```python
import numpy as np
import matplotlib.pyplot as plt
from eulers import solveSystemEulers
from RK4 import RK4

#derivative functions -----------------
def dydt(t, y, z):
    return -2*y + 4*np.e**(-t)

def dzdt(t, y, z):
    return -y*z**2/3
#-------------------------------------

#Finding solutions using eulers
y, z, tRange = solveSystemEulers( [dydt, dzdt] , 2, 4, 0, 1, .1)

#Finding solution using runge kutta
f = [dydt, dzdt]
y0 = [2, 4]
tRange, yR = RK4(0, y0, f, .1, 1 )

# Plotting the results
plt.figure()
plt.plot( tRange, y, label='Eulers')
plt.plot( tRange, yR[:,0], label='Runge-kutta')
plt.title('Y over time')
plt.ylabel('Y')
plt.xlabel('t')
plt.legend()
plt.figure()
plt.plot( tRange, z, label='Eulers')
plt.plot( tRange, yR[:,1], label='Runge-kutta')
plt.title('Z over time')
plt.ylabel('Z')
plt.xlabel('t')
plt.legend()
plt.show()
```