The Ideal Gas Law with Alternate Units

# Abstract

Allow the user to enter alternate units for pressure and temperature. Then convert those inputs using a unit’s library making the conversion simple, fast, and accurate.

# Problem Statement

Many times we will get different units for a standard problem, but the units need to be converted to the correct unit before calculation. By using Pint we can convert any unit into another one rather simple. The equation for the ideal gas law and variables is listed below.

# Methodology

Utilizing Python and Pint the conversion of units are quick and painless. The volume is what we are calculating for, the minimum we need is pressure and temperature to calculate the volume. We need to be able to take the input values and convert them to the correct unit i.e. Kelvin(temperature) and Atmosphere(pressure). The program will ask you to run the assignment values or to enter values manually. If you choose the assignment values they will be printed in the terminal. For the manually entering data you will need to enter the value with a space and then the unit, i.e. 400 C. The following units you can use are for temperature: Celsius, Fahrenheit, and Kelvin. For pressure: Atmosphere (atm), inches of Mercury (inhg), and Pounds of Square Inch (psi).

'''

Chris Oakley

Arrays and Matrices

10/3/2020

EGN3214 - Assignment 7

Variables:

ureg = UnitRegistry for Pint

'''

from pint import UnitRegistry

ureg = UnitRegistry()

Q = ureg.Quantity

continue\_yn='y'

question = 'ask'

volume = (

['500 C', '1 atm'],

['100 F', '30 inHg'],

['500 K', '5 psi'],

)

def get\_temp(temp):

# Convert F to Kelvin

if temp.split()[1] == 'F':

temp = int(temp[:-2])

T = Q(temp, ureg.degF)

# Convert C to Kelvin

elif temp.split()[1] == 'C':

temp = int(temp[:-2])

T = Q(temp,ureg.degree\_Celsius)

# It is Kelvin

elif temp.split()[1] == 'K':

temp = int(temp[:-2])

T = Q(temp, ureg.kelvin)

else:

print('Please enter a valid temperature and unit!')

return T.to('kelvin')

def get\_pressure(pressure):

if pressure.split()[1] == 'PSI':

pressure = int(pressure[:-3])

P = pressure \* ureg.psi

# Convert C to Kelvin

elif pressure.split()[1] == 'INHG':

pressure = int(pressure[:-4])

P = pressure \* ureg.inHg

# It is Kelvin

elif pressure.split()[1] == 'ATM':

pressure = int(pressure[:-3])

P = pressure \* ureg.atm

# It is not Kelvin

else:

print('Please enter a valid temperature and unit!')

return P.to('atm')

def calculateVolume(P, T):

n = 1 \* ureg.mole

R = Q(0.0821, 'L\*atm/(mole\*K)') # Universal Gas Constant on atm.L/(mol.K)

V = (n\*R\*T)/P

return V

while continue\_yn=='y':

if question == 'ask':

question = str(input('Do you wish to (M)anually Calculate the Volume, or run the (A)ssignment Numbers (M or A)?').upper())

elif question == 'M':

temp = str(input('Enter Temperature with a space and the unit i.e. (C,F,K): ').upper())

pressure = str(input('Enter Pressure with a space and the unit i.e. (ATM, inHG, PSI): ').upper())

T = get\_temp(temp)

P = get\_pressure(pressure)

P.to('atm')

T.to('kelvin')

V = calculateVolume(P,T)

print()

print(f'The volume is: {V:.2f}')

print()

continue\_yn = input('Do you wish to continue? (Y or N)').lower()

elif question == 'A':

for items in volume:

T = get\_temp(items[0].upper())

P = get\_pressure(items[1].upper())

V = calculateVolume(P,T)

print(f'The Pressure is: {P:.2f}')

print(f'The Temperature is: {T:.2f}')

print('#'\*41)

print(f'The volume is: {V:.2f}')

print()

print()

question = 'ask'

elif question == 'Q':

break

else:

print('Not a valid choice, please choose M or A')

question = 'ask'

# Solution

The table below is a list of the assignment variables, the red values are the calculations.

Table : Assignment Variables

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Temperature** | **Pressure** | **Converted**  **Temperature (Kelvin)** | **Converted Pressure**  **(Atmosphere)** | **Volume**  **(Liters)** |
| 500 C | 1 ATM | **773.15** | **1** | **63.48** |
| 100 F | 30 in HG | **310.93** | **1** | **25.46** |
| 500 K | 5 PSI | **500** | **0.34** | **120.65** |

# Conclusion

Being able to take in different variables for common measurements allows a greater flexibility in computation of the results. By utilizing a unit’s library it allowed us to focus on other options in the script than making sure the conversion numbers were correct. Using Pint in higher math or chemistry were many units are utilized it makes calculating them a breeze.