COMPSCI 120 E: Introduction To Computer Science

Programmer: Samuel Bartholomew

Assignment: Programming Assignment #2

Professor: Dr. Stanelli

Step 1: Problem

Write a program, customized to the user, which performs two different conversion calculations between the three domain units. (ounces / milliliters / tablespoons)

Step 2: Solution

Strings

userName = ""

userValue = "" #initilaze as a string but will be converted into a float after spaces removed and checked.

userConvertChoice = ""

userConvertTo = ""

userConvertString = ""

userConvertToString = ""

convertString = ""

stringFilter = "123456789"

Floats

convertValue = 0.0

mLConversionRate = 29.574

tbsConversionRate = 14.787

Lists

stringToList = []

**CalculateValue(userConvertTo, userConvertChoice, userValue)**

Step 1: if userConvertTo is 0 set userConvertToString to “Ounce(s)”

Step 2: if userValue == 0 set convertValue to 0

Step 3: if userConvertTo is 0 and userConvertChoice is 0 and userValue is not 0 set convertValue to userValue

Step 4: if userConvertTo is 0 and userConvertChoice is 1 and userValue is not 0 set convertValue to userValue / mLConversionRate

Step 5: if userConvertTo is 0 and userConvertChoice is 2 and userValue is not 0 set convertValue to userValue / 2

Step 6: if userConvertTo is 1set userConvertToString to “milliliter(s)”

Step 7: if userValue == 0 set convertValue to 0

Step 8: if userConvertTo is 1 and userConvertChoice is 0 and userValue is not 0 set convertValue to userValue \* mlConversionRate

Step 9: if userConvertTo is 1 and userConvertChoice is 1 and userValue is not 0 set convertValue to userValue

Step 10: if userConvertTo is 1 and userConvertChoice is 2 and userValue is not 0 set convertValue to userValue \* tbsConversionRate

Step 11: if userConvertTo is 2 set userConvertToString to “Tablespoon(s)”

Step 12: if userValue == 0 set convertValue to 0

Step 13: if userConvertTo is 2 and userConvertChoice is 0 and userValue is not 0 set convertValue to userValue \* 2

Step 14: if userConvertTo is 2 and userConvertChoice is 1 and userValue is not 0 set convertValue to userValue / tbsConversionRate

Step 15: if userConvertTo is 2 and userConvertChoice is 2 and userValue is not 0 set convertValue to userValue

Step 16: return convertValue

**takeUserValue()**  
Step 1: Take Input for userValue

Step 2: take the userValue string and replace all spaces with null / empty.

Step 3: Call a lambda function with the argument userValue, this will iterate through the string character by character and create a list in order of stringToList, also filtering out everything outside of stringFilter

Step 4: Set userValue to a null string + stringToList, taking each entry as a character for the string

Step 5: Try userValue = float(userValue)

Step 6: if Exception ValueError occurs print the value taken was not a number you may try again

Step 7: if exception Value error occurs also call takeUserValue setting userValue to equal it so the point from the main function does not get set to just “” aka null

Step 8 return userValue

**convertUnit(userName)**  
Step 1: Get the user’s Value through takeUserValue function

Step 2: Print the choices of conversion with the correlating numbers.

Step 3: Take input from user asking for which choice they would like to store that into userConvertChoice

Step 4: Take userConvertChoice and replace all spaces “ “ with “” null

Step 5: check to see if the first character of userConvertChoice aka userConvertChoice[:1] is equal to 0 or 1 or 2  
Step 6: If so continue and do nothing

Step 7: If not tell the user there has been an error and say what the error is

Step 8: If not now exit the program

Step 9: if userConvertChoice is 0 set userConvertString to “Ounce(s)”

Step 10: if userConvertChoice is 1 set userConvertString to “Milliliter(s)”

Step 11: if userConvertChoice is 2set userConvertString to “Tablespoon(s)”

Step 12: set convertValue and userConvertToString to calculateValue(userConvertTo, userConvertChoice, userValue)

Step 13: Print the Hello userName, userValue userConvertString is equal to convertValue userConvertToString

**START**

Step 1: print the program’s purpose

Step 2: Get the userName from the user with an input function

Step 3: Enter a For loop with 2 iterations

Step 4: Print the conversation number + the iteration number

Step 5: Call convertUnit with the argument userName inside.

Step 6: Complete the for loop for the second iteration

Step 7: Print the programmer’s name

Step 8: end the program

Step 3: Formal Coding

https://onlinegdb.com/cTIumRIANc  
Step 4: Testing  
Values to test  
Case 1: Samuel, 0, 12, 1, 2 this should return 12 milliners to 0.8 TablespoonsA black screen with white text

Description automatically generated

Case 2: Robert Smith, 24, 2, 0 this should return 24 tablespoons to 12 ouncesA screenshot of a computer program

Description automatically generated

Case 3 Samuel Full 201, 0, 1, 12, 0, 1, 2 this should return the full project working along with 201 ounces to 5944.28 Milliliters in Conversion #1 and in conversion#2 It should return 0 milliliters to 0 tablespoons and print the programmers name at the end A screenshot of a computer program

Description automatically generated

Step 5: Documentation / User guide

User Guide  
The user will be entering 7 pieces of data in full, in order of their Name, What amount of liquid, the units, the units to convert to, what amount of liquid, the units, the units to convert to.

This will then be calculated based on what the user inputs and what conversions they want done.

Then reply to the user what is the results

Documentation

When writing this I learned about functions, for loops, try except statements, and if else statements. This program will be able to do 3-way conversion of Ounces Milliliters and Tablespoons. You can do it any way possible including ounces to ounces etc. Which will feel freer to the user. The conversions are stored into a float variable to provide the exact conversion.