

gui_unifiée/class_TabMilleFeuilleIRM.py

```
import sys

#from PyQt5.QtCore import *
#from PyQt5.QtGui import *
#from PyQt5.QtWidgets import *
from PyQt5.QtCore import *
from PyQt5.QtWidgets import QApplication, QWidget, QGridLayout, QScrollBar,
QHBoxLayout, QVBoxLayout, QLabel, QGroupBox

from class_MilleFeuilleIRM import MilleFeuilleIRM

from class_Parametres import Parametres # Ne sert que si est exécuté
séparemment

"""
Classe TabMilleFeuilleIRM, hérite de la classe QGridLayout, c'est donc une
grille
Cette classe représente le contenu d'une fenêtre PyQt
Elle peut donc aussi être utilisée comme un onglet dans une fenêtre
@author Amaury
"""
class TabMilleFeuilleIRM(QGridLayout) :
    """
    Constructeur, crée le contenu de l'onglet
    """
    def __init__(self, objParams, parent=None) :
        super(TabMilleFeuilleIRM, self).__init__(parent) # Appel du
constructeur de QGridLayout

        self.objParams = objParams

        # Graphe à afficher
        self.milleFeuilleIRM = MilleFeuilleIRM()

        # Défilement de la couche X
        self.barreDeScrollIRMCoucheX = QScrollBar()
        self.barreDeScrollIRMCoucheX.setMaximum(
self.objParams.nombreImagesPlanYZ() )
        self.barreDeScrollIRMCoucheX.valueChanged.connect(
self.dessinerMilleFeuilleIRM )

        # Défilement de la couche Y
        self.barreDeScrollIRMCoucheY = QScrollBar()
        self.barreDeScrollIRMCoucheY.setMaximum(
self.objParams.nombreImagesPlanXZ() )
        self.barreDeScrollIRMCoucheY.valueChanged.connect(
self.dessinerMilleFeuilleIRM )

        # Défilement de la couche Z
        self.barreDeScrollIRMCoucheZ = QScrollBar()
        self.barreDeScrollIRMCoucheZ.setMaximum(
self.objParams.nombreImagesPlanXY() )
        self.barreDeScrollIRMCoucheZ.valueChanged.connect(
self.dessinerMilleFeuilleIRM )

        # Défilement temporel
        self.barreDeScrollIRMTemps = QScrollBar(Qt.Horizontal)
```

```

        self.barreDeScrollIRM Temps.setMaximum(
self.objParams.nombreInstantsTemporels() )
        self.barreDeScrollIRM Temps.valueChanged.connect(
self.dessinerMilleFeuilleIRM )

# Ajout des Widgets
#
self.addWidget( self.milleFeuilleIRM, 2, 1 )
self.addWidget( self.barreDeScrollIRMCoucheX, 1, 2 )
self.addWidget( self.barreDeScrollIRMCoucheY, 1, 3 )
self.addWidget( self.barreDeScrollIRMCoucheZ, 1, 4 )
self.addWidget( self.barreDeScrollIRM Temps, 2, 1 )

# Ajout Maylis
self.valeur_temps = QLabel("Temps : 0")
self.valeur_X = QLabel("X : 0")
self.valeur_Y = QLabel("Y : 0")
self.valeur_Z = QLabel("Z : 0")

group_box=QGroupBox("Positions courantes des barres de scroll")
horizontal_layout = QHBoxLayout()
group_box.setLayout(horizontal_layout)
vertical_layout = QVBoxLayout()

horizontal_layout.addWidget(self.valeur_temps)
horizontal_layout.addWidget(self.valeur_X)
horizontal_layout.addWidget(self.valeur_Y)
horizontal_layout.addWidget(self.valeur_Z)
vertical_layout.addWidget(group_box)
vertical_layout.addWidget(self.milleFeuilleIRM,stretch=2)

self.addLayout(vertical_layout,1,1)
# Fin Ajout Maylis

self.dessinerMilleFeuilleIRM(0)

"""
Gère le dessin et les changements de l'affichage IRM
"""
def dessinerMilleFeuilleIRM(self, value) :
    imageX = self.objParams.genererURLdesPGM3D( 'YZ',
self.barreDeScrollIRM Temps.value(), self.barreDeScrollIRMCoucheX.value() )
    imageY = self.objParams.genererURLdesPGM3D( 'XZ',
self.barreDeScrollIRM Temps.value(), self.barreDeScrollIRMCoucheX.value() )
    imageZ = self.objParams.genererURLdesPGM3D( 'XY',
self.barreDeScrollIRM Temps.value(), self.barreDeScrollIRMCoucheX.value() )

    self.milleFeuilleIRM.dessinerMilleFeuilleIRM( [imageX,
self.barreDeScrollIRMCoucheX.value() ],
                                                    [imageY,
self.barreDeScrollIRMCoucheY.value() ],
                                                    [imageZ,
self.barreDeScrollIRMCoucheZ.value() ] )

# Ajout Maylis
self.valeur_temps.setText("Temps : " +
str(self.barreDeScrollIRM Temps.value()))
self.valeur_X.setText("X : " +
str(self.barreDeScrollIRMCoucheX.value()))

```

```

        self.valeur_Y.setText("Y : " +
str(self.barreDeScrollIRMCoucheY.value()))
        self.valeur_Z.setText("Z : " +
str(self.barreDeScrollIRMCoucheZ.value()))
        # Fin ajouts Maylis

        print( "[Info TabMilleFeuilleIRM] X : " + str(
self.barreDeScrollIRMCoucheX.value() ) + ", Y : " + str(
self.barreDeScrollIRMCoucheY.value() ) + ", Z : " + str(
self.barreDeScrollIRMCoucheZ.value() ) + ", Temps : " + str(
self.barreDeScrollIRMTemps.value() ) )
        print( "[Info TabMilleFeuilleIRM] Affichage : " + imageX + ", " +
imageY + ", " + imageZ )

"""
Code principal pour démonstration
"""
# Si on est le script principal
# Cela permet de ne pas exécuter ce bloc de codes lorsque ce script est
importé par un autre
# Source : https://stackoverflow.com/questions/419163/what-does-if-name-main-do
if __name__ == '__main__' :
    application = QApplication(sys.argv) # Crée un objet de type
QApplication (Doit être fait avant la fenêtre)
    fenetre = QWidget() # Crée un objet de type QWidget
    fenetre.setWindowTitle("MODE DÉMONSTRATION") # Définit le nom de la
fenêtre
    fenetre.setLayout( TabMilleFeuilleIRM( Parametres() ) )
    fenetre.show() # Affiche la fenêtre
    application.exec_() # Attendre que tout ce qui est en cours soit
exécuté

```