Санкт-Петербургский национальный исследовательский университет ИТМО

Факультет Программной Инженерии и Компьютерной техники

Информатика

Лабораторная работа №4

Выполнил:

Мухамеджанов Артур Илдусович

P3114

Преподаватели:

Балакшин Б. В.

Машина Е. А.

**Санкт-Петербург**

**2022**

**Задание**

**Вариант 6**

1)Обязательное задание (позволяет набрать до 65 процентов от максимального числа баллов БаРС за данную лабораторную): написать программу на языке Python 3.x, которая бы осуществляла парсинг и конвертацию исходного файла в новый

JSON -> XML, вторник.

2) Дополнительное задание №1 (позволяет набрать +10 процентов от максимального числа баллов БаРС за данную лабораторную). a) Найти готовые библиотеки, осуществляющие аналогичный парсинг и конвертацию файлов. b) Переписать исходный код, применив найденные библиотеки. Регулярные выражения также нельзя использовать. c) Сравнить полученные результаты и объяснить их сходство/различие.

3) Дополнительное задание №2 (позволяет набрать +10 процентов от максимального числа баллов БаРС за данную лабораторную). a) Переписать исходный код, добавив в него использование регулярных выражений. 2 b) Сравнить полученные результаты и объяснить их сходство/различие.

4) Дополнительное задание №3 (позволяет набрать +10 процентов от максимального числа баллов БаРС за данную лабораторную). a) Используя свою исходную программу из обязательного задания, программу из дополнительного задания №1 и программу из дополнительного задания №2, сравнить стократное время выполнения парсинга + конвертации в цикле. b) Проанализировать полученные результаты и объяснить их сходство/различие.

5) Дополнительное задание №4 (позволяет набрать +5 процентов от максимального числа баллов БаРС за данную лабораторную). c) Переписать исходную программу, чтобы она осуществляла парсинг и конвертацию исходного файла в любой другой формат (кроме JSON, YAML, XML, HTML): PROTOBUF, TSV, CSV, WML и т.п. d) Проанализировать полученные результаты, объяснить особенности использования формата.

1.

Файл JSON:

{  
 "lessons": {  
 "day": "Вт",  
 "lesson": [  
 {  
 "time": "8:20-9:50",  
 "room": "",  
 "lesson-name": "ИНФОРМАТИКА(ЛЕК) ZOOM Балакшин Павел Валерьевич",  
 "lesson-format": "Дистанционно"  
 },  
 {  
 "time": "8:20-9:50",  
 "room": "2308 (БЫВШ. 306) АУД. Кронверкский пр., д.49, лит.А",  
 "lesson-name": "ИНФОРМАТИКА(ЛАБ) четная неделя Машина Екатерина Алексеевна",  
 "lesson-format": "Очно - дистанционный"  
 },  
 {  
 "time": "8:20-9:50",  
 "room": "ул.Ломоносова, д.9, лит. М",  
 "lesson\_name": "ИНФОРМАТИКА(ЛЕК) АКТОВЫЙ ЗАЛ Балакшин Павел Валерьевич",  
 "lesson-format": "Очно - дистанционный"  
 },  
 {  
 "time": "10:00-11:30",  
 "room": "ул.Ломоносова, д.9, лит. М",  
 "lesson-name": "ОСНОВЫ ПРОФЕССИОНАЛЬНОЙ ДЕЯТЕЛЬНОСТИ(ЛЕК) АКТОВЫЙ ЗАЛ Клименков Сергей Викторович",  
 "lesson-format": "Очно - дистанционный"  
 },  
 {  
 "time": "10:00-11:30",  
 "room": "",  
 "lesson-name": "ОСНОВЫ ПРОФЕССИОНАЛЬНОЙ ДЕЯТЕЛЬНОСТИ(ЛЕК) ZOOM Клименков Сергей Викторович",  
 "lesson-format": "Дистанционный"  
 },  
 {  
 "time": "10:00-11:30",  
 "room": "2308 (БЫВШ. 306) АУД. Кронверкский пр., д.49, лит.А",  
 "lesson-name": "ИНФОРМАТИКА(ЛАБ) четная неделя Машина Екатерина Алексеевна",  
 "lesson-format": "Очно - дистанционный"  
 },  
 {  
 "time": "11:40-13:10",  
 "room": "2308 (БЫВШ. 306) АУД. Кронверкский пр., д.49, лит.А",  
 "lesson-name": "ПРОГРАММИРОВАНИЕ(ЛАБ) четная неделя Чагина Инна Николаевна",  
 "lesson-format": "Очно - дистанционный"  
 },  
 {  
 "time": "11:40-13:10",  
 "room": "",  
 "lesson-name": "ПРОГРАММИРОВАНИЕ(ЛЕК) ZOOM Письмак Алексей Евгеньевич",  
 "lesson-format": "Дистанционно"  
 },  
 {  
 "time": "8:20-9:50",  
 "room": "ул.Ломоносова, д.9, лит. М",  
 "lesson-name": "ПРОГРАММИРОВАНИЕ(ЛЕК) ZOOM Письмак Алексей Евгеньевич",  
 "lesson-format": "Очно-дистанционный"  
 },  
 {  
 "time": "13:30-15:00",  
 "room": "2308 (БЫВШ. 306) АУД. Кронверкский пр., д.49, лит.А",  
 "lesson-name": "ПРОГРАММИРОВАНИЕ(ЛАБ) четная неделя Чагина Инна Николаевна",  
 "lesson-format": "Очно - дистанционный"  
 }  
 ]  
 }  
}

Файл программы

import time  
  
  
def json\_to\_xml(read\_file, write\_file):  
 amount\_of\_tabs = 0  
 write\_file.write("<?xml version=\"1.0\" encoding=\"UTF-8\" ?>\n")  
 tg\_array = []  
 multiple\_tg = ''  
 for line in read\_file:  
 s = line.rstrip()  
 s = s.lstrip()  
 s = s.rstrip()  
 if s[len(s) - 1] == ',':  
 s = s[:len(s) - 1]  
 #Проверка на исправность файла  
 if len(s) == 1 and (s != "{" and s != "]" and s != "}" and s != ']'):  
 print("JSON файл ошибочен")  
 exit(0)  
 if len(s) == '':  
 continue  
 if len(s) > 1 and s.count(":") == 0:  
 print("JSON файл ошибочен")  
 exit(0)  
  
 if s == "{":  
 if multiple\_tg != '':  
 write\_file.write(' ' \* amount\_of\_tabs + '<' + multiple\_tg + '>\n')  
 amount\_of\_tabs += 1  
 continue  
  
 if s.count('}'):  
 if multiple\_tg != '':  
 amount\_of\_tabs -= 1  
 write\_file.write(' ' \* amount\_of\_tabs + '</' + multiple\_tg + '>\n')  
 continue  
 if tg\_array != []:  
 amount\_of\_tabs -= 1  
 write\_file.write(' ' \* amount\_of\_tabs + '</' + tg\_array[len(tg\_array) - 1] + '>\n')  
 tg\_array = tg\_array[:len(tg\_array) - 1]  
 continue  
  
 if s == "]":  
 multiple\_tg = ''  
  
 if s[len(s) - 1] == ',':  
 s = s[:len(s) - 1]  
 if len(s) < 2:  
 continue  
 s = s.split(':')  
 if s[0] == "\"time\"":  
 s[1] = s[1] + ":" + s[2] + ":" + s[3]  
 s = s[0:2]  
 s[0] = s[0].lstrip()  
 s[0] = s[0].rstrip()  
 s[1] = s[1].lstrip()  
 s[1] = s[1].rstrip()  
  
 if s[0][0] != '\"' or s[0][len(s[0]) - 1] != '\"':  
 print("JSON файл ошибочен")  
 exit(0)  
  
 if s[1] == '{':  
 write\_file.write(' ' \* amount\_of\_tabs + '<' + s[0] + '>\n')  
 tg\_array.append(s[0].replace('\"', ''))  
 amount\_of\_tabs += 1  
 continue  
 if s[1] == '[':  
 multiple\_tg = s[0].replace('\"', '')  
 continue  
 if s[1][0] != '\"' or s[1][len(s[1]) - 1] != '\"':  
 print("JSON файл ошибочен")  
 exit(0)  
 write\_file.write(' ' \* amount\_of\_tabs + '<' + s[0] + '>' + s[1] + '</' + s[0] + '>\n')  
  
  
start = time.time()  
for i in range(100):  
 read\_file = open('Vtornik\_json', "r", encoding="utf8")  
 write\_file = open('Vtornik\_xml', "w", encoding="utf8")  
 json\_to\_xml(read\_file, write\_file)  
 read\_file.close()  
 write\_file.close()  
print(time.time() - start)

XML файл

<?xml version="1.0" encoding="UTF-8" ?>  
<lessons>  
 <day>Вт</day>  
 <lesson>  
 <time>8:20-9:50</time>  
 <room></room>  
 <lesson-name>ИНФОРМАТИКА(ЛЕК)ZOOMБалакшинПавелВалерьевич</lesson-name>  
 <lesson-format>Дистанционно</lesson-format>  
 </lesson>  
 <lesson>  
 <time>8:20-9:50</time>  
 <room>2308(БЫВШ.306)АУД.Кронверкскийпр.,д.49,лит.А</room>  
 <lesson-name>ИНФОРМАТИКА(ЛАБ)четнаянеделяМашинаЕкатеринаАлексеевна</lesson-name>  
 <lesson-format>Очно-дистанционный</lesson-format>  
 </lesson>  
 <lesson>  
 <time>8:20-9:50</time>  
 <room>ул.Ломоносова,д.9,лит.М</room>  
 <lesson\_name>ИНФОРМАТИКА(ЛЕК)АКТОВЫЙЗАЛБалакшинПавелВалерьевич</lesson\_name>  
 <lesson-format>Очно-дистанционный</lesson-format>  
 </lesson>  
 <lesson>  
 <time>10:00-11:30</time>  
 <room>ул.Ломоносова,д.9,лит.М</room>  
 <lesson-name>ОСНОВЫПРОФЕССИОНАЛЬНОЙДЕЯТЕЛЬНОСТИ(ЛЕК)АКТОВЫЙЗАЛКлименковСергейВикторович</lesson-name>  
 <lesson-format>Очно-дистанционный</lesson-format>  
 </lesson>  
 <lesson>  
 <time>10:00-11:30</time>  
 <room></room>  
 <lesson-name>ОСНОВЫПРОФЕССИОНАЛЬНОЙДЕЯТЕЛЬНОСТИ(ЛЕК)ZOOMКлименковСергейВикторович</lesson-name>  
 <lesson-format>Дистанционный</lesson-format>  
 </lesson>  
 <lesson>  
 <time>10:00-11:30</time>  
 <room>2308(БЫВШ.306)АУД.Кронверкскийпр.,д.49,лит.А</room>  
 <lesson-name>ИНФОРМАТИКА(ЛАБ)четнаянеделяМашинаЕкатеринаАлексеевна</lesson-name>  
 <lesson-format>Очно-дистанционный</lesson-format>  
 </lesson>  
 <lesson>  
 <time>11:40-13:10</time>  
 <room>2308(БЫВШ.306)АУД.Кронверкскийпр.,д.49,лит.А</room>  
 <lesson-name>ПРОГРАММИРОВАНИЕ(ЛАБ)четнаянеделяЧагинаИннаНиколаевна</lesson-name>  
 <lesson-format>Очно-дистанционный</lesson-format>  
 </lesson>  
 <lesson>  
 <time>11:40-13:10</time>  
 <room></room>  
 <lesson-name>ПРОГРАММИРОВАНИЕ(ЛЕК)ZOOMПисьмакАлексейЕвгеньевич</lesson-name>  
 <lesson-format>Дистанционно</lesson-format>  
 </lesson>  
 <lesson>  
 <time>8:20-9:50</time>  
 <room>ул.Ломоносова,д.9,лит.М</room>  
 <lesson-name>ПРОГРАММИРОВАНИЕ(ЛЕК)ZOOMПисьмакАлексейЕвгеньевич</lesson-name>  
 <lesson-format>Очно-дистанционный</lesson-format>  
 </lesson>  
 <lesson>  
 <time>13:30-15:00</time>  
 <room>2308(БЫВШ.306)АУД.Кронверкскийпр.,д.49,лит.А</room>  
 <lesson-name>ПРОГРАММИРОВАНИЕ(ЛАБ)четнаянеделяЧагинаИннаНиколаевна</lesson-name>  
 <lesson-format>Очно-дистанционный</lesson-format>  
 </lesson>  
</lessons>

2.

Воспользуемся библиотекой json и библиотекой dict2xml

import json  
from dict2xml import dict2xml  
  
read\_file = open('Vtornik\_json', "r", encoding="utf8")  
write\_file = open('Vtornik\_xml', "w", encoding="utf8")  
text = json.load(read\_file)  
xml = dict2xml(text)  
write\_file.write(xml)  
read\_file.close()  
write\_file.close()

3.

import re  
  
def json\_to\_xml(read\_file, write\_file):  
 amount\_of\_tabs = 0  
 write\_file.write("<?xml version=\"1.0\" encoding=\"UTF-8\" ?>\n")  
 tg\_array = []  
 multiple\_tg = ''  
 for line in read\_file:  
 s = line.rstrip()  
 s = re.sub(" ", "", s)  
 s = re.sub('**\"**', '', s)  
  
 if re.match('^{', s):  
 if multiple\_tg != '':  
 write\_file.write(' ' \* amount\_of\_tabs + '<' + multiple\_tg + '>\n')  
 amount\_of\_tabs += 1  
 continue  
  
 if re.match('^}', s):  
 if multiple\_tg != '':  
 amount\_of\_tabs -= 1  
 write\_file.write(' ' \* amount\_of\_tabs + '</' + multiple\_tg + '>\n')  
 continue  
 if tg\_array != []:  
 amount\_of\_tabs -= 1  
 write\_file.write(' ' \* amount\_of\_tabs + '</' + tg\_array[len(tg\_array) - 1] + '>\n')  
 tg\_array = tg\_array[:len(tg\_array) - 1]  
 continue  
  
 if re.match('^]', s):  
 multiple\_tg = ''  
  
 if re.match('.$', s):  
 s = s[:len(s) - 1]  
 s = s.split(':')  
 if len(s) > 2:  
 s[1] = s[1] + ":" + s[2] + ":" + s[3]  
 s = s[0:2]  
 if len(s) < 2:  
 continue  
 if re.match('{', s[1]):  
 write\_file.write(' ' \* amount\_of\_tabs + '<' + s[0] + '>\n')  
 tg\_array.append(s[0])  
 amount\_of\_tabs += 1  
 continue  
 if re.match('\[', s[1]):  
 multiple\_tg = s[0]  
 continue  
 write\_file.write(' ' \* amount\_of\_tabs + '<' + s[0] + '>' + s[1] + '</' + s[0] + '>\n')  
  
  
read\_file = open('Vtornik\_json', "r", encoding="utf8")  
write\_file = open('Vtornik\_xml', "w", encoding="utf8")  
json\_to\_xml(read\_file, write\_file)  
read\_file.close()  
write\_file.close()

4.

Для фиксирования времени используем библиотеку time

Зафиксируем время начала программы

start = time.time()

В конце выполнения программы выведем разницу между нынешним временем системы и временем начала программы

print(time.time() - start)

Тюк при каждом запуске программы время выполнения меняется, я возьму среднее время выполнения для каждой программы.

Программа 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Запуск | 1 | 2 | 3 | 4 | 5 |
| Время | 0.13346147 | 0.149774074 | 0.116769552 | 0.117773532 | 0.117773532 |

Среднее время выполнения t1 = 0.127110432

Программа 2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Запуск | 1 | 2 | 3 | 4 | 5 |
| Время | 0.18357801 | 0.14992785 | 0.183043 | 0.131527 | 0.16524434 |

Среднее время выполнения t2 = 0.16266404

Программа 3

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Запуск | 1 | 2 | 3 | 4 | 5 |
| Время | 0.18282508 | 0.14994335 | 0.1662857 | 0.15012788 | 0.1708827 |

Среднее время выполнения t2 = 0.164012942

Могу предположить, что программа 1 работает быстрее из-за меньшего количества импортируемых библиотек.

5.

JSON -> CSV

import pandas as pd  
  
read\_file = open('Vtornik\_json', encoding='utf8')  
df = pd.read\_json(read\_file)  
  
df.to\_csv('Vtornik.csv', encoding='cp1251', index = False)