Individual Project Milestone Specifications

A Simple Unit Conversion Program

In the Milestone of your Individual Project, You will be starting the unit conversion program project which is the major individual assessment task for the first half of this semester.

For this project begin by creating a **script** that converts a temperature from **Fahrenheit to Celsius**.

Test that your program outputs the correct values by using the following **small set of test data** to compare with your results:

Sample test data				
$100^o\,F = 37.78^o\,C$	$0^o\,F = -17.78^o\,C$	$32^o\,F = 0^o\,C$	$-40^o\,F = -40^o\,C$	$55^o\,F = 12.78^o\,C$

Now write some more code that converts from Celsius to Fahrenheit. Repeat this for all the following unit-types, use MATLAB sections (%%) to create stand-alone conversion.

Type	To imperial	To metric
Temperature		
${}^oC \longleftrightarrow {}^oF$	$T_{oF} = \left(T_{oC} \times \frac{9}{5}\right) + 32$	$T_{oC} = (T_{oF} - 32) \times \frac{5}{9}$
Length & Distance		
cm $\longleftrightarrow$ inch	$L_{\text{inch}} = \frac{L_{cm}}{2.54}$	$L_{\text{cm}} = L_{\text{inch}} \times 2.54$
$m \longleftrightarrow$ feet	$L_{\text{ft}} = \frac{L_m}{0.3048}$	$L_m = L_{\text{ft}} \times 0.3048$
km $\longleftrightarrow$ miles	$D_{\text{mi}} = \frac{D_{\text{km}}}{1.609344}$	$D_{\text{km}} = D_{\text{mi}} \times 1.609344$
Mass		
grams $\longleftrightarrow$ ounces	$m_{\text{oz}} = \frac{m_{gr}}{28.3495}$	$m_{\text{gr}} = m_{oz} \times 28.3495$
kg $\longleftrightarrow$ pounds	$m_{\text{lb}} = \frac{m_{kg}}{0.4536}$	$m_{\text{kg}} = m_{lb} \times 0.4536$
tonne (met) $\longleftrightarrow$ ton (imp)	$m_{\text{ton}} = \frac{m_{tonne}}{1.016}$	$m_{\text{tonne}} = m_{ton} \times 1.016$

Once you have the basic conversions all working add a **text-based user-interface** to each of the sections you've created that allows the user to **input the number** they want converted onto the Command Window.

Your interface must contain **clear and appropriate instructions** to the user on how they should input the number and should **clearly show the result,** so it is neatly formatted and **easily interpreted** by the user.

To do this, use a combination of the functions in the table below:

Function	Description
Input function(s):	

`input()`	Obtains user input from the Command Window and stores that input as a workspace variable.
	
	Syntax example:
	`x = input('Enter a number: ') %Prompts user to enter a number onto the command line`
	`x = input('Enter a string: ','s') %Prompts user to enter a string onto the command line`

Output function(s):	

`disp()`	Displays a variable or a string on the command window.
	
	Syntax example:
	`disp('Hello World!') %Displays Hello World! On the command window.`
	`disp(x) %Displays the contents of the variable x on the command window.`

`fprintf()`	Displays formatted data on the command window. **Syntax example:** `fprintf('The answer is: %1.3f', x) %Displays a floating point number (x) formatted to display to a three decimal place precision (e.g. %1.3f is the formatting tag). This will be preceded by the text 'The answer is: '`

Due date and further details on this task:

The submission for Individual Project Milestone is due at the end of Week 4, Sunday, 24h of July before 23:59pm in the form of ONE file: MATLAB file – m.file.

You are required to submit your MATLAB script file in .m file format for your tutor to test for functionality.

Individual Project Milestone should consist of, You need to do the following:

- Design and implement a Basic Unit Converter that converts numbers that have been entered by the user onto the Command Window (using the input( ) function).
- Your script should contain code that converts all the units given on the previous page to and from their imperial and metric types (i.e. both directions). These conversions should take place in individual MATLAB sections created using the %% sign.
- The output must be displayed neatly on the Command Window with an informative message to the user.
- There should be **NO conditional statements, loops or user-defined functions** at this point; these will be included in Individual Project Report.
- You also **must NOT USE the in-built converter functions** (including but not limited to: convpres(), convtemp() and convforce())

Good Luck!

Please Note:

- **It is an Individual Task, NOT completed with your practical Team members or other students.**
- **Plagiarism will result in harsh penalties ALL cases will be reported to the Dean of SSET and misconduct charges will be applied!**