Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Ульяновский государственный технический университет»

Кафедра «Вычислительная техника»

**Отчет по лабораторной работе №3**

Дисциплина: «Разработка профессиональных приложений»

Файлы и словари

Вариант № 5

Выполнил:

студент группы ИВТАПбд-21

Гаязов Т. Р.

Проверил:

преподаватель кафедры

«Вычислительная техника»

Исхаков И.И.

Ульяновск, 2023

**Задание по варианту**

Необходимо реализовать 5 вариант задания, а именно: из исходного .csv файла с начальными данными Временная шкала температурных показаний: №, дата и время, сумма, наименование 1 товарной позиции.

**Требования к программе:**

1.    Пусть дана некоторая директория (папка). Посчитайте количество файлов в данной директории (папке) и выведите на экран.

2.    Пусть   дан   файл   data.csv,  в   котором   содержится информация в соответствии с вариантом:

Считайте информацию из файла в соответствующую структуру (словарь):

2.1. Выведите информацию об объектах, отсортировав их по одному полю (строковому).

2.2. Выведите информацию об объектах, отсортировав их по одному полю (числовому).

2.3. Выведите информацию, соответствующую какому-либо критерию (например, для студентов - тех, у кого возраст больше какого-либо значения)

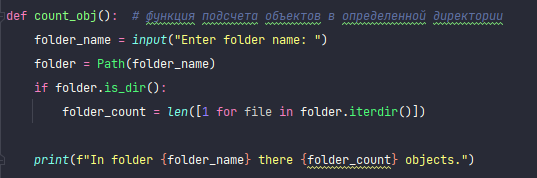
3.    Добавьте к программе возможность сохранения новых данных обратно в файл.

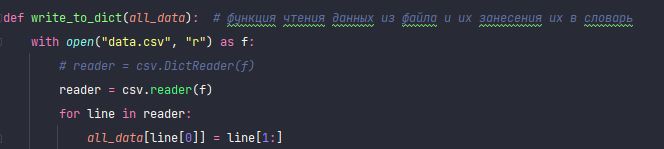
**Описание реализации**

Созданы 6 функций:

1. Функция вывода кол-ва файлов в заданной директории

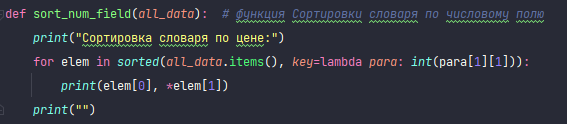
В которой вы считаем кол-во файлов в заданной папке при помощи класса Path, библиотеки pathlib.



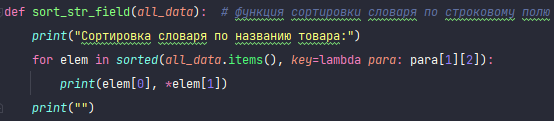
2. Функция чтения данных из файла и занесение их в словарь 

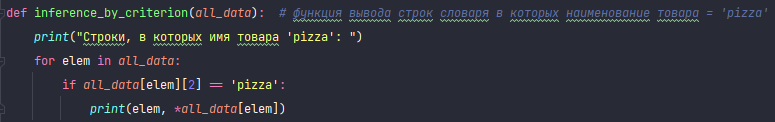
Происходит открытие файла формата .csv и считывание, дальше с помощью цикла for пробегаемся по данным и записываем их в словарь

3. Функция сортировки по целочисленному значению:



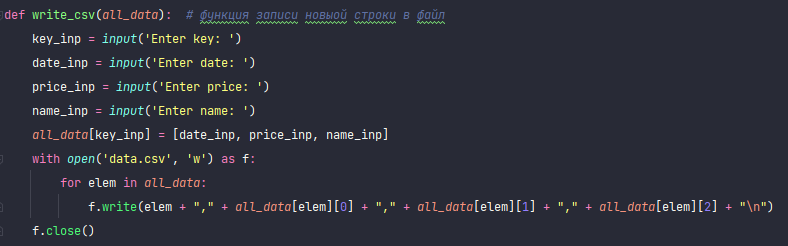
С помощью цикла for пробегаемся по элементам словаря. Сортировка происходит благодаря методу sorted() в параметрах которого указаны элементы словаря и функция lambda.

4. Сортировка по строковому полю  


5. Сортировка по критерию 

С помощью цикла for пробегаемся по словарю с данными и выводим строки содержащие нужное нам название товара.

6. Запись в файл



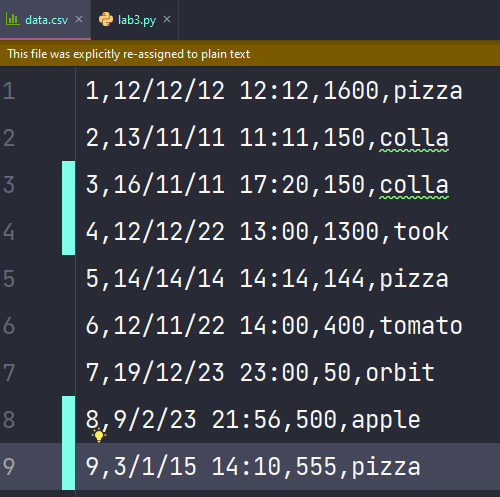
В начале функции происходит считывание данных с клавиатуры, а именно: ключ, дата, цена, название товара. Данные записываются в словарь. После этого открываем файл на чтение и благодаря циклу for записываем в него словарь, содержавший предыдущие и новые данные. После этого закрываем файл.

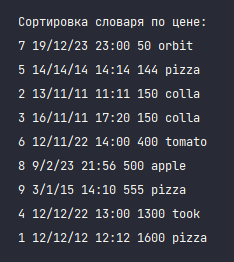
**Описание возникших затруднений**

Затруднения возникли с сортировкой словаря по строковому и числовому полям так как в словаре в качестве значения ключа был список. Немного изучив данную ситуацию, проблема была решена. Других затруднений лабораторная работа не вызвала

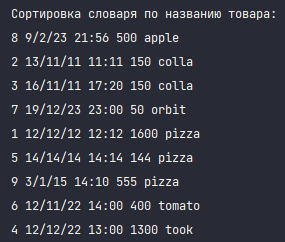
**Результаты работы программы**

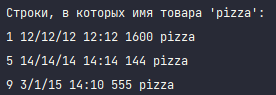
**  
1 рис. Кол-во файлов в папке Загрузки**

**  
2 рис. Исходный .csv файл**

****

**3 рис. Сортировка по цене**

**  
4 рис. Сортировка по названию товара**

**  
5 рис. Сортировка по критерию (название товара = ‘pizza’)**

**Код программы**

from pathlib import Path  
import csv  
  
  
def count\_obj(): *# функция подсчета объектов в определенной директории* folder\_name = *input*("Enter folder name: ")  
 folder = Path(folder\_name)  
 if folder.is\_dir():  
 folder\_count = *len*([1 for file in folder.iterdir()])  
  
 *print*(f"In folder {folder\_name} there {folder\_count} objects.")  
  
  
def write\_csv(*all\_data*): *# функция записи новыой строки в файл* key\_inp = *input*('Enter key: ')  
 date\_inp = *input*('Enter date: ')  
 price\_inp = *input*('Enter price: ')  
 name\_inp = *input*('Enter name: ')  
 *all\_data*[key\_inp] = [date\_inp, price\_inp, name\_inp]  
 with *open*('data.csv', 'w') as f:  
 for elem in *all\_data*:  
 f.write(elem + "," + *all\_data*[elem][0] + "," + *all\_data*[elem][1] + "," + *all\_data*[elem][2] + "\n")  
 f.close()  
  
  
def write\_to\_dict(*all\_data*): *# функция чтения данных из файла и их занесения их в словарь* with *open*("data.csv", "r") as f:  
 *# reader = csv.DictReader(f)* reader = csv.reader(f)  
 for line in reader:  
 *all\_data*[line[0]] = line[1:]  
  
  
def sort\_num\_field(*all\_data*): *# функция Сортировки словаря по числовому полю  
 print*("Сортировка словаря по цене:")  
 for elem in *sorted*(*all\_data*.items(), *key*=lambda *para*: *int*(para[1][1])):  
 *print*(elem[0], \*elem[1])  
 *print*("")  
  
  
def sort\_str\_field(*all\_data*): *# функция сортировки словаря по строковому полю  
 print*("Сортировка словаря по названию товара:")  
 for elem in *sorted*(*all\_data*.items(), *key*=lambda *para*: para[1][2]):  
 *print*(elem[0], \*elem[1])  
 *print*("")  
  
  
def inference\_by\_criterion(*all\_data*): *# функция вывода строк словаря в которых наименование товара = 'pizza'  
 print*("Строки, в которых имя товара 'pizza': ")  
 for elem in *all\_data*:  
 if *all\_data*[elem][2] == 'pizza':  
 *print*(elem, \**all\_data*[elem])  
  
  
def main():  
 all\_data = {}  
 count\_obj()  
 write\_to\_dict(all\_data)  
  
 *print*("ID, Date, Price, Product Name")  
 for elem in all\_data:  
 *print*(elem, \*all\_data[elem])  
 *print*("\n")  
  
 sort\_num\_field(all\_data)  
 sort\_str\_field(all\_data)  
 inference\_by\_criterion(all\_data)  
 *print*("")  
  
 check = *input*('Add newline to file? Enter y/n: ')  
 if check == 'y':  
 write\_csv(all\_data)  
 *print*("Данные успешно сохранены!")  
 else:  
 *print*("Программа остановлена")  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()