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 №7

Выполнила:

студентка группы ИВТАПбд-22

Ренжина А.С.

Проверил:

преподаватель кафедры

«Вычислительная техника»

Исхаков И.И.

Ульяновск, 2023

**Задание по варианту:** История посещений банка: №, ФИО, дата и время, тип обращения.

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:**

Первым делом пишем условие, если файл называется main, тогда он запускает функцию с идентичным названием.

В функции main, мы вернемся позже.

Для работы программы и выполнения задания, мы напишем нужные нам функции, которые будут использоваться далее.

Первая функция,это функция которая находит нужную папку, если не находит, то просит ввести еще раз.

Далее, функция которая считывает количество файлов в папке, она же используется в функции, которая будет выводить количество папок и их названия.

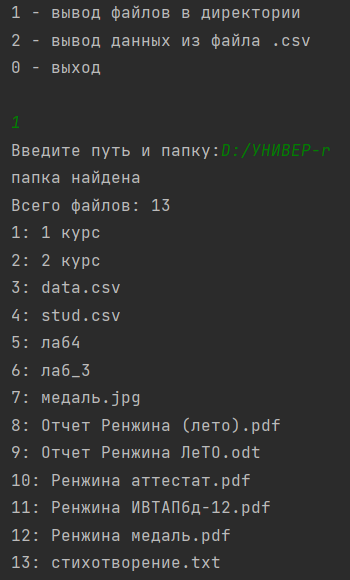
Функции отвечающие за файлы csv: первая функция, ищет нужный нам файл, а также если он не найден, то просит ввести еще раз название.

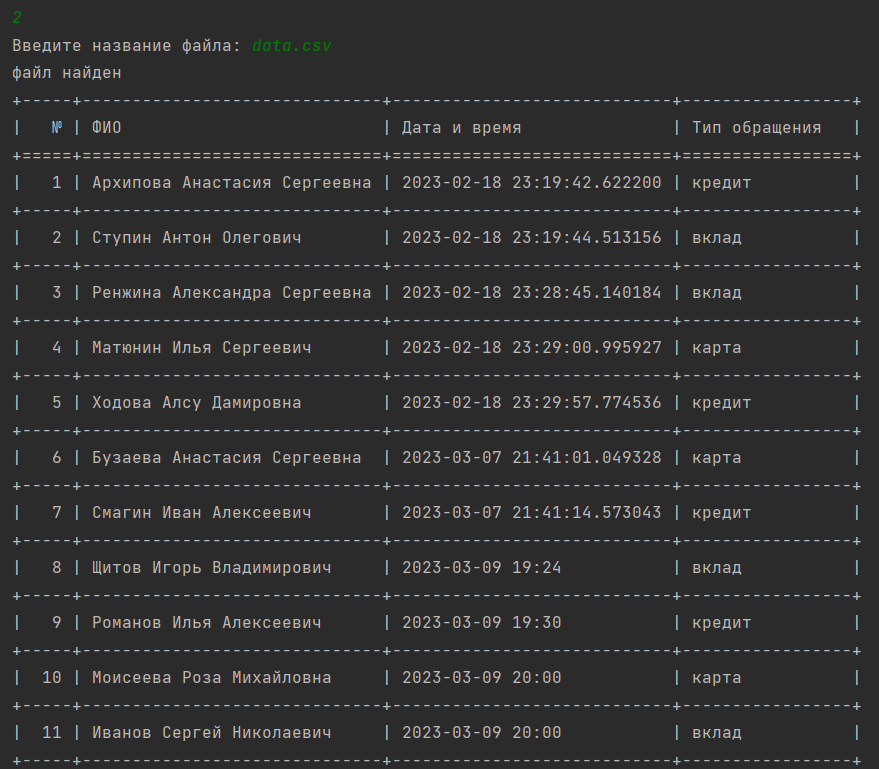
Функция, отвечающая за чтение данных из файла, с помощью словарей, она считывает нужные данные. Ну а также функция вывода данных.

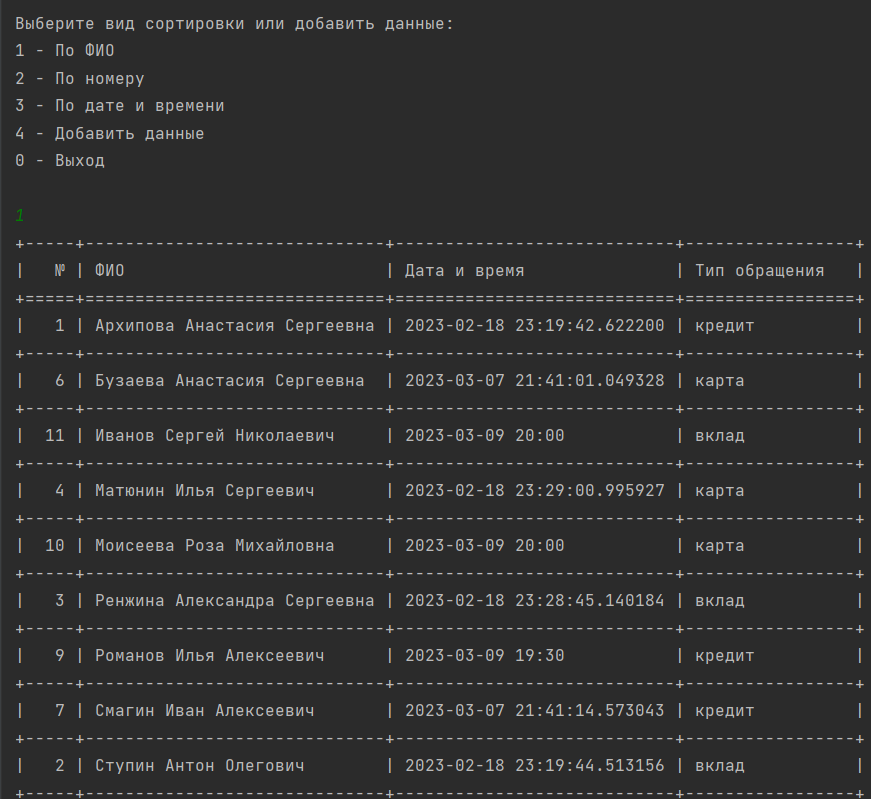
По заданию мы можем узнать, что нам нужно сделать сортировку, а также добавление, создаем также эти две функции.

В самом main(), мы прописываем все эти функции в том порядке, в котором нам это надо, при этом написано все в виде мини консольного приложения, при вводе определенной команды, вызываются определенные функции.

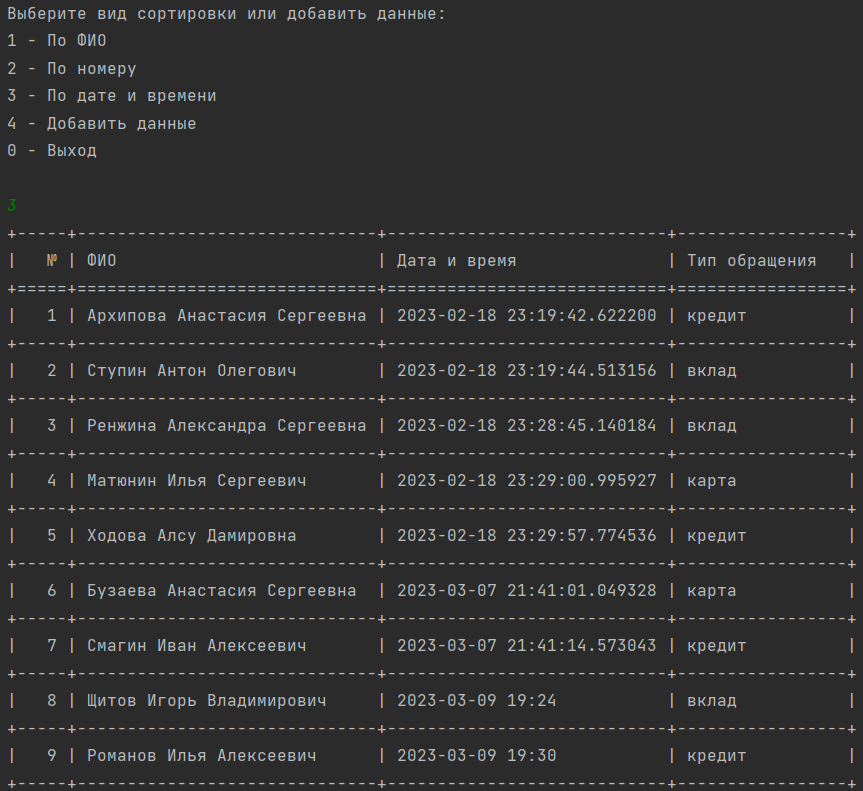
**Тестовые примеры:**

1. Количество файлов в папке:
2. Вывод данных:

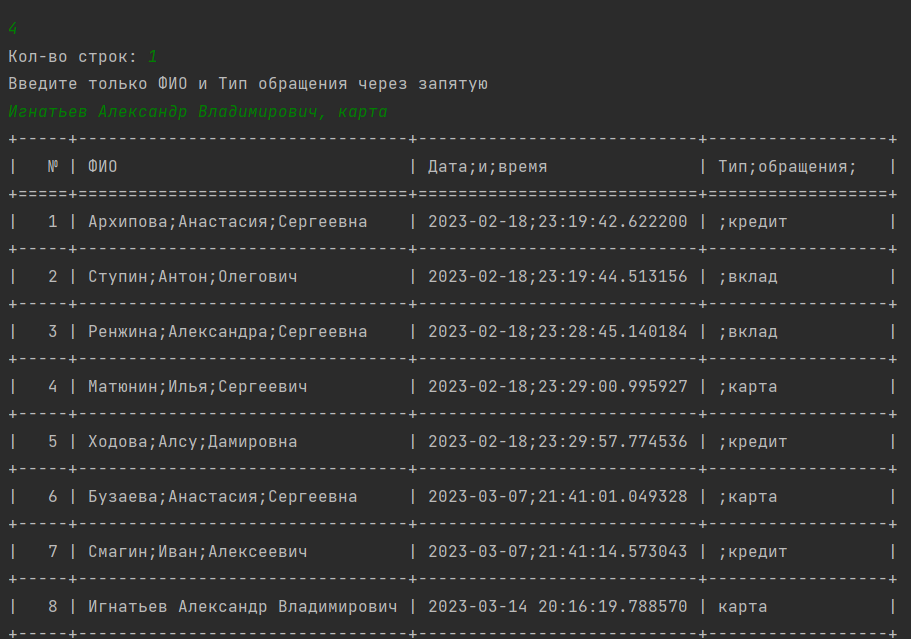


2.1 Сортировка по строковому значению:

2.2 По числовому по номеру сортирует

2.3 По какому либо критерию:

1. Добавление данных:



Исходный код:

import operator

import os

import csv

import datetime

from tabulate import tabulate

import pandas as pd

def add\_new\_data(data, keys):

n = int(input("Кол-во строк: "))

print("Введите только ФИО и Тип обращения через запятую")

for i in range(len(data), len(data) + n):

row = input().split(",")

new\_row = {

keys[0]: str(len(data) + 1),

keys[1]: row[0],

keys[2]: str(datetime.datetime.now()),

keys[3]: row[1],

}

data.append(new\_row)

def write\_file(data, keys, file):

data.sort(key=operator.itemgetter(keys[0]))

with open(file, "w", newline='') as file:

writer = csv.DictWriter(file, fieldnames=keys)

writer.writeheader()

writer.writerows(data)

file.close()

def action\_data(sort\_i, data, keys, file):

if sort\_i == 1:

return data.sort(key=operator.itemgetter(keys[1]))

if sort\_i == 2:

return data.sort(key=operator.itemgetter(keys[0]))

if sort\_i == 3:

return data.sort(key=operator.itemgetter(keys[2]))

if sort\_i == 4:

add\_new\_data(data, keys)

write\_file(data, keys, file)

if sort\_i == 0:

return data

def print\_data(data, keys):

flag = True

while flag:

if len(data) != 0:

print(tabulate(pd.DataFrame(data), headers=keys, tablefmt='grid', showindex=False))

flag = False

else:

print("Пустой файл")

def get\_data(file):

data, keys = [], []

with open(file, "r") as file:

reader = csv.DictReader(file)

keys = reader.fieldnames

for row in reader:

data.append(row)

file.close()

return data, keys

def file\_set(path):

file\_name = input("Введите название файла: ")

file = f"{path}/{file\_name}"

if os.path.exists(file):

print("файл найден")

return file

else:

print("Ошибка: такой файл не существует.\n"

"1 - попробовать ещё раз\n"

"0 - выход")

op = int(input("введите команду: "))

if op == 1:

file = file\_set(path)

return file

if op == 0:

return

def read\_file():

with open('data.csv') as File:

reader = csv.reader(File, delimiter=',', quotechar=',',

quoting=csv.QUOTE\_MINIMAL)

for row in reader:

print(row)

def print\_information(arr):

print(f"Всего файлов: {len(arr)}")

k = 1

for i in arr:

print(f"{k}: {i}")

k += 1

def director():

path = input("Введите путь и папку:")

if os.path.exists(path):

print("папка найдена")

return path

else:

print("Ошибка: такой папки не существует.\n"

"1 - попробовать ещё раз\n"

"0 - выход")

op = int(input("введите команду: "))

if op == 1:

director()

if op == 0:

return

def print\_file(path):

flag = True

while flag:

if os.path.exists(path):

print\_information(os.listdir(path))

return

else:

print("папки по этому пути не существует.\n"

"1 - попробовать ещё раз \n"

"0 -выход \n")

op = int(input("введите команду: "))

if op == 0:

flag = False

if op == 1:

path = print\_file(path)

return path

if op > 1:

print("введите 1 или 0")

def main():

path = ''

file = ''

data = []

keys = []

while True:

print(f"1 - вывод файлов в директории\n"

f"2 - вывод данных из файла .csv\n"

f"0 - выход\n")

itype = int(input())

if itype == 1:

path = director()

print\_file(path)

if itype == 2:

now = True

called1 = False

called2 = False

while now:

if path == '' or path is None:

if not called1:

path = director()

called1 = True

else:

now = False

else:

if file == '' or file is None:

if not called2:

file = file\_set(path)

called2 = True

else:

now = False

else:

if len(data) == 0:

data, keys = get\_data(file)

else:

print\_data(data, keys)

print(f"Выберите вид сортировки или добавить данные:\n"

f"1 - По ФИО\n"

f"2 - По номеру\n"

f"3 - По дате и времени\n"

f"4 - Добавить данные\n"

f"0 - Выход\n")

sort\_i = int(input())

action\_data(sort\_i, data, keys, file)

print\_data(data, keys)

now = False

if itype == 0:

exit(1)

if \_\_name\_\_ == '\_\_main\_\_':

main()

**Описание возникших затруднений:**

При выполнении лабораторной работы, затруднений не возникло.

**Описание альтернативных способ решения:**

Альтернативных способ решения при выполнении лабораторной работы не возникло.

**Вывод о проделанной работе:**

Во время выполнения лабораторной работы, познакомились с файлами и словарями, библиотеками, которые работают с файлами и папками.