Маратулы Темирболат

Написать программу, с классом «Студент» (с полями (Фамилия, Имя, Отчество, ИИН)). Программа должна принимать от пользователя список студентов, сохранять его в файле students.txt, а так же считывать этот файл и выводить н экран список студентов.

ONE\_STEP = 1

class Student:

    def \_\_init\_\_(self,first\_name,last\_name,middle\_name,id):

        self.\_\_first\_name = first\_name

        self.\_\_last\_name = last\_name

        self.\_\_middle\_name = middle\_name

        self.\_\_id = id

    def get\_first\_name(self):

        return self.\_\_first\_name

    def set\_first\_name(self,first\_name):

        self.\_\_first\_name = first\_name

    def get\_last\_name(self):

        return self.\_\_last\_name

    def set\_last\_name(self,last\_name):

        self.\_\_last\_name = last\_name

    def get\_middle\_name(self):

        return self.\_\_middle\_name

    def set\_middle\_name(self,middle\_name):

        self.\_\_middle\_name = middle\_name

    def get\_id(self):

        return self.\_\_id

    def set\_id(self,id):

        self.\_\_id = id

    def \_\_str\_\_(self):

        return 'ФИО студента: "{} {} {}". ИИН: "{}"\

'.format(self.\_\_last\_name,self.\_\_first\_name,self.\_\_middle\_name,self.\_\_id)

students = []

file\_students = open(r"students.txt","a+")

def show\_menu\_options():

    print('\n1. Создать нового студента\n\

2. Просмотреть всех существующих студентов\n\

3. Выйти из программы')

def get\_user\_option():

    return int(input('Ваш выбор: '))

def get\_student\_first\_name():

    return input('Введите Имя студента: ')

def get\_student\_last\_name():

    return input('Введите Фамилию студента: ')

def get\_student\_middle\_name():

    return input('Введите Отчество студента: ')

def get\_student\_id():

    return int(input('Введите ИИН студента: '))

def create\_new\_student():

    student\_first\_name = get\_student\_first\_name()

    student\_last\_name = get\_student\_last\_name()

    student\_middle\_name = get\_student\_middle\_name()

    student\_id = get\_student\_id()

    student = Student(student\_first\_name,student\_last\_name,student\_middle\_name,student\_id)

    students.append(student)

    student\_full\_name = student\_last\_name + ' ' + student\_first\_name + ' ' + student\_middle\_name

    student\_id = str(student\_id)

    student\_info = student\_full\_name + ',' + student\_id + "\n"

    file\_students.write(student\_info)

    print('\nСтудент успешно создан!')

def load\_students():

    student\_lines = []

    with open("students.txt") as f:

        student\_lines = f.readlines()

    for line in student\_lines:

        student\_info = line.split(",")

        student\_full\_name = student\_info[0].split(" ")

        student\_last\_name = student\_full\_name[0]

        student\_first\_name = student\_full\_name[1]

        student\_middle\_name = student\_full\_name[2]

        student\_id = student\_info[1].replace("\n","")

        student = Student(student\_first\_name,student\_last\_name,student\_middle\_name,student\_id)

        students.append(student)

def view\_all\_students():

    current\_student = 1

    for student in students:

        print(current\_student,student)

        current\_student += ONE\_STEP

menu\_functions = [create\_new\_student,view\_all\_students]

def main\_menu():

    while True:

        try:

            show\_menu\_options()

            user\_choice = get\_user\_option()

            if(user\_choice < 1 or user\_choice > len(menu\_functions) + ONE\_STEP):

                print('\nИзвините, но такой опции нет!')

                continue

            elif(user\_choice == len(menu\_functions) + ONE\_STEP):

                print('\nСпасибо за использование программы! До свидания!')

                break

            menu\_functions[user\_choice - ONE\_STEP]()

        except ValueError:

            print('Извините, но вы ввели не число!')

        except Exception as ex:

            print(type(ex))

            print(ex)

load\_students()

main\_menu()

file\_students.close()





