

```

# ===== #
# RESOLUTION D'UNE EQUATION DIFFERENTIELLE DU SECOND ORDRE : y'' = f(t,y,y'). #
# Le pendule amorti. #
# METHODE D'EULER. SCHEMA EXPLICITE. #
# Version du 05/12/2015. #
# ===== #

# Importations
# =====
from math import sin, pi
import matplotlib.pyplot as plt

# La fonction donnant la dérivée y'' en fonction de y, y' (notée dy) et de la
# variable de référence t.
# =====
def f(t,y,dy):
    global L, k
    return -9.81/L*sin(y)-k*dy

# Programme principal.
# =====

# Initialisations
# Paramètres
L, k = 0.5, 0.2
pas = 0.01
# Conditions initiales
t = [0.]
y = [pi/6]
dy = [0]

# Coeur du programme, le schéma numérique
while abs(dy[-1])>0.01 or abs(y[-1])>0.01:
    t.append(t[-1] + pas)
    y.append(y[-1] + pas * dy[-1])
    dy.append(dy[-1] + pas * f(t[-1],y[-1],dy[-1]))

# Affichage graphique
plt.clf()
# Premier graphique : diagramme des phases
plt.subplot(1,2,1)
plt.plot(y,dy,label = 'Vitesse en fonction de l\'angle',linewidth=1.5)
plt.legend(loc = 'upper center')
plt.xlabel('Angle')
plt.ylabel('Vitesse')
plt.grid()
# Deuxième graphique : l'angle en fonction du temps
plt.subplot(1,2,2)
plt.plot(t,y,label = 'Angle en fonction du temps',color='red',linewidth=1.5)
plt.legend(loc = 'upper center')
plt.xlabel('Temps')
plt.ylabel('Angle')
plt.grid()
# Affichage
plt.show()

# Fin du programme principal.
# =====

```