Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития

Кафедра инфокоммуникаций

**ОТЧЕТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ №2**

**дисциплины «Анализ данных»**

|  |  |  |  |
| --- | --- | --- | --- |
|  | | Выполнил:  Кожуховский Виктор Андреевич  2 курс, группа ИВТ-б-о-22-1,  09.03.01 «Информатика и вычислительная техника», направленность (профиль) «Программное обеспечение средств вычислительной техники и автоматизированных систем  », очная форма обучения  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) | |
|  | | Руководитель практики:  Воронкин Роман Александрович  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) | |
|  | |  | |

Отчет защищен с оценкой \_\_\_\_\_\_\_\_\_\_\_ Дата защиты\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ставрополь, 2024 г.

Порядок выполнения работы:

1. Изучил теоретический материал работы.

2. Создал общедоступный репозиторий на GitHub, в котором использована лицензия MIT и язык программирования Python.

3. Выполнил клонирование созданного репозитория.

4. Дополнил файл .gitignore необходимыми правилами для работы с IDE PyCharm.

5. Организовал свой репозиторий в соответствие с моделью ветвления git-flow.

6. Создал проект в папке репозитория.

7. Проработал пример лабораторной работы.

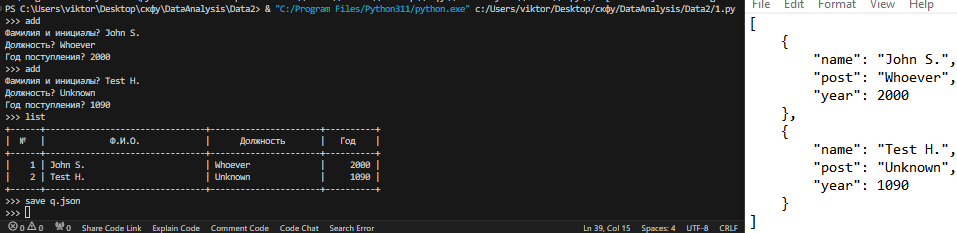


Рисунок 1. Выполнение примера 1

8. Выполнил индивидуальное задание и задание повышенной сложности.

Для своего варианта лабораторной работы 2.8 необходимо дополнительно реализовать сохранение и чтение данных из файла формата JSON. Необходимо также проследить за тем, чтобы файлы генерируемый этой программой не попадали в репозиторий лабораторной работы.



Рисунок 2. Код решения индивидуального задания 1 и его выполнение

Очевидно, что программа в примере 1 и в индивидуальном задании никак не проверяет правильность загружаемых данных формата JSON. В следствие чего, необходимо после загрузки из файла JSON выполнять валидацию загруженных данных. Валидацию данных необходимо производить с использованием спецификации JSON Schema, описанной на сайте https://json-schema.org/. Одним из возможных вариантов работы с JSON Schema является использование пакета jsonschema, который не является частью стандартной библиотеки Python. Таким образом, необходимо реализовать валидацию загруженных данных с помощью спецификации JSON Schema.

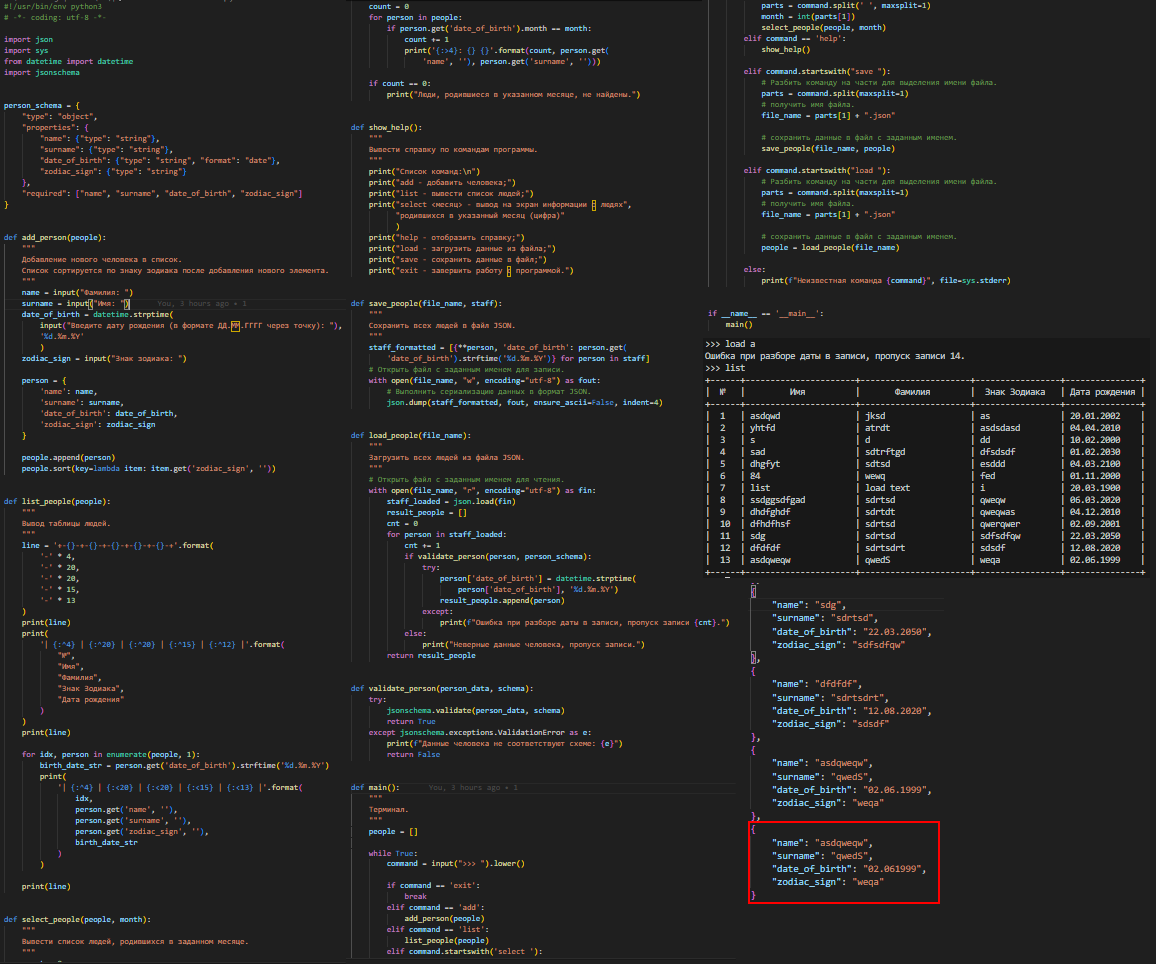


Рисунок 3. Код решения задания повышенной сложности и его выполнение

9. Зафиксировал сделанные изменения в репозитории.

10. Добавил отчет по лабораторной работе в формате PDF в папку doc репозитория. Зафиксировал изменения.

11. Выполнил слияние ветки для разработки с веткой master/main.

12. Отправил сделанные изменения на сервер GitHub.

Контрольные вопросы:

1. Для чего используется JSON?

JSON используется для обмена данными между приложениями. Это легкий формат обмена данных, который удобен для чтения и записи как людьми, так и компьютерами.

1. Какие типы значений используются в JSON?

В JSON используются следующие типы значений: строки (в двойных кавычках), числа, логические значения (true/false), массивы (списки значений), объекты (связанные пары ключ-значение), null.

1. Как организована работа со сложными данными в JSON?

Для работы со сложными данными в JSON используется структура объектов, массивов и вложенных объектов, позволяющая организовать информацию иерархически.

1. Самостоятельно ознакомьтесь с форматом данных JSON5? В чем отличие этого формата от формата данных JSON?

Формат данных JSON5 - это расширение JSON, поддерживающее комментарии, необязательные запятые в конце элементов массива и объекта, поддержку некоторых дополнительных типов данных, таких как даты. Основное отличие от JSON - это более гибкий и читаемый синтаксис.

1. Какие средства языка программирования Python могут быть использованы для работы с данными в формате JSON5?

Для работы с данными в формате JSON5 в Python можно использовать библиотеку json5 или другие сторонние библиотеки, которые поддерживают этот формат.

1. Какие средства предоставляет язык Python для сериализации данных в формате JSON?

В Python для сериализации данных в формат JSON используются встроенная библиотека json и метод json.dumps().

1. В чем отличие функций json.dump() и json.dumps()?

json.dump() используется для записи данных в поток (например, файл), а json.dumps() используется для преобразования данных в строку.

1. Какие средства предоставляет язык Python для десериализации данных из формата JSON?

Для десериализации данных из формата JSON в Python используется метод json.loads() и библиотека json.

1. Какие средства необходимо использовать для работы с данными формата JSON, содержащими кирилицу?

Для работы с данными формата JSON, содержащими кириллицу, необходимо использовать кодировку UTF-8 при чтении и записи данных с помощью библиотеки json в Python.

1. Самостоятельно ознакомьтесь со спецификацией JSON Schema? Что такое схема данных? Приведите схему данных для примера 1.

Схема данных представляет собой описание ожидаемой структуры JSON, включая типы данных, ограничения и правила валидации, которые должны соответствовать данным.

Схема данных для примера 1:

{

"type": "object",

"properties": {

"name": {"type": "string"},

"post": {"type": "string"},

"year": {"type": "integer"}

},

"required": ["name", "post", "year"]

}