```
"""
Created on Wed Oct 26 15:45:14 2016

@author: Haley
eulermethod.py
"""

# ta: initial value of t
# tb: final value of t
# ya: initial value of y
# n : number of steps
# Returns value of y at tb.


def func(t, y):
       return 3*y*(t**-1) + (t**-4)*(y**2)

#----------Euler's Method----------

def Euler(ta, tb, ya, n):
       h = (tb - ta) / float(n)
       t = ta
       y = ya
       for i in range(n):
       y += h * func(t, y)
       t += h

       #print 't value ' + str(t)
       #print 'y value ' + str(y)
       #print '--------'
       return y

#----------Runge-Kutta Method----------

def oneEuler(t, y, h):
       y += h * func(t, y)
       t += h
       return (t, y)

def RungeKutta(ta, tb, ya, n):
       h = (tb - ta) / float(n)
       t = ta
       y = ya
       for i in range(n):
       y += func(((t + oneEuler(t, y, h)[0])/2), ((y + oneEuler(t, y, h)[1])/2)) * h
       t += h

       #print 't value ' + str(t)
       #print 'y value ' + str(y)
       #print '--------'

       return y

#---With Time Step .5---
print '--With a Time Step of 0.5--'
```

```
ta = 1 #initial time
tb = 2.5 #final time
ya = .5 #initial y
n = 3 #number of steps

print "Euler's Method: " + str(Euler(ta, tb, ya, n))
print "Runge-Kutta Method: " + str(RungeKutta(ta, tb, ya, n))

print ''

#---With Time Step .25---
print '--With a Time Step of 0.25--'
ta = 1 #initial time
tb = 2.5 #final time
ya = .5 #initial y
n = 6 #number of steps

print "Euler's Method: " + str(Euler(ta, tb, ya, n))
print "Runge-Kutta Method: " + str(RungeKutta(ta, tb, ya, n))
```