import os

import sys

# 원본 파일 29번 줄 삭제 (data\_queue 사용하지 않음)

import platform

from PyQt5 import QtWidgets, QtGui, QtCore

import vlc

# 원본 파일 34번 줄 삭제 (클라이언트 사용하지 않음)

class MiniPlayer(QtWidgets.QMainWindow):

"""Stripped-down PyQt5-based media player class to sync with "master" video.

"""

def \_\_init\_\_(self, master=None): # 코드 수정 (data\_queue 사용 안함)

QtWidgets.QMainWindow.\_\_init\_\_(self, master)

# 원본 파일 43번 줄 이동 (self.\_\_init\_\_ 에서 main 으로 이동)

# 원본 파일 44, 45번 줄 삭제 (statusbar 사용할 필요 없음)

# 코드 추가 (추가 점수 2번 구현)

self.setWindowFlags(QtCore.Qt.WindowStaysOnTopHint) #ui를 맨 상단에 고정시킴

# Create a basic vlc instance

self.instance = vlc.Instance()

self.media = None

# Create an empty vlc media player

self.mediaplayer = self.instance.media\_player\_new()

self.init\_ui()

self.open\_file()

self.timer = QtCore.QTimer(self)

self.timer.setInterval(10)

# 코드 수정(update\_ui의 이름 check\_endState로) 변경

self.timer.timeout.connect(self.check\_endState)

# 원본 파일 62번 줄 삭제

self.timer.start()

def init\_ui(self):

"""Set up the user interface

"""

self.widget = QtWidgets.QWidget(self)

self.setCentralWidget(self.widget)

# In this widget, the video will be drawn

if platform.system() == "Darwin": # for MacOS

self.videoframe = QtWidgets.QMacCocoaViewContainer(0)

else:

self.videoframe = QtWidgets.QFrame()

self.palette = self.videoframe.palette()

self.palette.setColor(QtGui.QPalette.Window, QtGui.QColor(0, 0, 0))

self.videoframe.setPalette(self.palette)

self.videoframe.setAutoFillBackground(True)

self.vboxlayout = QtWidgets.QVBoxLayout()

self.vboxlayout.addWidget(self.videoframe)

self.widget.setLayout(self.vboxlayout)

def open\_file(self):

"""Open a media file in a MediaPlayer

"""

# 원본 파일 89번 줄 삭제

filename = 'targetfile.webm' # 코드 수정(파일 이름을 대상 영상파일로 고정)

# 원본 파일 91, 92번 줄 삭제

# getOpenFileName returns a tuple, so use only the actual file name

self.media = self.instance.media\_new(filename) # 코드 수정(변수 filename이 문자열로 변경됨)

# Put the media in the media player

self.mediaplayer.set\_media(self.media)

# Parse the metadata of the file

self.media.parse()

# Set the title of the track as the window title

self.setWindowTitle("{}".format(self.media.get\_meta(0)))

# The media player has to be 'connected' to the QFrame (otherwise the

# video would be displayed in it's own window). This is platform

# specific, so we must give the ID of the QFrame (or similar object) to

# vlc. Different platforms have different functions for this

if platform.system() == "Linux": # for Linux using the X Server

self.mediaplayer.set\_xwindow(int(self.videoframe.winId()))

elif platform.system() == "Windows": # for Windows

self.mediaplayer.set\_hwnd(int(self.videoframe.winId()))

elif platform.system() == "Darwin": # for MacOS

self.mediaplayer.set\_nsobject(int(self.videoframe.winId()))

# Start playing the video as soon as it loads

self.mediaplayer.play()

def check\_endState(self): # 코드 수정 (기능 변경에 따른 함수 이름 변경)

#원본 파일 121~146번 줄 삭제

# 코드 추가 (추가 점수 1 구현)

while self.mediaplayer.get\_state() == vlc.State.Ended: #mediaplayer 객체가 끝난 상태가 되면

self.mediaplayer.stop() #mediaplayer 멈추기

self.mediaplayer.set\_time(0) #재생 시각을 0으로 설정

self.mediaplayer.play() #mediaplayer 재생

return

# 원본 파일 148~151번 줄 삭제 (statusbar 사용할 필요 없음)

def main():

"""Entry point for our simple vlc player

"""

app = QtWidgets.QApplication(sys.argv)

# 원본 파일 159번 줄 삭제 (data\_queue 사용 안함)

player = MiniPlayer() # 코드 수정 (data\_queue 사용 안함)

player.setWindowTitle("한양대에리카 홍보영상")

player.show()

player.resize(480, 480)

# 원본 파일 165번 줄 삭제 (클라이언트 사용 안함)

sys.exit(app.exec\_())

if \_\_name\_\_ == "\_\_main\_\_":

main()