'''

gdrive (Google Drive ) for KODI / XBMC Plugin

Copyright (C) 2013-2016 ddurdle

This program is free software: you can redistribute it and/or modify

it under the terms of the GNU General Public License as published by

the Free Software Foundation, either version 3 of the License, or

(at your option) any later version.

This program is distributed in the hope that it will be useful,

but WITHOUT ANY WARRANTY; without even the implied warranty of

MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the

GNU General Public License for more details.

You should have received a copy of the GNU General Public License

along with this program. If not, see <http://www.gnu.org/licenses/>.

'''

import os

import re

import urllib, urllib2

import cookielib

import xbmc, xbmcaddon, xbmcgui, xbmcplugin

import authorization

import crashreport

from resources.lib import package

from resources.lib import file

from resources.lib import folder

class TMDB:

API\_KEY = '8877595a1e6647d272148c99133df1fb'

def \_\_init\_\_(self, service, addon, user\_agent):

self.addon = addon

self.service = service

# self.crashreport = crashreport

# self.crashreport.sendError('test','test')

self.user\_agent = user\_agent

return

#

# Search for a movie by title+year

# Return the movieID #

#

def movieSearch(self, title, year):

title = re.sub(' ', '%20', title)

url = 'https://api.themoviedb.org/3/search/movie?api\_key='+self.API\_KEY+'&language=en-US&query='+title+'&year='+year

mediaList = []

req = urllib2.Request(url, None, self.service.getHeadersList())

try:

response = urllib2.urlopen(req)

except urllib2.URLError, e:

if e.msg != '':

xbmcgui.Dialog().ok(self.addon.getLocalizedString(30000), e.msg)

xbmc.log(self.addon.getAddonInfo('movieSearch') + ': ' + str(e), xbmc.LOGERROR)

self.crashreport.sendError('movieSearch',str(e))

response\_data = response.read()

response.close()

for r in re.finditer('"id":(\d+),' ,

response\_data, re.DOTALL):

movieid = r.group(1)

return movieid

return None

#

# Search for a movie by title+year

# Return the movieID #

#

def movieDetails(self, movieID):

url = 'https://api.themoviedb.org/3/movie/'+movieID+'?api\_key='+self.API\_KEY + '&language=en-US'

mediaList = []

req = urllib2.Request(url, None, self.service.getHeadersList())

try:

response = urllib2.urlopen(req)

except urllib2.URLError, e:

if e.msg != '':

xbmcgui.Dialog().ok(self.addon.getLocalizedString(30000), e.msg)

xbmc.log(self.addon.getAddonInfo('movieDetails') + ': ' + str(e), xbmc.LOGERROR)

self.crashreport.sendError('movieDetails',str(e))

response\_data = response.read()

response.close()

for r in re.finditer('"id":(\d+),' ,

response\_data, re.DOTALL):

movieid = r.group(1)

return movieid

return None