter totalparts

this function will add up all the integer values in partlist

and get it substracted from totalparts.

The answer will be returned as an integer

\*You are not required to validate if sum of partlist will exceed

totalparts or not

\*You may also assume all members of partlist are integers

"""

return None

#Question 2

def chalk\_mixture(preparedToAmount):

""" Question 2 instructions

You are required to complete this function to

provide the recipe in preparing chalk mixture.

This function suppose to accept the prepareTOAmount (in ml)

from the caller

and return the component list given, amount list after adjusted

according to the preparedToAmount

and the unit list given

WORK EXAMPLE for QUESTION 2

A pharmacist has been handed a prescription that asks for 200mL of chalk mixture, pediatric BP.

The formula is listed as follows:

-Chalk 20g

-Tragacanth powder 2g

-Concentrated cinnamon water 4mL

-Syrup 100mL

-Double strength chloroform water 500mL

-Water for preparation to 1000mL

The recipe we have prepares for 1,000mL, but the prescription asks for 200mL.

As a result, we must divide each ingredient in the formula by 5 – meaning the formula is now prepared to 200mL

rather than 1,000mL.

"""

component = ["Chalk", "Tragacanth powder","Concentrated cinnamon water","Syrup","Double strength chloroform water","Water for preparation to"]

amount = [20, 2,4,100,500,1000]

unit = ["g", "g","mL","mL","mL","mL"]

return None

#Question 3

def IMS\_BP(recipeItemList, recipeItemPart, requiredVolume):

""" Question 3 instructions

You are required to complete this function to

provide the recipe in preparing IMS BP.

This function suppose to accept the recipeItemList,

recipeItemPart (as parts) and requiredVolume

from the caller

and return a list containing how much of each ItemPart

should be used to prepare the requiredVolume

WORK EXAMPLE for question 3

Consider the standard for industrial methylated spirits (IMS) BP – which states that ingredients should be in the

ratio 95 parts spirit to 5 parts wood naphtha. In IMS, both ingredients are liquids so the parts must be v/v.

How much of each ingredient is needed to produce 300L?

First, we need to add the parts – (95 + 5 = 100) – and relate this to the 300L required.

The multiplication difference here is 3 (300L divided by 100 parts – we now have a relationship between both:

95 parts + 5 parts = 100 parts

300L/100 Parts = 3L per part

5 parts = 15L

95 parts = 285L

We need 15L of wood naphtha and 285L of spirit to produce 300L of (IMS) BP.

\*You may assume that the length of both input lists are of the same length

In the example above the pass in parameters should contain

["wood naphtha","spirit"] for recipeItemList

[95,5]for recipeItemPart and the total may NOT sum up to 100.

"""

return None

#Question 4

def WhiteSoftParaffinTo100(weightToProduce,items ,pc):

""" Question 4 instructions

You are required to complete this function to

provide the recipe in preparing white Soft Paraffin mixture.

This function suppose to accept the weightToProduce (in g)

from the caller

and return the list containing the percentage of component given,

component list given and the list of weight amount used for each component.

WORK EXAMPLE for question 4

Using the following formula, calculate the amount of each ingredient needed to produce 75g.

Sulfur – 6%

Salicylic acid – 4%

White soft paraffin to 100% , i.e. (100%-6%-4%=90%)

First, we need to recognize that 75g is the equivalent of “to 100%”.

6 percent of 75g is 4.5g.

By the same method, there is 3g of salicyclic and,

by subtraction both values from 75g,

there must be 92.5 grams of white soft paraffin

"""

return None

#Question 5

def Package(quantity, baseQty):

""" Question 5 instructions

This function help to work out the production qty and loose balance whenever there is a packaging

WORK EXAMPLE for question 5

if the quantity required is 30,

and the packaging is in dozens (12), hence the baseQty = 12

so when calling this function,

this function should return a tuple 2,6

"""

return None

#Question 6

def askLotsLoose(prompt):

""" Question 6 instructions

This function accepts 1 parameter, prompt

The function will prompt and expecting a input string

containing 2 integers separated by ,

example 3,5

This function will be graded also besides functionalities,

able to continuing prompting the user until a correct

2 integers separated by , is entered.

The return from this function are the 2 integer numbers return as

an integer (not string) tuple

"""

return None

#Question 7

def Add\_Balance(HoldingLots,HoldingLooseQty, addLotQty, addLooseQty =0):

""" Question 7 instructions

This function accepts 4 parameters,

the HoldingLots and HoldingLooseQty refers to the current balance

stock quantities;

the addLotQty and addLooseQty refers to the new stock quanties to be

added.

This function adds up the Lots and Loose quantities and

return the new balances accordingly as a tuple of Lot quantity and

loose quantity

"""

return None

#Question 8

def Dispense\_medicine(holdingLots, holdingLoose, issuingLotsQty, issuingLooseQty, QtyPerLot):

""" Question 8 instructions

This function accepts 5 parameters,

the HoldingLots and HoldingLooseQty refers to the current balance

stock quantities;

the issuingLotQty and issuingLooseQty refers to the new stock quanties to be

dispense;

QtyPerLots is the packaging base quantity for this item

When this function is called, if the holdingLoose quantity is

enough to dispense the issuingLoose quantity, they are to be used

before you 'break' a new Lot.

This function will return the new balances accordingly as a tuple

of Lot quantity and loose quantity

"""

return None

if \_\_name\_\_ == "\_\_main\_\_":

'''

Test codes with expected results are provided for you to validate

the functions you have completed above

Please note that you are required to perform as many tests as possible

to avoid any logical or run time error.

'''

#-------------------------------------------------------------------

"""

#Question 1 testing

percentlist = [23,14,45]

print("If is by 100%, the remaining is", balancePart(percentlist))

print("If is of total of 70 parts, the remaining is", balancePart(percentlist,70))

'''Expected Output

If is by 100%, the remaining is 18

If is of total of 70 parts, the remaining is -12

'''

"""

#-------------------------------------------------------------------

"""

#Question 2 testing

print("Q2 tester")

howMuch = float(input("Hi, pharmacist,the prescription asks for ? mL of chalk mixture, pediatric BP. "))

if howMuch >0 :

comp, amt, unit= chalk\_mixture(howMuch)

for c,a,u in zip(comp,amt,unit):

print(c,a,u)

'''Expected Output :

Hi, pharmacist,the prescription asks for ? mL of chalk mixture, pediatric BP. 600

Chalk 12.0 g

Tragacanth powder 1.2 g

Concentrated cinnamon water 2.4 mL

Syrup 60.0 mL

Double strength chloroform water 300.0 mL

Water for preparation to 600.0 mL

'''

"""

#-------------------------------------------------------------------

"""

#Question 3 testing

print("Q3 tester")

howMuch = float(input("Hi, pharmacist,the prescription asks for ? mL of IMS\_BP. "))

if howMuch >0 :

comp =["wood naphtha","spirit"]

parts = [95,5]

amt = IMS\_BP(comp,parts,howMuch)

for c,a in zip(comp,amt):

print(a,"mL of",c)

'''Expected Output

Hi, pharmacist,the prescription asks for ? mL of IMS\_BP. 300

285.0 mL of wood naphtha

15.0 mL of spirit

'''

"""

#-------------------------------------------------------------------

"""

#Question 4 testing

print("Q4 tester")

howMuch = float(input("Hi, pharmacist,the prescription asks for ? g of White soft paraffin mixture. "))

if howMuch >0 :

ingredients = ["Sulfur","Salicylic acid"]

percentage = [6,4]

percent, comp, weight= WhiteSoftParaffinTo100(howMuch,ingredients , percentage)

for pc,c,w in zip(percent,comp,weight):

print(pc,f'percent of {c} is', w,'g')

'''Expected Output

Hi, pharmacist,the prescription asks for ? g of White soft paraffin mixture. 75

6 percent of Sulfur is 4.5 g

4 percent of Salicylic acid is 3.0 g

90 percent of soft paraffin is 67.5 g

'''

"""

#-------------------------------------------------------------------

"""

#Question 5 testing

print("Q5 tester")

howMuch = int(input("Hi, pharmacist, what is the quantity prepare for repackaging? "))

baseQty = int(input("What is the packaging base quantity?"))

lots, loose = Package(howMuch,baseQty)

print(lots, "lots", loose , "loose")

'''Expected Output

Hi, pharmacist, what is the quantity prepare for repackaging? 100

What is the packaging base quantity?24

4 lots 4 loose

'''

"""

#-------------------------------------------------------------------

"""

#Question 6 testing

lots, loose = askLotsLoose("Please enter quantity[lot,loose] ")

print("You requested for", lots, "lots", loose, "loose")

'''Expected Output

Please enter quantity[lot,loose] 1,2

You requested for 1 lots 2 loose

'''

"""

#-------------------------------------------------------------------

"""

#Question 7 testing

print("Q7 tester")

holdingLots, holdingLoose = askLotsLoose("Please enter current balance[lot,loose] ")

newLots, newLoose = askLotsLoose("Please enter new stock quantity[lot,loose] ")

print("new balance (Lots, Loose) as ", Add\_Balance(holdingLots,holdingLoose,newLots,newLoose))

'''Expected output

Q7 tester

Please enter current balance[lot,loose] 10,12

Please enter new stock quantity[lot,loose] 22,24

new balance (Lots, Loose) as (32, 36)

'''

"""

#-------------------------------------------------------------------

"""

#Question 8 testing

print("Q8 tester")

holdingLots, holdingLoose = askLotsLoose("Please enter current balance[lot,loose] ")

while True:

giveLots, giveLoose = askLotsLoose("Please enter dispense[lot,loose] ")

holdingLots, holdingLoose = Dispense\_medicine(holdingLots, holdingLoose,giveLots, giveLoose ,12)

print("New balance", holdingLots, "lots", holdingLoose, "loose")

if input("Enter -1 to end") =='-1':

break

'''Expected output

Q8 tester

Please enter current balance[lot,loose] 12

Error, please enter again

Please enter current balance[lot,loose] 12,2

Please enter dispense[lot,loose] 2,12

New balance 9 lots 2 loose

Enter -1 to end1

Please enter dispense[lot,loose] 1,2

New balance 8 lots 0 loose

Enter -1 to end

Please enter dispense[lot,loose] 7,11

New balance 0 lots 1 loose

Enter -1 to end

Please enter dispense[lot,loose] 0,1

New balance 0 lots 0 loose

Enter -1 to end-1

'''

"""