

```
"""
```

```
Este programa controla el equipo neocera ltc-11  
Nos vamos a conectar con pyvisa
```

```
Medimos la temperatura, la graficamos en tiempo real y lo guardamos  
en un txt.
```

```
Seteamos un setpoint de temperatura para que el equipo  
vaya a ese valor
```

```
"""
```

```
import visa  
import time  
import matplotlib.pyplot as plt  
import numpy as np
```

```
class handler():
```

```
    """
```

```
        Maneja los eventos de matplotlib y el control de  
temperatura
```

```
    """
```

```
    def __init__(self, line, puerto):  
        self.line = line # line es un objeto de matplotlib, es sobre  
lo que se grafica  
        self.puerto = puerto  
        self.setP = 0 #set point  
        self.setPex = 0 #punto decimal del set point  
        self.new = True # habilita la entrada de un nuevo dato  
        self.xs = line.get_xdata()  
        self.ys = line.get_ydata()  
        self.STOP = False # frena la iteracion  
        self.cid =  
line.figure.canvas.mpl_connect('close_event', self.stop) #Evento de  
cerrar la ventana  
        self.cidkey =  
line.figure.canvas.mpl_connect('key_press_event', self.setPoint) #Evento  
de apretar una tecla
```

```
    def stop(self, event):
```

```
        """
```

```
            Cuando se cierra la ventana se llama a este metodo  
para que cambie
```

```
            el valor de Stop y se frene la iteracion que esta  
corriendo siempre
```

```
        """
```

```
        self.STOP = True
```

```
    def cambiarSetP(self):
```

```
        """
```

```
            Nos conectamos a traves del puerto serie para cambiar  
el setpoint
```

```
        """
```

```
        print('nuevo set point: ', self.setP)  
        self.puerto.write('SMON;') #modo monitor (Esta linea no es  
estrictamente necesaria)
```

```
self.puerto.write('SETP1,'+str(self.setP)+';')# setea el
setPoint
```

```
self.puerto.write('SCONT;')# volvemos a modo control
```

```
def setPoint(self, event):
```

```
    """
```

```
        Maneja los eventos cuando se aprieta una tecla
```

```
        Para cambiar el setPoint escribir un numero con el
teclado(acepta punto decimal)
```

```
        No se ve en ninguna parte de la pantalla lo que se
escribe, pero igual esta escribiendo
```

```
        Apretar enter para mandar el setpoint al equipo
```

```
        La tecla borrar borra todo lo que se escribio para
empezar a escribir devuelta
```

```
    """
```

```
    try:
```

```
        n = int(event.key)
```

```
        if self.new:
```

```
            self.setP = n
```

```
            self.setPex = 0
```

```
            self.new = False
```

```
        elif self.setPex==0 :
```

```
            self.setP = self.setP*10 + n
```

```
        else:
```

```
            self.setP = self.setP + n*(10**self.setPex)
```

```
            self.setPex += 1
```

```
    except:
```

```
        if event.key == '.':
```

```
            self.setPex = -1
```

```
        if event.key == 'enter':
```

```
            self.new = True
```

```
            self.cambiarSetP()
```

```
        if event.key == 'backspace':
```

```
            self.new = True
```

```
        if event.key == 'r':
```

```
            plt.cla()
```

```
            print ('reset grafico')
```

```
print("hello world")
```

```
rm = visa.ResourceManager()
```

```
rm.list_resources()
```

```
tc = rm.open_resource('GPIB::10::INSTR')#a traves de tc nos conectamos
con el tc neocera Ltc 11
```

```
#revisar en cada caso el nombre correcto para el equipo
```

```
print(tc.query('*IDN?;'))
```

```
file = open('Temperatura.txt','a') #aca poner el nombre del archivo
donde se va a guardar
```

```
t0 = time.time()
t = t0

fig = plt.figure()
ax = fig.add_subplot(111)
line, = ax.plot([],[])

H = handler(line,tc)

while not H.STOP:
    T_aux = tc.query('QSAMP?1;')
    T = float(T_aux[0:len(T_aux)-3])
    t = time.time()

    file.write(T_aux+str(t-t0))

    ax.plot(t-t0,T,"r*")
    plt.pause(0.5) # muestra los datos y espera 0.5 segundos

file.close()
```