Національний технічний університет України
«Київський політехнічний інститут»
Фізико-Технічний Інститут
Кафедра математичних методів захисту інформації

Аналітика Big Data

Практичне заняття №1

Формування масиву даних для подальшої аналітичної обробки 5 ст. RSS

Виконав:
студент групи ФІ-73
Чіхладзе Вахтанг

Перевірив:
Ланде Д.В.

Київ 2021

Список RSS-фідів:

https://habr.com/ru/rss/all/all/?fl=ru
https://3dnews.ru/news/rss/
https://mobile-review.com/news/feed/
https://techtoday.in.ua/feed
https://pcnews.ru/feeds/latest/news/
https://pcnews.ru/feeds/latest/articles/
https://ko.com.ua/rss.xml/
https://ko.com.ua/rss/article
https://www.helpnetsecurity.com/feed/
https://www.itweek.ru/rss/

Завдання на самостійну роботу

1)Розширити список rss-каналів каналами комп'ютерної і телекомунікаційної спрямованості (але не торговельними майданчиками).
2)Створити процедуру (скрипт) для періодичного скачування інформації із створенного переліку RSS-фідів.
3)Реалізувати процедуру створення файлу, в якому об'єднуються всі скачані RSS-фіди, і підключити її до скрипту скачування.

1)Мої RSS-канали:

http://feeds.feedburner.com/Itcua?format=xml
https://ichip.ru/out/rss/main/main.xml

2),3)Було реалізовано скрипт мовою python 3.8.3, який використовує функціонал операційної системи Linux для інціалізації файлу та запису rss каналів утілітою curl. Також було створено функцію, яка обробляє rss-канали для виділення та перенесення вмісту унікальних item у новий файл. Періодичність забезпечується функцією, яка визиває почергово функції в інтервалах часу за допомогою функції sleep() від різниці часу.

```
#!python 3.8.3
#!OS Linux ippolit-PC 5.0.0-32-generic #34~18.04.2-Ubuntu SMP Thu Oct 10 10:36:02 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux

import os
import re
import time
from time import sleep
from datetime import datetime, timedelta
import xml.etree.ElementTree as ET
from xml.etree.ElementTree import ParseError


rss_links_given=[
"https://habr.com/ru/rss/all/all/?fl=ru",
"https://3dnews.ru/news/rss/",
"https://www.mobile-review.com/news/feed/",
```

```
"https://techtoday.in.ua/feed",
"https://pcnews.ru/feeds/latest/news/",
"https://pcnews.ru/feeds/latest/articles/",
"https://ko.com.ua/rss.xml/",
"https://ko.com.ua/rss/article",
"https://www.helpnetsecurity.com/feed/",
"https://www.itweek.ru/rss/"
]

rss_links_my=["http://feeds.feedburner.com/Itcua?format=xml",
"https://ichip.ru/out/rss/main/main.xml"]

rss_links=rss_links_given + rss_links_my

def get_rss_feeds_using_os():
iteration=0;
for link in rss_links:
    file_name=str(iteration)+"_"+re.match(r"http[s]*://([\
w.]*)",link).group(1)+"_"+str(time.asctime()).replace(" ","_")+".xml";
    os.system("curl -o "+"rss_feeds/"+file_name+" "+link);
    iteration+=1;


def create_result_file():
os.system('touch rss_feeds.xml')
items=ET.Element('items')
item=ET.SubElement(items,'item')
item.text="first item"
data = ET.tostring(items, encoding='utf-8', method='xml').decode('utf-8')
resultfile = open("rss_feeds.xml", "w")
resultfile.write(data)


def add_items():
    filetree=ET.parse("./rss_feeds.xml")
    items=filetree.getroot()
    for (_, _, filenames) in os.walk("rss_feeds"):
        for filename in filenames:
            print(filename)
            try:
            tree=ET.parse("rss_feeds/"+filename);
            except ParseError:
            continue;
            root = tree.getroot()
                for item in root.findall('./channel/item'):
                    newitem=ET.SubElement(items,'item')
                    for component in item:
                        newcomponent=ET.SubElement(newitem,component.tag)
                        newcomponent.text=component.text
                        final=ET.tostring(items, encoding='utf-8',
method='xml').decode('utf-8')
```

```
        file=open("rss_feeds.xml","w")
        file.write(final)
        file.close()
        os.system("rm -r ./rss_feeds/*")


def remove_occuring_items():
    filetree=ET.parse("./rss_feeds.xml")
    items=filetree.getroot()
    for i in range(len(items)):
        for j in range(i+1,len(items)):
            try:
                if items[i].find('.//link').text==items[j].find('.//link').text:
                items.remove(items[j]);
            except AttributeError:
                continue;
            except IndexError:
                continue;
    final=ET.tostring(items, encoding='utf-8', method='xml').decode('utf-8')
    file=open("rss_feeds.xml","w")
    file.write(final)
    file.close()

def init():
    get_rss_feeds_using_os();
    create_result_file();
    add_items();
    remove_occuring_items();

def periodic(hour):
    while True:
        get_rss_feeds_using_os();
        add_items();
        remove_occuring_items();
        now=datetime.now()
        start = now+timedelta(hours=hour)
        sleep((start-now).total_seconds())


if __name__ == "__main__":
    print('0.init\n1.collect data')
    choise=int(input())
    if choise == 0:
        init();
    if choise == 1:
        period=0
        period_enter=False
        while period<=0:
        if period_enter:
            print('Reenter period(in hours):')
            period=int(input());
```

```
else:
    print('Enter period(in hours):')
    period=int(input());
    period_enter=period_enter|True
    periodic(period)
else:
    exit()
```

Висновок: отже, я навчився скачувати rss-канали та добавляти їх у файл. Також я отримав навички з обробки xml файлів.