Университет ИТМО

Факультет ПИиКТ

Студент Птицын Максим Евгеньевич

Группа P3130

Преподаватель Рудникова Тамара Владимировна

Лабораторная работа №4.

«Исследование протоколов, форматов обмена информацией и языков разметки документов».

Вариант № 20.

г. Санкт-Петербург

2021 г.

# Задание 1.

## Parse To Text:

import codecs

fileObj = codecs.open(r"C:\edu\inf\lab4\Thursday schedule.xml", "r", "utf\_8\_sig" )

fileResult = open(r"C:\edu\inf\lab4\result.txt", "w")

parse\_raw=[]

j=-1

for line in fileObj:

for i in range(len(line)-6):

if(line[i]+line[i+1]+line[i+2]+line[i+3]+line[i+4]+line[i+5]=='<para>'):

parse\_raw.append('')

j+=1

l=i+6

while (line[l]+line[l+1]+line[l+2]+line[l+3]+line[l+4]+line[l+5]+line[l+6]!='</para>'):

parse\_raw[j]+=line[l]

l+=1

parse=[]

for i in range(len(parse\_raw)):

for j in range(len(parse\_raw[i])):

if(parse\_raw[i][j]=='>')and(j!=len(parse\_raw[i])-1):

s=''

k=j+1

while(parse\_raw[i][k]!='<'):

s+=parse\_raw[i][k]

k+=1

parse\_raw[i]=s

break

for i in parse\_raw:

if (i not in parse) and(i[0]!='<'):

if(i[0]!='$'):

parse.append(i)

else:

parse.append(i[-2:])

for i in range(len(parse)):

fileResult.write(parse[i]+'\n')

fileObj.close()

fileResult.close()

## Parse To Json:

import codecs

fileObj = codecs.open(r"C:\edu\inf\lab4\Thursday schedule.xml", "r", "utf\_8\_sig" )

fileResult = open(r"C:\edu\inf\lab4\result.json", "w")

json=''

parse\_raw=[]

parse=[]

tab\_count=-1

contsruction\_count=0

j=-1

for line in fileObj:

if not 'version' in line:

for i in range(len(line)):

if line[i]=='<':

if line[i+1]=='/':

contsruction\_count+=1

parse\_raw.append('}')

else:

k=i+1

s='"'

while line[k]!='>' and k!=len(line):

s+=line[k]

k+=1

s+='":'

if line[k-1]!='/':

s+='{'

parse\_raw.append(s)

k+=1

s='"'

while k!=len(line) and line[k]!='<':

s+=line[k]

k+=1

s+='"'

parse\_raw.append(s)

for l in parse\_raw:

if (l!='""') and (l!='"\n"'):

parse.append(l)

for line in parse:

if line[-1:]=='{':

tab\_count+=1

json+=" "\*tab\_count

json+=line+'\n'

if line=='}':

tab\_count-=1

fileResult.write(json)

fileObj.close()

fileResult.close()

# Задание 2.

import xmltodict,json

import codecs

fileObj = codecs.open(r"C:\edu\inf\lab4\Thursday schedule.xml", "r", "utf\_8\_sig" )

fileResult = open(r"C:\edu\inf\lab4\result with lib.json", "w")

file=fileObj.read()

obj= xmltodict.parse(file)

fileObj.close()

fileResult.write(json.dumps(obj))

fileResult.close()

# Задание 3.

import re,codecs  
fileObj = codecs.open(r"C:\edu\inf\lab4\Thursday schedule.xml", "r", "utf\_8\_sig" )  
fileResult= open(r"C:\edu\inf\lab4\result with reg.txt", 'w')  
file=fileObj.read()  
who=re.search(r'[А-Я]\w+\s[А-Я]\w+\s[А-Я]\w+',file).group()  
d=re.findall(r'<title>\w+</title>',file)  
day=re.search(r'[а-я]\w+',d[0], flags=re.IGNORECASE).group()  
when=re.search(r'\d\d:\d\d-\d\d:\d\d', file).group()  
where=re.search(r'\d\d\d\s\w+',file).group()  
address=re.search(r'[А-Я]\w+\s[а-я]\D+\d+\D+\s',file).group()  
science=re.search(r'[А-Я]\w+\s[а-я]\w{5,}',file).group()  
result=day+'\n'+when+'\n'+address+'\n'+where+'\n'+who+'\n'+science  
fileResult.write(result)  
fileObj.close()  
fileResult.close()

# Задание 4.

import re,codecs,time,xmltodict,json

#ParseToJson

start\_time = time.time()

for i in range(10):

fileObj = codecs.open(r"C:\edu\inf\lab4\Thursday schedule.xml", "r", "utf\_8\_sig" )

fileResult = open(r"C:\edu\inf\lab4\result.json", "w")

json\_final=''

parse\_raw=[]

parse=[]

tab\_count=-1

contsruction\_count=0

j=-1

for line in fileObj:

if not 'version' in line:

for i in range(len(line)):

if line[i]=='<':

if line[i+1]=='/':

contsruction\_count+=1

parse\_raw.append('}')

else:

k=i+1

s='"'

while line[k]!='>' and k!=len(line):

s+=line[k]

k+=1

s+='":'

if line[k-1]!='/':

s+='{'

parse\_raw.append(s)

k+=1

s='"'

while k!=len(line) and line[k]!='<':

s+=line[k]

k+=1

s+='"'

parse\_raw.append(s)

for l in parse\_raw:

if (l!='""') and (l!='"\n"'):

parse.append(l)

for line in parse:

if line[-1:]=='{':

tab\_count+=1

json\_final+=" "\*tab\_count

json\_final+=line+'\n'

if line=='}':

tab\_count-=1

fileResult.write(json\_final)

fileObj.close()

fileResult.close()

print("--- %s seconds ---" % (time.time() - start\_time))

#ParseToText

start\_time = time.time()

for i in range(10):

fileObj = codecs.open(r"C:\edu\inf\lab4\Thursday schedule.xml", "r", "utf\_8\_sig")

fileResult = open(r"C:\edu\inf\lab4\result.txt", "w")

parse\_raw = []

j = -1

for line in fileObj:

for i in range(len(line) - 6):

if (line[i] + line[i + 1] + line[i + 2] + line[i + 3] + line[i + 4] + line[i + 5] == '<para>'):

parse\_raw.append('')

j += 1

l = i + 6

while (line[l] + line[l + 1] + line[l + 2] + line[l + 3] + line[l + 4] + line[l + 5] + line[

l + 6] != '</para>'):

parse\_raw[j] += line[l]

l += 1

parse = []

for i in range(len(parse\_raw)):

for j in range(len(parse\_raw[i])):

if (parse\_raw[i][j] == '>') and (j != len(parse\_raw[i]) - 1):

s = ''

k = j + 1

while (parse\_raw[i][k] != '<'):

s += parse\_raw[i][k]

k += 1

parse\_raw[i] = s

break

for i in parse\_raw:

if (i not in parse) and (i[0] != '<'):

if (i[0] != '$'):

parse.append(i)

else:

parse.append(i[-2:])

for i in range(len(parse)):

fileResult.write(parse[i] + '\n')

fileObj.close()

fileResult.close()

print("--- %s seconds ---" % (time.time() - start\_time))

#ex2

start\_time = time.time()

for i in range(10):

fileObj = codecs.open(r"C:\edu\inf\lab4\Thursday schedule.xml", "r", "utf\_8\_sig")

fileResult = open(r"C:\edu\inf\lab4\result with lib.json", "w")

file = fileObj.read()

obj = xmltodict.parse(file)

fileObj.close()

fileResult.write(json.dumps(obj))

fileResult.close()

print("--- %s seconds ---" % (time.time() - start\_time))

#ex3

start\_time = time.time()

for i in range(10):

fileObj = codecs.open(r"C:\edu\inf\lab4\Thursday schedule.xml", "r", "utf\_8\_sig")

fileResult = open(r"C:\edu\inf\lab4\result with reg.txt", 'w')

file = fileObj.read()

who = re.search(r'[А-Я]\w+\s[А-Я]\w+\s[А-Я]\w+', file).group()

d = re.findall(r'<title>\w+</title>', file)

day = re.search(r'[а-я]\w+', d[0], flags=re.IGNORECASE).group()

when = re.search(r'\d\d:\d\d-\d\d:\d\d', file).group()

where = re.search(r'\d\d\d\s\w+', file).group()

address = re.search(r'[А-Я]\w+\s[а-я]\D+\d+\D+\s', file).group()

science = re.search(r'[А-Я]\w+\s[а-я]\w{5,}', file).group()

result = day + '\n' + when + '\n' + address + '\n' + where + '\n' + who + '\n' + science

fileResult.write(result)

fileObj.close()

fileResult.close()

print("--- %s seconds ---" % (time.time() - start\_time))

# Результаты.

## Исходник:

<?xml version="1.0" encoding="UTF-8"?>

<article lang="">

<sect1>

<sect2><title/><sect3><title>Четверг</title><para><anchor id="\_4day"/></para><informaltable frame="all"><tgroup cols="3"><colspec colnum="1" colname="c1"/><colspec colnum="2" colname="c2"/><colspec colnum="3" colname="c3"/><tbody><row><entry><para>$("#4day\_btn").show();Чт</para></entry><entry><para>13:30-15:00 </para><para>нечетная неделя </para><para>285 ауд. </para><para>Кронверкский пр., д.49, лит.А </para></entry><entry><para>285 ауд. </para><para>Кронверкский пр., д.49, лит.А </para></entry><entry><para>Дискретная математика(Лек)нечетная неделя</para><para><ulink url="https://itmo.ru/ru/viewperson/81/polyakov\_vladimir\_ivanovich.htm">Поляков Владимир Иванович</ulink> </para></entry><entry><para>Очно - дистанционный </para></entry></row></tbody></tgroup></informaltable></sect3></sect2></sect1>

</article>

## Полученные данные:

### Parse To Text:

Чт

13:30-15:00

нечетная неделя

285 ауд.

Кронверкский пр., д.49, лит.А

Дискретная математика(Лек)нечетная неделя

Поляков Владимир Иванович

Очно - дистанционный

### Parse To Json:

"article lang=""":{

"sect1":{

"sect2":{

"title/":

"sect3":{

"title":{

"Четверг"

}

"para":{

"anchor id="\_4day"/":

}

"informaltable frame="all"":{

"tgroup cols="3"":{

"colspec colnum="1" colname="c1"/":

"colspec colnum="2" colname="c2"/":

"colspec colnum="3" colname="c3"/":

"tbody":{

"row":{

"entry":{

"para":{

"$("#4day\_btn").show();Чт"

}

}

"entry":{

"para":{

"13:30-15:00 "

}

"para":{

"нечетная неделя "

}

"para":{

"285 ауд. "

}

"para":{

"Кронверкский пр., д.49, лит.А "

}

}

"entry":{

"para":{

"285 ауд. "

}

"para":{

"Кронверкский пр., д.49, лит.А "

}

}

"entry":{

"para":{

"Дискретная математика(Лек)нечетная неделя"

}

"para":{

"ulink url="https://itmo.ru/ru/viewperson/81/polyakov\_vladimir\_ivanovich.htm"":{

"Поляков Владимир Иванович"

}

}

}

"entry":{

"para":{

"Очно - дистанционный "

}

}

}

}

}

}

}

}

}

}

### Ex 2:

{"article": {"@lang": "", "sect1": {"sect2": {"title": null, "sect3": {"title": "\u0427\u0435\u0442\u0432\u0435\u0440\u0433", "para": {"anchor": {"@id": "\_4day"}}, "informaltable": {"@frame": "all", "tgroup": {"@cols": "3", "colspec": [{"@colnum": "1", "@colname": "c1"}, {"@colnum": "2", "@colname": "c2"}, {"@colnum": "3", "@colname": "c3"}], "tbody": {"row": {"entry": [{"para": "$(\"#4day\_btn\").show();\u0427\u0442"}, {"para": ["13:30-15:00", "\u043d\u0435\u0447\u0435\u0442\u043d\u0430\u044f \u043d\u0435\u0434\u0435\u043b\u044f", "285 \u0430\u0443\u0434.", "\u041a\u0440\u043e\u043d\u0432\u0435\u0440\u043a\u0441\u043a\u0438\u0439 \u043f\u0440., \u0434.49, \u043b\u0438\u0442.\u0410"]}, {"para": ["285 \u0430\u0443\u0434.", "\u041a\u0440\u043e\u043d\u0432\u0435\u0440\u043a\u0441\u043a\u0438\u0439 \u043f\u0440., \u0434.49, \u043b\u0438\u0442.\u0410"]}, {"para": ["\u0414\u0438\u0441\u043a\u0440\u0435\u0442\u043d\u0430\u044f \u043c\u0430\u0442\u0435\u043c\u0430\u0442\u0438\u043a\u0430(\u041b\u0435\u043a)\u043d\u0435\u0447\u0435\u0442\u043d\u0430\u044f \u043d\u0435\u0434\u0435\u043b\u044f", {"ulink": {"@url": "https://itmo.ru/ru/viewperson/81/polyakov\_vladimir\_ivanovich.htm", "#text": "\u041f\u043e\u043b\u044f\u043a\u043e\u0432 \u0412\u043b\u0430\u0434\u0438\u043c\u0438\u0440 \u0418\u0432\u0430\u043d\u043e\u0432\u0438\u0447"}}]}, {"para": "\u041e\u0447\u043d\u043e - \u0434\u0438\u0441\u0442\u0430\u043d\u0446\u0438\u043e\u043d\u043d\u044b\u0439"}]}}}}}}}}}

### Ex 3:

Четверг

13:30-15:00

Кронверкский пр., д.49, лит.А

285 ауд

Поляков Владимир Иванович

Дискретная математика

### Ex 4:

--- 0.00798940658569336 seconds ---

--- 0.011000394821166992 seconds ---

--- 0.004997730255126953 seconds ---

--- 0.0030035972595214844 seconds ---