1. Создайте XML-файл, который содержит результаты вступительных испытаний   
абитуриентов по математике, русскому языку и информатике. Шифр абитуриента   
является атрибутом. Фамилия и названия вступительных испытаний тегами.   
Определить:   
1.1. Средний балл для какого-нибудь студента по фамилии.   
1.2. Максимальный и минимальный балл по каждому вступительному испытанию для всех абитуриентов   
1.3. ФИО студентов, у которых сумма баллов больше 250.   
1.4. ФИО студентов, у которых по каждому вступительному испытанию больше 50 баллов.

Xml file:

<enrolles>

    <enrolle id = '1'>

        <name>Gavrilushkin</name>

        <math>74</math>

        <russian>78</russian>

        <informatics>84</informatics>

    </enrolle>

    <enrolle id = '2'>

        <name>Sidelev</name>

        <math>0</math>

        <russian>95</russian>

        <informatics>50</informatics>

    </enrolle>

    <enrolle id = '3'>

        <name>Visheneev</name>

        <math>50</math>

        <russian>50</russian>

        <informatics>50</informatics>

    </enrolle>

    <enrolle id = '4'>

        <name>Bogoslovskiy</name>

        <math>100</math>

        <russian>100</russian>

        <informatics>100</informatics>

    </enrolle>

</enrolles>

Code:

import xml.etree.ElementTree as ET

tree = ET.parse('лаб 3/1.xml')

root = tree.getroot()

students\_info = {}

for enrolle in root.findall("enrolle"):

        students\_info[enrolle.find('name').text] = [int(enrolle.findtext('math')),

                                                    int(enrolle.findtext('russian')),

                                                    int(enrolle.findtext('informatics'))]

over\_50 = []

over\_250 = []

for student in students\_info.keys():

        scores = students\_info[student]

        scores.sort()

        if sum(scores) > 50: over\_50.append(student)

        if sum(scores) > 250: over\_250.append(student)

        print(student, 'avg score =', sum(scores)/len(scores))

        print(student, 'min score =', scores[0])

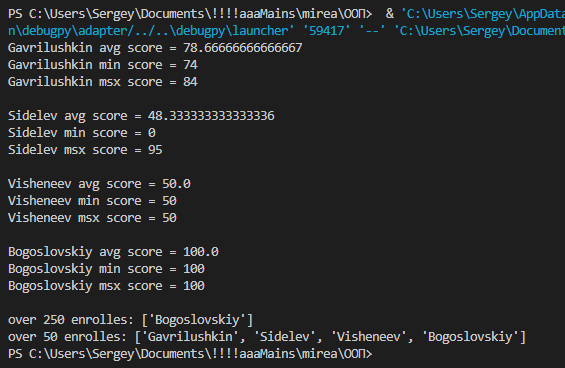
        print(student, 'msx score =', scores[-1])

        print()

print('over 250 enrolles:', over\_250)

print('over 50 enrolles:', over\_50)

result:



2. Создайте структуру XML-файла: фамилия, имя, отчество студента; группа, название   
дисциплины; общее количество выполненных работ; количество защищенных   
(сданных) работ.   
2.1. Напечатать список студентов, определенной группы в формате: Фамилия И.О.   
2.2.Определить общее количество выполненных, но не сданных работ для заданного   
студента по всем дисциплинам.   
2.3.Вывести процент готовности студента к сессии.   
2.4. Определить разницу в процентах между количеством выполненных работ и   
количеством защищенных работ для заданного студента по каждой дисциплине.   
2.5.Вывести дисциплины для заданного студента, где разница между количеством   
выполненных работ и количеством защищенных работ составляет более 50%.

Xml file:

<students>

    <student name = 'Гаврилушкин Сергей Андреевич' group = 'ТШБО-01-22'>

        <subject name = 'Информатика'>

            <tasks>5</tasks>

            <tasks\_done>3</tasks\_done>

        </subject>

        <subject name = 'Физика'>

            <tasks>2</tasks>

            <tasks\_done>1</tasks\_done>

        </subject>

        <subject name = 'Линейная алгебра'>

            <tasks>10</tasks>

            <tasks\_done>9</tasks\_done>

        </subject>

        <subject name = 'Математический анализ'>

            <tasks>8</tasks>

            <tasks\_done>3</tasks\_done>

        </subject>

        <subject name = 'Философия'>

            <tasks>10</tasks>

            <tasks\_done>0</tasks\_done>

        </subject>

        <subject name = 'История'>

            <tasks>12</tasks>

            <tasks\_done>4</tasks\_done>

        </subject>

    </student>

    <student name = 'Вышенеев Ярослав Германович' group = 'ТШБО-01-22'>

        <subject name = 'Информатика'>

            <tasks>5</tasks>

            <tasks\_done>3</tasks\_done>

        </subject>

        <subject name = 'Физика'>

            <tasks>2</tasks>

            <tasks\_done>1</tasks\_done>

        </subject>

        <subject name = 'Линейная алгебра'>

            <tasks>10</tasks>

            <tasks\_done>9</tasks\_done>

        </subject>

        <subject name = 'Математический анализ'>

            <tasks>8</tasks>

            <tasks\_done>3</tasks\_done>

        </subject>

        <subject name = 'Философия'>

            <tasks>10</tasks>

            <tasks\_done>0</tasks\_done>

        </subject>

        <subject name = 'История'>

            <tasks>12</tasks>

            <tasks\_done>4</tasks\_done>

        </subject>

    </student>

    <student name = 'Иванов Иван Иванович' group = 'ТШБО-01-23'>

        <subject name = 'Информатика'>

            <tasks>5</tasks>

            <tasks\_done>3</tasks\_done>

        </subject>

        <subject name = 'Физика'>

            <tasks>2</tasks>

            <tasks\_done>1</tasks\_done>

        </subject>

        <subject name = 'Линейная алгебра'>

            <tasks>10</tasks>

            <tasks\_done>9</tasks\_done>

        </subject>

        <subject name = 'Математический анализ'>

            <tasks>8</tasks>

            <tasks\_done>3</tasks\_done>

        </subject>

        <subject name = 'Философия'>

            <tasks>10</tasks>

            <tasks\_done>0</tasks\_done>

        </subject>

        <subject name = 'История'>

            <tasks>12</tasks>

            <tasks\_done>4</tasks\_done>

        </subject>

    </student>

    <student name = 'Иванов Андрей Васильевич' group = 'ТШБО-01-22'>

        <subject name = 'Информатика'>

            <tasks>5</tasks>

            <tasks\_done>3</tasks\_done>

        </subject>

        <subject name = 'Физика'>

            <tasks>2</tasks>

            <tasks\_done>1</tasks\_done>

        </subject>

        <subject name = 'Линейная алгебра'>

            <tasks>10</tasks>

            <tasks\_done>9</tasks\_done>

        </subject>

        <subject name = 'Математический анализ'>

            <tasks>8</tasks>

            <tasks\_done>3</tasks\_done>

        </subject>

        <subject name = 'Философия'>

            <tasks>10</tasks>

            <tasks\_done>0</tasks\_done>

        </subject>

        <subject name = 'История'>

            <tasks>12</tasks>

            <tasks\_done>4</tasks\_done>

        </subject>

    </student>

</students>

Code:

import xml.etree.ElementTree as ET

tree = ET.parse('лаб 3/2.xml')

root = tree.getroot()

students\_in\_group\_1 = []

for student in root.findall("student[@group = 'ТШБО-01-22']"):

    name = student.get('name').split(' ')

    students\_in\_group\_1.append(name[0] +' '+ name[1][0] +'. '+ name[2][0] +'.')

print('Студенты в группе ТШБО-01-22:', students\_in\_group\_1)

tasks\_counter = 0

tasks\_done\_counter = 0

subject\_tasks = {}

for subject in root.findall('student[@name = "Гаврилушкин Сергей Андреевич"]/subject'):

    tasks = int(subject.findtext('tasks'))

    tasks\_done = int(subject.findtext('tasks\_done'))

    tasks\_counter += tasks

    tasks\_done\_counter += tasks\_done

    subject\_tasks[subject.get('name')] = (tasks, tasks\_done)

print('Предмет, невыполнено задач')

subjects\_over\_50 = []

for key in subject\_tasks:

    if subject\_tasks[key][1] / subject\_tasks[key][0] > 0.5: subjects\_over\_50.append(key)

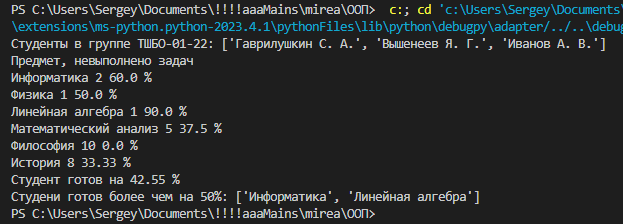
    print(key, subject\_tasks[key][0] - subject\_tasks[key][1],

          round(subject\_tasks[key][1] / subject\_tasks[key][0] \* 100,2), '%')

print('Студент готов на', round(tasks\_done\_counter/tasks\_counter\*100, 2), '%')

print('Студени готов более чем на 50%:', subjects\_over\_50)

result:



3. Используя XML-файла из п.2. требуется напечатать список студентов с указанием для каждого студента количества его однофамильцев (повышенный уровень).

Code:

import xml.etree.ElementTree as ET

tree = ET.parse('лаб 3/2.xml')

root = tree.getroot()

d = {}

names = []

for student in root.findall('student'):

    names.append(student.get('name').split(' ')[0])

for i in set(names):

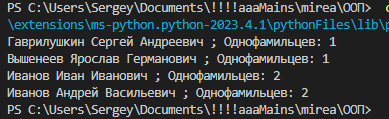
    d[i] = names.count(i)

for student in root.findall('student'):

    name = student.get('name')

    print(name, '; Однофамильцев:', d[name.split(' ')[0]])

Result:



4. Организуйте вывод данных в таблицу   
4.1. Подключите библиотеку:   
from prettytable import PrettyTable   
4.2. Создайте таблицу:   
x = PrettyTable()   
4.3.Добавьте заголовки:   
x.field\_names = ["ID", "Title", "Author"]   
4.4. Используя цикл, добавьте в таблицу новые строки:   
x.add\_row([id, title, author])   
4.5. Выводите таблицу, используя print.

Xml file:

<Catalog>

    <Books>

        <Book id="1" price="550">

            <Title>Война и мир</Title>

            <Author>Л.Н. Толстой</Author>

        </Book>

        <Book id="2" price="350">

            <Title>Мертвые души</Title>

            <Author>Н.В. Гоголь</Author>

        </Book>

        <Book id="3" price="470">

            <Title>Евгений Онегин</Title>

            <Author>А.С. Пушкин</Author>

        </Book>

    </Books>

</Catalog>

Code:

from prettytable import PrettyTable

import xml.etree.ElementTree as ET

tree = ET.parse('лаб 3/test.xml')

root = tree.getroot()

x = PrettyTable()

x.field\_names = ['ID', 'Title', 'Author']

for book in root.findall('Books/Book'):

    x.add\_row([book.get('id'), book.findtext('Title'), book.findtext('Author')])

print(x)

Result:

