Thursday Homework – Conversion Calculator

**Write a script or notebook that converts a given value from its original unit to a different unit**

You can either do the work in a new Jupyter Notebook or try to write a script.

In today's folder, there is a file called conversionMeasures.csv. It contains lines of data. Each line has three pieces of data, separated by commas: unit 1, conversion factor, unit 2.

The conversion equation for all of these lines is unit 1 x conversion factor = unit 2.

Your script should be able to convert the following test samples:

test\_unit = "pint"

test\_value = 2.5

final\_unit = "mL"

test\_unit = "cubic foot"

test\_value = 30

final\_unit = "liter"

test\_unit = "slug"

test\_value = "4.8" Yes, you should write your code to handle values that are entered as strings

final\_unit = "pound"

Your script should:

* store the conversion data from the csv file in some object (list of lists, dictionary of dictionaries, etc.)
  + open the file in read mode
  + save as a list of lines
  + exit the with/as statement
  + create an empty list
  + loop through the lines from the file
  + for each line, remove the trailing new line character and split the line on the commas, which will make a list
  + append the list to the empty list you made above
* provide a way to find the correct conversion factor from your data object
  + loop though the list of lists
  + if the test\_unit is equal to the item in the first position of the list and the final\_unit is equal to the third item in the list, save the second item in the list and break the loop
* include a function to convert between units
  + the function should take the test\_value and conversion factor as arguments and return the new value
* call the function with your test\_value and conversion factor
* print out a full sentence response with the final answer
* anticipate some errors (see below)
* run your script on the provided test examples

Errors to anticipate:

1. Someone might give the initial value as a string instead of a float/integer.
2. Someone might request a final unit that is not in your data – your script should print out an error message. Here’s a sample to test for this error:

test\_unit = "slug"

test\_value = 27.0

final\_unit = "snail"

Tips:

* You might write out the steps you need to complete each task before you start coding
* Take it one step at a time
* If you are stuck, there is another version of this document that contains more detailed steps (thursdayHWhelp)

BONUS CHALLENGES

1. Someone might give a test or final unit that has different capitalization than how it is presented in the csv file. Edit your script so that it can still process this sample:

test\_unit = "KM/H"

test\_value = 8.4

final\_unit = m/Sec

1. In the csv file, not all the units are included on both sides of the conversion factor. Someone might give you a test unit from the right side of the factor and ask you to convert it to the unit on the left side, which would require division instead of multiplication. Edit your script so that it can process this sample:

test\_unit = "ergs"

test\_value = 8.4

final\_unit = "joule"

1. As a special challenge, there’s a function called input() that can collect data from the user of your script in real time. Here’s a link to a website that works through the input() function (top half of the page only – stop before the Tkinter section): <https://datatofish.com/input-function-python/>. Try to use the input() function to collect the test\_unit, test\_value, and final\_unit values.