import sys

#!{sys.executable} -m pip install BeautifulSoup4

#!{sys.executable} -m pip install lxml

import requests

import csv

import time

import sys

from bs4 import BeautifulSoup

#from lxml import etree

from xml.etree import ElementTree

def stringEliminarapartirdeuncaracter(cadena, caracter):

num\_apariciones = cadena.count(caracter)

if num\_apariciones>0:

pos\_caracter =cadena.find(caracter)

return cadena[0:pos\_caracter]

#funcion que recorre un sitemap que recibe cómo parámetro y devuelve la lista de urls

def listadoPaginasSiteMap(sitemap\_url,tag):

page\_site\_map = requests.get(sitemap\_url)

sitemap = BeautifulSoup(page\_site\_map.content, "lxml")

tags = sitemap.find\_all(tag)

lista\_paginas=[]

for pagina\_site in tags:

print (pagina\_site.get\_text().strip())

lista\_paginas.extend([pagina\_site.get\_text().strip()])

return lista\_paginas

#funcion que crea un csv a partir de una lista

def crearficherocsv(nombrefichero,lista):

hoy= time.strftime("%y%m%d")

fichero = str(hoy)+"\_"+nombrefichero+'.csv'

with open ( fichero , 'w' , newline = '') as csvfile:

#with open ( fichero , 'w' , newline = '', encoding="utf-8" ) as csvfile:

spamwriter = csv.writer(csvfile, delimiter = ';' ,

quotechar = '|' , quoting = csv.QUOTE\_MINIMAL)

for listaitem in lista:

try:

print(listaitem)

spamwriter.writerow(listaitem)

except ValueError:

print("Algo ha ido mal al escribir el fichero")

#recupera todos los productos de una determinada pagina

def recuperaProductosPagina(url):

#carga una página que recibe como parametro, a priori recibe páginas que contienen listados de productos

#page = requests.get("https://www.mediamarkt.es/es/category/\_c%C3%A1maras-r%C3%A9flex-701197.html")

page = requests.get(url)

page\_soup = BeautifulSoup(page.content,"lxml")

#listado de productos

#todos los productos tienen las siguientes secciones

#<div class="product-wrapper

# <aside class="product-photo">

# <aside class="product-price alt">

# <div class="content ">

#recuperamos la categoria de los productos

listaproductos\_pagina=[] # lista en la cual guardamos la información de todos los productos

h1\_categoria =page\_soup .find("h1")

if (h1\_categoria is None):

return listaproductos\_pagina

categoria\_text= h1\_categoria.get\_text().strip()

categoria\_text= stringEliminarapartirdeuncaracter(categoria\_text,"(") #funcio que se queda con el substring

categoria\_text=str(categoria\_text).strip()

#recuperamos todos los productos de la lista

for product in page\_soup.find\_all("div", {"class": "product-wrapper"}):

item\_producto=[1,2,3,4,5,6]

item\_producto[0]= categoria\_text

#producto\_id

producto\_id\_text = product.get("data-modelnumber")

item\_producto[1]=producto\_id\_text

#producto\_nombre

h2\_producttitle=product.find("h2")

producto\_nombre = h2\_producttitle.get\_text().strip()

producto\_nombre= producto\_nombre.replace(";",",")

item\_producto[2]=producto\_nombre

#producto manufacturer

div\_content = product.find("div", {"class": "content"})

imgss= div\_content.findChildren("img" , recursive=True)

item\_producto[3]=""

for img in imgss:

marca = img.get("alt")

item\_producto[3]=marca

#print(marca)

break;

#producto\_precio

div\_price = product.find("div", {"class": "price"})

if div\_price is None:

item\_producto[4]=""

else:

precio =div\_price.get\_text().strip();

producto\_precio= precio.replace(",-","")

item\_producto[4]=producto\_precio

#producto\_precio\_anterior

producto\_precio\_ant="SINDESCUENTO"

div\_price\_old = product.find("div", {"class": "price-old"})

if div\_price\_old is None:

producto\_precio\_ant="SINDESCUENTO"

else:

precio\_tmp =div\_price\_old.get\_text().strip()

producto\_precio\_ant= precio\_tmp.replace(",-","")

item\_producto[5]=producto\_precio\_ant

listaproductos\_pagina.append(item\_producto)

return listaproductos\_pagina

#recuperamos todas las páginas que contienen listas de productos

lista\_paginas = listadoPaginasSiteMap("https://www.mediamarkt.es/sitemap/sitemap-productlist.xml", "loc")

#lista\_paginas= [1]

#lista\_paginas[0]="https://www.mediamarkt.es/es/category/\_inal%C3%A1mbricos-702527.html"

listaproductos=[] # lista en la cual guardamos la información de todos los productos de todas las paginas

aux =1

#iteramos por todas las listas de productos

for pagina\_url in lista\_paginas:

print( "pagina\_url: " +str(aux)+ " " + pagina\_url)

productos\_recuperados =recuperaProductosPagina(pagina\_url)

listaproductos.extend(productos\_recuperados)

#cada lista de productos puede tener a su ver paginacion

paginacion =2

while productos\_recuperados != []:

nueva\_url = pagina\_url + "?searchParams=&sort=&view=&page=" + str(paginacion)

paginacion = paginacion + 1

productos\_recuperados =recuperaProductosPagina(nueva\_url)

listaproductos.extend(productos\_recuperados)

print(paginacion)

aux = aux +1

# if aux >4:

# break

crearficherocsv("listaproductos",listaproductos)