## gui\_unifiée/class\_TabAide.py

```
import os
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, QTabWidget,
QHBoxLayout, QScrollArea, QLabel, QSizePolicy
from class Parametres import Parametres # Ne sert que si est exécuté
séparemment
PARAMETRES
FICHIER CONTENU AIDE = './contenu aide/'
from parametres import DISABLE IRM
Classe TabAide, 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 Maylis et Alexandre
class TabAide(QGridLayout) :
    Constructeur, crée le contenu de l'onglet
    def init__(self, objParams, parent=None, noVTK = False) :
        super(TabAide, self). init (parent) # Appel du constructeur de
QGridLayout
        self.objParams = objParams # Ne sert à rien pour le moment
        # Création d'onglets dans la page d'aide
        onglets = QTabWidget()
        onglets.setTabPosition(2)
        onglets.setTabShape(1) # C'est très moche quand même
        aide ongl = QWidget()
        aide onglet1 = QWidget()
        aide onglet2 = QWidget()
        if not DISABLE IRM : aide onglet3 = QWidget()
        aide onglet4 = OWidget()
        aide_onglet5 = QWidget()
        # Dictionnaire des onglets de la page d'aide
        self.ongl aide1 = { 'aide ongl' : [aide ongl,
"Introduction",
                                    'aide generale.html'] ,
                            'aide onglet4' : [aide_onglet4, "Aide vue
                          'aide coupes.html'] ,
Coupes",
                            'aide_onglet1' : [aide onglet1, "Aide
                              'aide visu graph.html'],
Trajectoires",
```

```
'aide onglet2' : [aide onglet2, "Aide Vue
                          'aide millefeuille.html'] }
Mille-feuilles",
        if not DISABLE IRM : self.ongl_aide2 = { 'aide_onglet3' :
[aide onglet3, "Aide Vue IRM",
                                                     'aide vision IRM.html']
       if not noVTK : self.ongl aide3 = { 'aide onglet5' : [aide onglet5,
                                    'aide VTK.html'] }
"Aide vue 3D",
        if not DISABLE IRM and not noVTK : self.ongl aide =
{**self.ongl aide1, **self.ongl aide2, **self.ongl aide3}
        elif not DISABLE_IRM and noVTK : self.ongl_aide =
{**self.ongl aide1, **self.ongl aide2}
        elif DISABLE IRM and not noVTK : self.ongl aide =
{**self.ongl aide1, **self.ongl aide3}
        else : self.ongl aide = {**self.ongl aide1}
        # Ajout dynamique des onglets à la page d'aide
        for ongl in self.ongl aide :
            onglets.addTab(self.ongl aide[ongl][0],
self.ongl aide[ongl][1])
            self.contenuOngletAide(ongl)
        # Ajout du conteneur d'onglets dans la grille et du layout dans
l'onglet Aide de la fenêtre
       self.addWidget(onglets)
    Affiche dans l'onglet indiqué le contenu du fichier .html associé
    def contenuOngletAide(self, nom onglet) :
        # Contenant avec barre de scroll
        zone de texte = QHBoxLayout()
        scroll area = QScrollArea()
        scroll area.setHorizontalScrollBarPolicy(Qt.ScrollBarAlwaysOff)
        scroll area.setWidgetResizable(True)
        # Ouverture du fichier contenant l'aide écrite au format HTML
        lien fichier = FICHIER CONTENU AIDE + self.ongl aide[nom onglet][2]
        if os.path.isfile( lien fichier ) : # Si le chemin d'accès existe
            fichier = open(lien fichier, 'r', encoding='utf-8')
            texte = QLabel(fichier.read())
            texte.setStyleSheet("QLabel { padding: 20px; margin-top: -40px;
}"); # CSS
        else :
            texte = QLabel("Le fichier suivant est manquant : " +
lien fichier)
            texte.setStyleSheet("QLabel { padding: 20px; }"); # CSS
            print( "[Erreur TabAide] " + lien fichier + " n'existe pas !" )
        # Ajutement de la forme du texte à la taille de la fenêtre
        texte.adjustSize()
        texte.setSizePolicy(QSizePolicy.Expanding, QSizePolicy.Expanding)
        texte.setWordWrap(True)
        texte.setAlignment(Qt.AlignJustify)
       scroll area.setWidget(texte)
        # Ajout dans le contenant et dans l'onglet correspondant
        zone de texte.addWidget(scroll area)
        self.ongl aide[nom onglet][0].setLayout(zone de texte)
```

```
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( TabAide( Parametres() ) )
    fenetre.show() # Affiche la fenêtre
    application.exec () # Attendre que tout ce qui est en cours soit
exécuté
```