```python
# -*- coding: utf-8 -*-
"""
Created on Tue Jan 15 16:26:30 2019

@author: Ryan Dalby
u0848407
ME EN 2450
HW1a

"""

#Imports
import numpy as np
import math
import matplotlib.pyplot as plt


#Define ODE for problem given
f = lambda t,y: 2 * (325/850) * (math.sin(t)**2) - (200*(1+y)**(3/2))/850


def plot_tank(h):
    '''Driver function to solve then plot solution given ODE from t=0 to t=10 and a given step size
    (note if certain non typical step sizes are given it is possible that the y value at a step may
    and as a result may get complex values for subsequent steps)'''
    w = eulers(f, 2, 0, 10, h)
    t = w[0]
    y = w[1]
    plt.plot(t, y)
    plt.xlabel("Time(s)")
    plt.ylabel("Water Level(m)")
    plt.suptitle("Water Level versus Time")
    plt.show()

def eulers(func, yInitial, tInitial, tFinal, h):
    '''General Euler's method implementation that takes a given first order ODE(dy/dt) func depende
    solves given an initial value (yInitial, tInitial) from tInitial to tFinal with step size h.
    Returns 2d NumPy array with [0] index being t values and [1] index being corresponding y soluti
    y = yInitial
    t = tInitial
    yAns = [yInitial]
    tAns = [tInitial]
    while(t < tFinal): #Will end once we are on tFinal or past it
        y = y + f(t,y) * h
        t += h
        yAns.append(y)
        tAns.append(t)
    return np.array([tAns, yAns])

plot_tank(.5)
```