**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный университет**

**Кафедра прикладной математики и процессов управления.**

**отчет**

**по лабораторной работе №2**

**по дисциплине «Функциональное программирование»**

**на тему «Решение задач при помощи функций map, filter, zip, reduce»**

|  |  |  |
| --- | --- | --- |
| Студент гр. 22.Б16-пу |  | Поляков Л.С. |
| Преподаватель |  | Киямов Ж. У. |

**Санкт-Петербург**

**2023 г.**

**Оглавление**

[**1.** **Введение** 3](#_Toc144847601)

[**2.** **Цель работы** 3](#_Toc144847602)

[**3.** **Задача** 3](#_Toc144847603)

[**4.** **Описание программы** 3](#_Toc144847604)

[**5.** **Рекомендации пользователю** 4](#_Toc144847605)

[**6.** **Рекомендации программисту** 4](#_Toc144847607)

[**7.** **Контрольный пример** 5](#_Toc144847608)

[**8.** **Вывод** 7](#_Toc144847611)

**1. Введение**

В данном отчете рассматриваются три программы, решающие различные задачи с использованием функций *map* и *filter*. Программы разработаны для обработки данных студентов, пользователей и заказов. Каждая из них выполняет определенные операции для анализа данных и предоставления соответствующей информации.

**2. Цель работы**

Целью работы является разработка программ, которые позволяют выполнять различные операции с данными, используя функции *map* и *filter*. Эти программы предназначены для обработки и анализа информации о студентах, пользователях и заказах с целью фильтрации, агрегации и анализа данных.

**3. Задачи:**

Основные задачи, решаемые в рамках этих программ, включают в себя:

* Фильтрация данных в соответствии с заданными критериями.
* Преобразование данных для расчета средних оценок, общих сумм расходов и других характеристик.
* Агрегация данных для поиска студента (пользователя) с максимальной оценкой (расходами) и другими агрегированными показателями

**4. Описание программ**

Программы написаны на языке Python и включает в себя следующий ряд компонентов:  
• Функции задачи 1:

|  |  |
| --- | --- |
| *filter\_students\_by\_age* | Фильтрация студентов определенного возраста |
| *filter\_students\_by\_subjects* | Фильтрация студентов, которые набрали более 85 баллов по всем предметам |
| *calculate\_student\_average* | Вычисление среднего балла для каждого студента |
| *calculate\_overall\_average* | Вычисление общего среднего балла по всем студентам |
| *find\_top\_student* | Нахождение студента с самым высоким средним баллом |

• Функции задачи 2:

|  |  |
| --- | --- |
| *criteria* | Фильтрация пользователей в соответствии с заданными критериями |
| *total\_expenses\_per\_user* | Рассчет общей суммы расходов для каждого пользователя |

• Функции задачи 3:

|  |  |
| --- | --- |
| *filter\_orders* | Фильтрация заказов только для определенного клиента |
| *calculate\_total\_amount* | Подсчет общей суммы заказов для данного клиента |
| *calculate\_average\_amount* | Подсчет средней стоимости заказов для данного клиента |

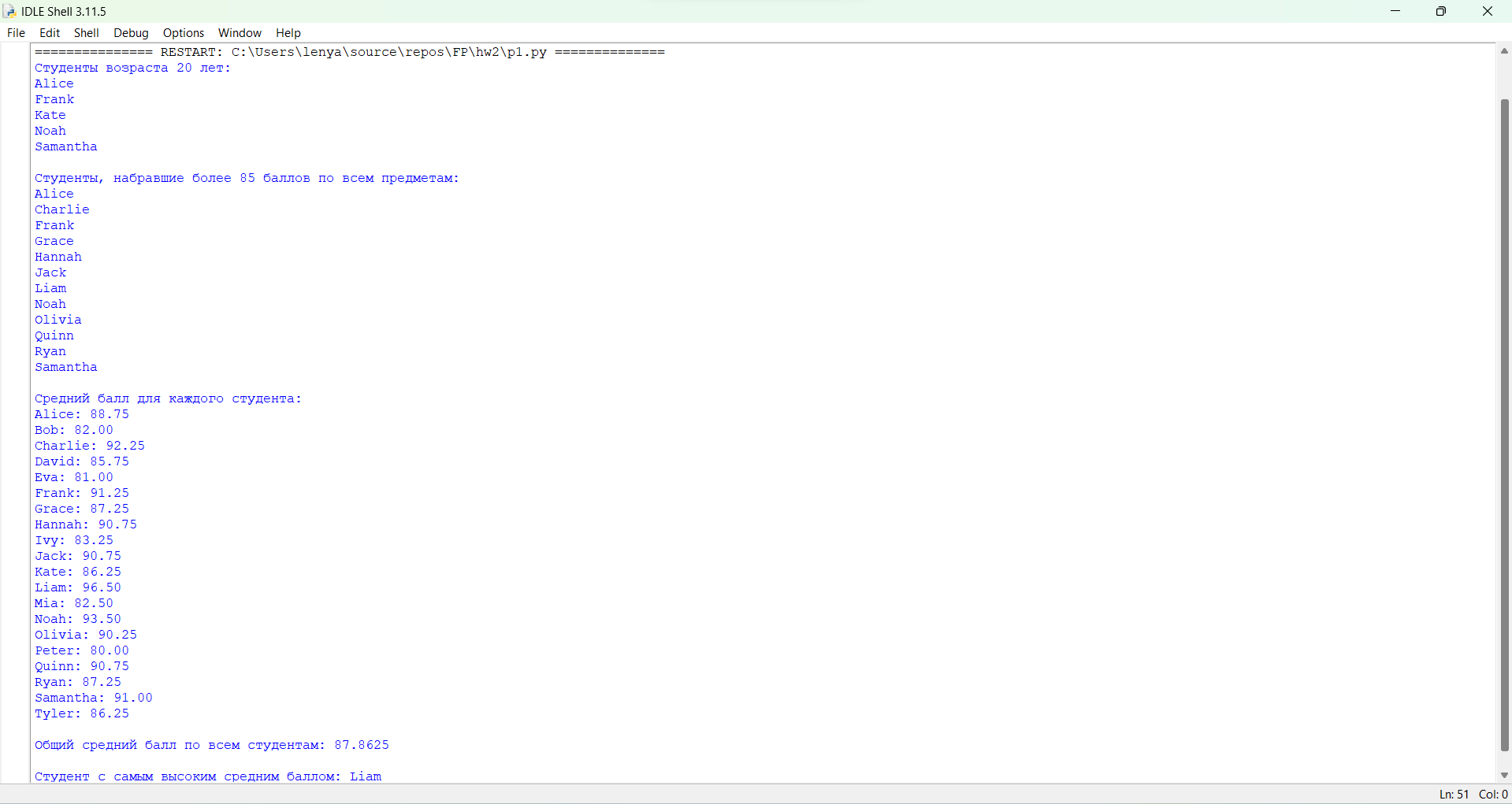
**5. Рекомендации пользователю**

* Для запуска программ рекомендуется использовать Python 3.x.
* При запуске программ убедитесь, что данные, предоставленные в виде списков словарей, правильно представлены, и их структура соответствует ожидаемой программой.
* В зависимости от задачи, убедитесь, что заданные критерии фильтрации и преобразования данных корректны.

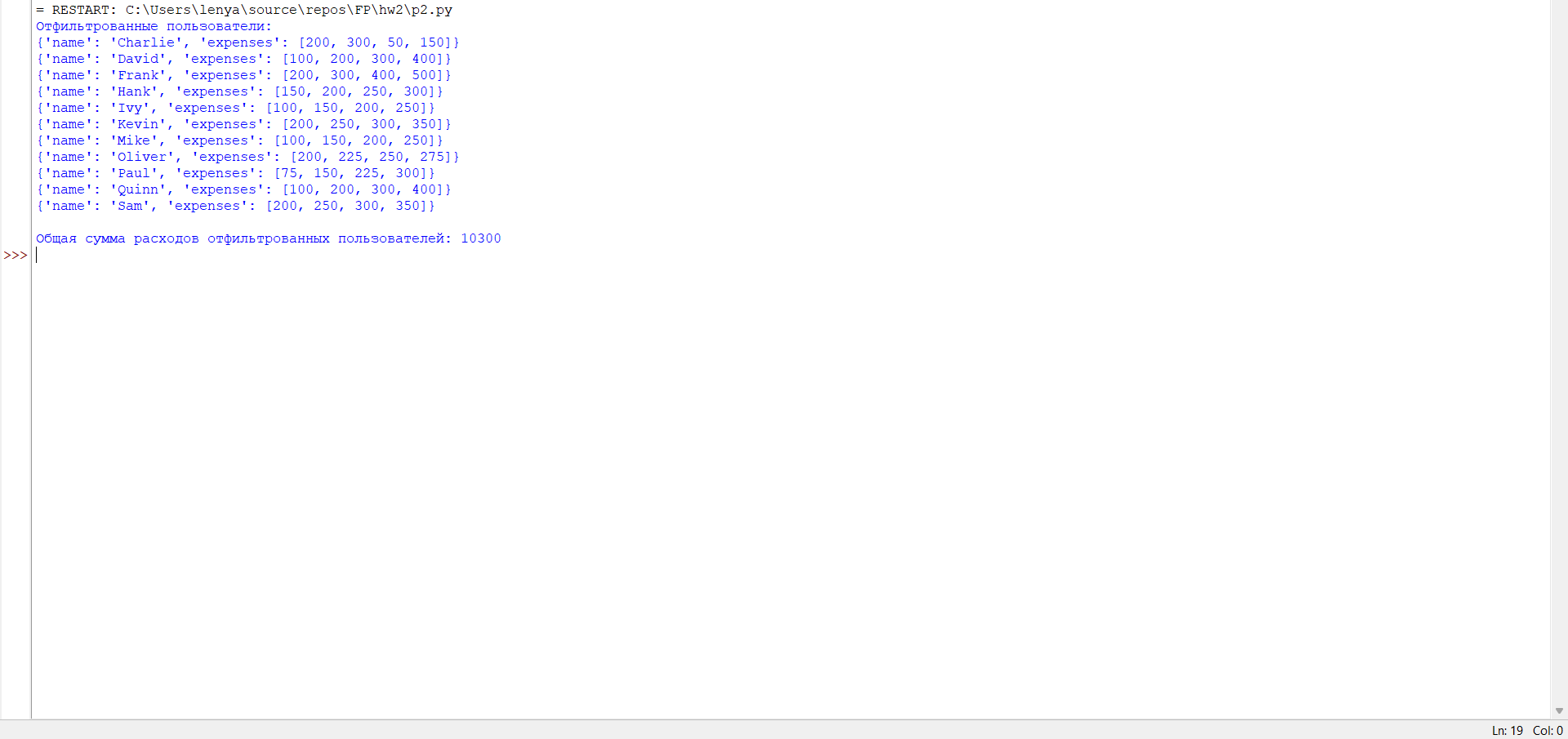
**6. Рекомендации программисту**

* Обеспечьте хорошую структуру кода, чтобы он был легко читаем и поддерживаем.
* Используйте функции *map* и *filter* с умом, чтобы оптимизировать обработку данных.
* Регулярно проверяйте исключения и обрабатывайте их, чтобы избежать ошибок во время выполнения программы.

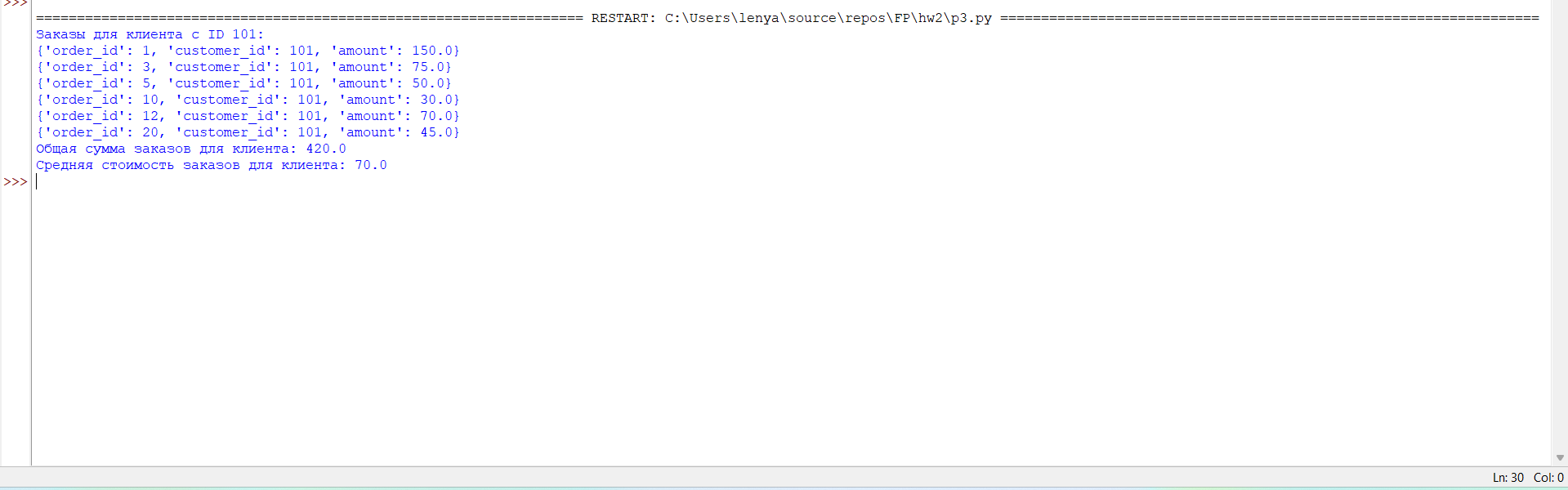
**7. Контрольный пример**



*Рис. 1 (ЗАДАЧА 1. При запуске программы в консоль выводятся требуемые данные о всех студентах, указанных в массиве)*

**

*Рис. 2 (ЗАДАЧА 2. При запуске программы в консоль выводятся требуемые данные о всех пользователях, указанных в массиве)*

**

*Рис. 3 (ЗАДАЧА 3. При запуске программы в консоль выводятся требуемые данные о выбранном клиенте из данного массива)*

**8. Вывод**

Разработанные программы позволяют решать разнообразные задачи, связанные с фильтрацией, агрегацией и анализом данных. Использование функций *map* и *filter* помогает упростить и ускорить обработку информации, делая программы более эффективными. Однако, важно убедиться в корректности данных и правильности заданных критериев для получения точных результатов.

Ссылка на GitHub: https://github.com/p0le0n/fp\_2