Инвариантная самостоятельная работа №3

**Тема:** Строки. Байты. Файлы

**Оборудование:** VS code, python3

**Задача 3.2**

**Постановка задачи:** Разработка сценария с реализацией операции поиска подстроки в тексте.

**Код программы:**

"""

Царулкова Анастасия Витальевна

2 группа 3 подгруппа

Copyright: 10.2019

Задание 3.2 ИСР

Cкрипт с функцией, которая ищет подстроку в строке.

"""

def search\_str(what="", where=""):

# 1 - поиск первого вхождения подстроки

flag = False

i = 0

while i < len(where):

if i+len(what)-1 < len(where) and where[i:i+len(what)] == what:

flag = True

index = i

i = len(where)-1

i+=1

if flag == True:

print("Начало ",index, " - Нашел")

else:

print("Не нашел")

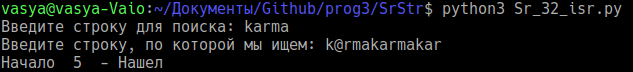
def main():

input\_str = input("Введите строку для поиска: ")

searchable\_str = input("Введите строку, по которой мы ищем: ")

search\_str(input\_str, searchable\_str)

main()

**Результат работы программы:**

**Задача 3.3**

**Постановка задачи:** Создание скрипта для считывания данных справочных логов из текстового файла и преобразования их в CSV-формат с последующей записью в новый файл. Формирование отчета по выполнению задания и размещение его в портфолио, персональном репозитории.

**Код программы:**

import csv

import json

with open('MOCKDATA.json') as f:

data\_dict = json.load(f)

with open('eggs.csv', 'w', newline='') as csvfile:

jsonwriter = csv.writer(

csvfile, delimiter=',', quotechar='"', quoting = csv.QUOTE\_MINIMAL)

jsonwriter.writerow(data\_dict[0].keys())

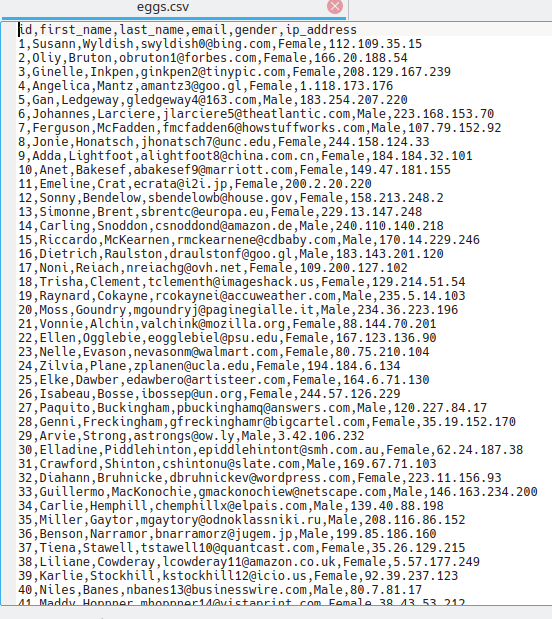
keys = data\_dict[0].keys()

for el in data\_dict:

jsonwriter.writerow(el.values())

print("Файл записан")

**Результат работы программы:**

****

**Задача 3.4**

**Постановка задачи:** Реализовать программу шифрующую строку, задаваемую пользователем, с помощью алгоритма шифрования ROT13. Формирование отчета по выполнению задания и размещение его в портфолио, персональном репозитории.

**Код программы:**

# ROT13

import string

alphabet\_low = "абвгдеёжзийклмнопрстуфхцчшщъыьэюя"

alphabet\_up = "АБВГДЕЁЖЗИЙКДМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"

def rot13(s):

new\_s = ""

for char in s:

if char in string.ascii\_lowercase:

if string.ascii\_lowercase.find(char)+13 < len(string.ascii\_lowercase):

new\_s += string.ascii\_lowercase[string.ascii\_lowercase.find(char)+13]

else:

add = abs(len(string.ascii\_lowercase)-string.ascii\_lowercase.find(char)-13)

new\_s += string.ascii\_lowercase[add]

elif char in string.ascii\_uppercase:

if string.ascii\_uppercase.find(char)+13 < len(string.ascii\_uppercase):

new\_s += string.ascii\_uppercase[string.ascii\_uppercase.find(char)+13]

else:

add = abs(len(string.ascii\_uppercase)-string.ascii\_uppercase.find(char)-13)

new\_s += string.ascii\_uppercase[add]

elif char in alphabet\_low:

if alphabet\_low.find(char)+13 < len(alphabet\_low):

new\_s += alphabet\_low[alphabet\_low.find(char)+13]

else:

add = abs(len(alphabet\_low)-alphabet\_low.find(char)-13)

new\_s += alphabet\_low[add]

elif char in alphabet\_up:

if alphabet\_up.find(char)+13 < len(alphabet\_up):

new\_s += alphabet\_up[alphabet\_up.find(char)+13]

else:

add = abs(len(alphabet\_up)-alphabet\_up.find(char)-13)

new\_s += alphabet\_up[add]

else:

new\_s+=char

print(new\_s)

def main():

s = str(input("Введите строку для шифрования:"))

rot13(s)

main()

**Результат работы программы**

