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.

<Catalog>

    <Students>

        <Student id="1">

            <Surname>Minakov</Surname>

            <Exams>

                <Math>90</Math>

                <RusLang>100</RusLang>

                <Inform>75</Inform>

            </Exams>

        </Student>

        <Student id="2">

            <Surname>Emelyanov</Surname>

            <Exams>

                <Math>35</Math>

                <RusLang>90</RusLang>

                <Inform>82</Inform>

            </Exams>

        </Student>

        <Student id="3">

            <Surname>Kovalchuk</Surname>

            <Exams>

                <Math>70</Math>

                <RusLang>34</RusLang>

                <Inform>100</Inform>

            </Exams>

        </Student>

    </Students>

</Catalog>

import xml.etree.ElementTree as ET

class student():

    Surname = str()

    Math = int()

    Infa = int()

    RusLang = int()

    def \_\_init\_\_(self, IDS):

        tree = ET.parse('C:/PyProgs/Lab3/tsst.xml') #Парсим

        root = tree.getroot() #Присваеваем Адрес

        STRFORM = "Students/Student[@id='{0}']".format(IDS) #Форматируем строку для поиска

        for student in root.findall(STRFORM): #Ищем

            self.Surname = student.find('Surname').text

            for exam in student.findall("Exams"): #Ищем в подкатологе Exams

                self.Math = int(exam.find('Math').text)

                self.Infa = int(exam.find('Inform').text)

                self.RusLang = int(exam.find('RusLang').text)

    def MidScore(self):

        return((self.Math + self.Infa + self.RusLang)//3)

    def MaxScore(self):

        List = [self.Math, self.Infa, self.RusLang]

        return (max(List))

    def MinScore(self):

        List = [self.Math, self.Infa, self.RusLang]

        return (min(List))

    def Summary(self):

        return int(self.Math + self.Infa + self.RusLang)

Stud1 = student(1)

Stud2 = student(2)

Stud3 = student(3)

Stud0 = [Stud1, Stud2, Stud3]

for i in range(0,2):

    print(Stud0[i].Surname,':')

    print('Средний Балл: ', Stud0[i].MidScore())

    print('Макс. Балл: ', Stud0[i].MaxScore())

    print('Мин. Балл: ', Stud0[i].MinScore())

print('Абитуриенты with 250+ баллов:')

for i in range(0,2):

    if Stud0[i].Summary() > 250:

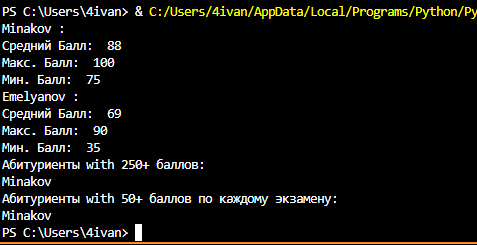
        print(Stud0[i].Surname)

print('Абитуриенты with 50+ баллов по каждому экзамену:')

for i in range(0,2):

    if (Stud0[i].Math > 50) and (Stud0[i].RusLang > 50) and (Stud0[i].Infa > 50):

        print(Stud0[i].Surname)



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%.

<Catalog>

    <Groups>

        <Group GNum = 'ТШБО-01-22'>

            <Student name = 'Сергей' surname = 'Гаврилушкин' midname = 'Андреевич' ID = '1'>

                <Subject SName = 'Математический анализ'>

                    <Tasks>34</Tasks>

                    <Tasks\_Done>12</Tasks\_Done>

                </Subject>

                <Subject SName = 'Английский язык'>

                    <Tasks>12</Tasks>

                    <Tasks\_Done>5</Tasks\_Done>

                </Subject>

                <Subject SName = 'Философия'>

                    <Tasks>25</Tasks>

                    <Tasks\_Done>10</Tasks\_Done>

                </Subject>

                <Subject SName = 'История'>

                    <Tasks>9</Tasks>

                    <Tasks\_Done>1</Tasks\_Done>

                </Subject>

                <Subject SName = 'Линейная алгебра'>

                    <Tasks>54</Tasks>

                    <Tasks\_Done>0</Tasks\_Done>

                </Subject>

            </Student>

            <Student name = 'Сергей' surname = 'Радаев' midname = 'Дмитриевич' ID = '2'>

                <Subject SName = 'Математический анализ'>

                    <Tasks>34</Tasks>

                    <Tasks\_Done>22</Tasks\_Done>

                </Subject>

                <Subject SName = 'Английский язык'>

                    <Tasks>12</Tasks>

                    <Tasks\_Done>10</Tasks\_Done>

                </Subject>

                <Subject SName = 'Философия'>

                    <Tasks>25</Tasks>

                    <Tasks\_Done>20</Tasks\_Done>

                </Subject>

                <Subject SName = 'История'>

                    <Tasks>9</Tasks>

                    <Tasks\_Done>8</Tasks\_Done>

                </Subject>

                <Subject SName = 'Линейная алгебра'>

                    <Tasks>54</Tasks>

                    <Tasks\_Done>25</Tasks\_Done>

                </Subject>

            </Student>

            <Student name = 'Иван' surname = 'Косыгин' midname = 'Алексеевич' ID = '3'>

                <Subject SName = 'Математический анализ'>

                    <Tasks>34</Tasks>

                    <Tasks\_Done>34</Tasks\_Done>

                </Subject>

                <Subject SName = 'Английский язык'>

                    <Tasks>12</Tasks>

                    <Tasks\_Done>0</Tasks\_Done>

                </Subject>

                <Subject SName = 'Философия'>

                    <Tasks>25</Tasks>

                    <Tasks\_Done>0</Tasks\_Done>

                </Subject>

                <Subject SName = 'История'>

                    <Tasks>9</Tasks>

                    <Tasks\_Done>9</Tasks\_Done>

                </Subject>

                <Subject SName = 'Линейная алгебра'>

                    <Tasks>54</Tasks>

                    <Tasks\_Done>53</Tasks\_Done>

                </Subject>

            </Student>

            <Student name = 'Николай' surname = 'Маслов' midname = 'Иванович' ID = '4'>

                <Subject SName = 'Математический анализ'>

                    <Tasks>34</Tasks>

                    <Tasks\_Done>0</Tasks\_Done>

                </Subject>

                <Subject SName = 'Английский язык'>

                    <Tasks>12</Tasks>

                    <Tasks\_Done>10</Tasks\_Done>

                </Subject>

                <Subject SName = 'Философия'>

                    <Tasks>25</Tasks>

                    <Tasks\_Done>25</Tasks\_Done>

                </Subject>

                <Subject SName = 'История'>

                    <Tasks>9</Tasks>

                    <Tasks\_Done>5</Tasks\_Done>

                </Subject>

                <Subject SName = 'Линейная алгебра'>

                    <Tasks>54</Tasks>

                    <Tasks\_Done>54</Tasks\_Done>

                </Subject>

            </Student>

        </Group>

        <Group GNum = 'ТШБО-02-22'>

            <Student name = 'Андрей' surname = 'Марков' midname = 'Кирилович' ID = '1'>

                <Subject SName = 'Математический анализ'>

                    <Tasks>34</Tasks>

                    <Tasks\_Done>18</Tasks\_Done>

                </Subject>

                <Subject SName = 'Английский язык'>

                    <Tasks>12</Tasks>

                    <Tasks\_Done>10</Tasks\_Done>

                </Subject>

                <Subject SName = 'Философия'>

                    <Tasks>25</Tasks>

                    <Tasks\_Done>19</Tasks\_Done>

                </Subject>

                <Subject SName = 'История'>

                    <Tasks>9</Tasks>

                    <Tasks\_Done>9</Tasks\_Done>

                </Subject>

                <Subject SName = 'Линейная алгебра'>

                    <Tasks>54</Tasks>

                    <Tasks\_Done>10</Tasks\_Done>

                </Subject>

            </Student>

            <Student name = 'Ростислав' surname = 'Орский' midname = 'Алексеевич' ID = '2'>

                <Subject SName = 'Математический анализ'>

                    <Tasks>34</Tasks>

                    <Tasks\_Done>22</Tasks\_Done>

                </Subject>

                <Subject SName = 'Английский язык'>

                    <Tasks>12</Tasks>

                    <Tasks\_Done>10</Tasks\_Done>

                </Subject>

                <Subject SName = 'Философия'>

                    <Tasks>25</Tasks>

                    <Tasks\_Done>20</Tasks\_Done>

                </Subject>

                <Subject SName = 'История'>

                    <Tasks>9</Tasks>

                    <Tasks\_Done>8</Tasks\_Done>

                </Subject>

                <Subject SName = 'Линейная алгебра'>

                    <Tasks>54</Tasks>

                    <Tasks\_Done>25</Tasks\_Done>

                </Subject>

            </Student>

            <Student name = 'Алексей' surname = 'Косыгин' midname = 'Ивановичвич' ID = '3'>

                <Subject SName = 'Математический анализ'>

                    <Tasks>34</Tasks>

                    <Tasks\_Done>34</Tasks\_Done>

                </Subject>

                <Subject SName = 'Английский язык'>

                    <Tasks>12</Tasks>

                    <Tasks\_Done>0</Tasks\_Done>

                </Subject>

                <Subject SName = 'Философия'>

                    <Tasks>25</Tasks>

                    <Tasks\_Done>0</Tasks\_Done>

                </Subject>

                <Subject SName = 'История'>

                    <Tasks>9</Tasks>

                    <Tasks\_Done>9</Tasks\_Done>

                </Subject>

                <Subject SName = 'Линейная алгебра'>

                    <Tasks>54</Tasks>

                    <Tasks\_Done>53</Tasks\_Done>

                </Subject>

            </Student>

        </Group>

    </Groups>

</Catalog>

import xml.etree.ElementTree as ET

class SubjectClass():

    def \_\_init\_\_(self, name, Tasks, TD):

        self.Name = name

        self.Task = int(Tasks)

        self.TaskDone = int(TD)

class student():

    Surname = str()

    Midame = str()

    ID = str()

    SubjectList = []

    def \_\_init\_\_(self, name, surname, midname, ID):

        self.Name = name

        self.Surname = surname

        self.Midame = midname

        self.ID = ID

        self.SubjectList = []

class group():

    def \_\_init\_\_(self, Name):

        self.Name = Name

        self.GroupList = []

tree = ET.parse('C:/PyProgs/Lab3/tsst2.xml')

root = tree.getroot()

SuperList = [group(Groups.get("GNum")) for Groups in root.findall("Groups/Group")]

for GroupObj in SuperList:

    Formattext = "Groups/Group[@GNum='{0}']/Student".format(GroupObj.Name)

    GroupObj.GroupList = [student(Students.get("name"), Students.get("surname"), Students.get("midname"), Students.get("ID")) for Students in root.findall(Formattext)]

    for StudObj in GroupObj.GroupList:

        Formattext2 = "Groups/Group[@GNum='{0}']/Student[@ID='{1}']/Subject".format(GroupObj.Name, StudObj.ID)

        StudObj.SubjectList = [SubjectClass(Subjects.get('SName'), Subjects.find("Tasks").text, Subjects.find("Tasks\_Done").text) for Subjects in root.findall(Formattext2)]

print('2.1')

for GroupObj in SuperList:

    print(GroupObj.Name,': ')

    for Studobj in GroupObj.GroupList:

        print(Studobj.Surname, Studobj.Name, Studobj.Midame)

print('2.2, 2.3, 2.4')

print(SuperList[0].GroupList[2].Surname)

TaskAll = 0

TaskDoneAll = 0

for SubjObj in SuperList[0].GroupList[2].SubjectList:

   PercentSub = SubjObj.TaskDone/SubjObj.Task\*100

   print(SubjObj.Name,': ', SubjObj.Task,'работ сделано', PercentSub,'Процент готовности по предмету')

   TaskAll = TaskAll + SubjObj.Task

   TaskDoneAll = TaskDoneAll + SubjObj.TaskDone

print(TaskDoneAll/TaskAll\*100,' общий процент готовности')

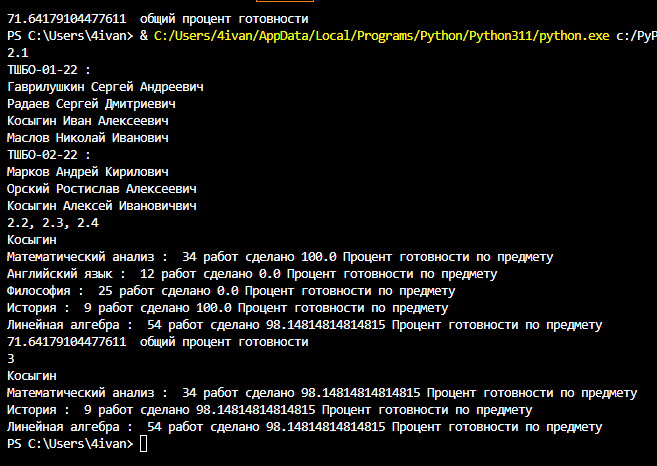
print(3)

print(SuperList[0].GroupList[2].Surname)

for SubjObj in SuperList[0].GroupList[2].SubjectList:

   if SubjObj.TaskDone/SubjObj.Task\*100 > 50 :

    print(SubjObj.Name,': ', SubjObj.Task,'работ сделано', PercentSub,'Процент готовности по предмету')

3.

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

import xml.etree.ElementTree as ET

class SubjectClass():

    def \_\_init\_\_(self, name, Tasks, TD):

        self.Name = name

        self.Task = int(Tasks)

        self.TaskDone = int(TD)

class student():

    Surname = str()

    Midame = str()

    ID = str()

    SubjectList = []

    def \_\_init\_\_(self, name, surname, midname, ID):

        self.Name = name

        self.Surname = surname

        self.Midame = midname

        self.ID = ID

        self.SubjectList = []

class group():

    def \_\_init\_\_(self, Name):

        self.Name = Name

        self.GroupList = []

tree = ET.parse("C:/Lab3/tsst2.xml")

root = tree.getroot()

SuperList = [group(Groups.get("GNum")) for Groups in root.findall("Groups/Group")]

for GroupObj in SuperList:

    Formattext = "Groups/Group[@GNum='{0}']/Student".format(GroupObj.Name)

    GroupObj.GroupList = [student(Students.get("name"), Students.get("surname"), Students.get("midname"), Students.get("ID")) for Students in root.findall(Formattext)]

    for StudObj in GroupObj.GroupList:

        Formattext2 = "Groups/Group[@GNum='{0}']/Student[@ID='{1}']/Subject".format(GroupObj.Name, StudObj.ID)

        StudObj.SubjectList = [SubjectClass(Subjects.get('SName'), Subjects.find("Tasks").text, Subjects.find("Tasks\_Done").text) for Subjects in root.findall(Formattext2)]

print('3')

for GroupObj in SuperList:

    print(GroupObj.Name,': ')

    for Studobj in GroupObj.GroupList:

        i = 0

        print(Studobj.Surname, Studobj.Name, Studobj.Midame)

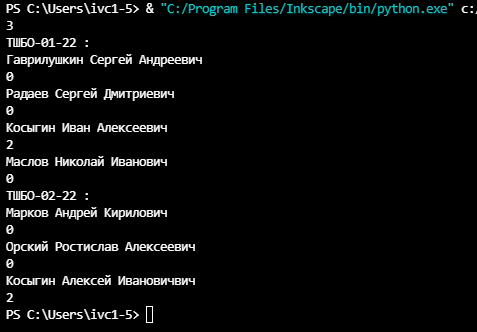
        for GroupObj2 in SuperList:

            for Studobj2 in GroupObj2.GroupList:

                if (StudObj.Surname == Studobj2.Surname) and ((Studobj.ID == Studobj2.ID)):

                    i = i + 1

        print(i)



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.

from prettytable import PrettyTable

import xml.etree.ElementTree as ET

tree = ET.parse('C:/Lab3/tsst.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)