

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций**

**Отчет по лабораторной работе №8  
Работа со словарями в языке Python  
по дисциплине «Технологии программирования»**

Выполнил студент группы ИВТ-б-о-20-1

Ищенко Т.С. « » \_\_\_\_\_ 20\_\_ г.

Подпись студента \_\_\_\_\_

Работа защищена « » \_\_\_\_\_ 20\_\_ г.

Проверила Воронкин Р.А. \_\_\_\_\_

(подпись)

Ставрополь 202

Цель работы: приобретение навыков по работе со словарями при написании программ с помощью языка программирования Python версии 3.x.

Методика и порядок выполнения работы

1. Изучить теоретический материал работы.
2. Создать общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT и язык программирования Python.
3. Выполните клонирование созданного репозитория.
4. Дополните файл .gitignore необходимыми правилами для работы с IDE PyCharm.
5. Организуйте свой репозиторий в соответствие с моделью ветвления git-flow.
6. Создайте проект PyCharm в папке репозитория.
7. Проработайте пример лабораторной работы. Создайте для него отдельный модуль языка Python. Зафиксируйте изменения в репозитории.
8. Приведите в отчете скриншоты результатов выполнения примера при различных исходных данных вводимых с клавиатуры.
9. Решите задачу: создайте словарь, связав его с переменной school , и наполните данными, которые бы отражали количество учащихся в разных классах (1а, 1б, 2б, 6а, 7в и т. п.). Внесите изменения в словарь согласно следующему: а) в одном из классов изменилось количество учащихся, б) в школе появился новый класс, с) в школе был расформирован (удален) другой класс. Вычислите общее количество учащихся в школе.
10. Зафиксируйте сделанные изменения в репозитории.
11. Решите задачу: создайте словарь, где ключами являются числа, а значениями – строки. Примените к нему метод items(), с помощью полученного объекта dict\_items создайте новый словарь, "обратный" исходному, т. е. ключами являются строки, а значениями – числа.

12. Зафиксируйте сделанные изменения в репозитории.
13. Приведите в отчете скриншоты работы программ и UML-диаграммы деятельности решения индивидуального задания.
14. Зафиксируйте сделанные изменения в репозитории.
15. Выполните слияние ветки для разработки с веткой master.
16. Отправьте сделанные изменения на сервер GitHub.

Ход работы:

1. Создал новый репозиторий

## Create a repository

C:\my\work\skfy\3sim\tef\_program\8lab Обзор

8lab

Git

☒ Создать репозиторий на аккаунте:

Учётная запись

GitHub: Timofej8971

Владелец

Timofej8971

Lab-8

☐ Is Private

Создать

Рисунок 1.1 – Создание нового репозитория

2. Произвёл выполнение примера

```

#!/usr/bin/env python3
# -*- coding: utf-8 -*-
from datetime import date
import sys

if __name__ == '__main__':
    # Список работников.
    workers = []

    # Организовать бесконечный цикл запроса команд.
    while True:
        # Запросить команду из терминала.
        command = input(">>> ").lower()

        # Выполнить действие в соответствие с командой.
        if command == 'exit':
            break

        elif command == 'add':
            # Запросить данные о работнике.
            name = input("Фамилия и инициалы? ")
            post = input("Должность? ")
            year = int(input("Год поступления? "))

            # Создать словарь.
            worker = {
                'name': name,
                'post': post,
                'year': year,
            }

            # Добавить словарь в список.
            workers.append(worker)
            # Осортировать список в случае необходимости.
            if len(workers) > 1:
                workers.sort(key=lambda item: item.get('name', ''))

        elif command == 'list':
            # Заголовки таблиц.
            line = '-+-(+)-+-(+)-+-(+)-+-(+)-+-.format(
                '1',
                '2',
                '3',
                '4'
            )
            print(line)
            print!

```

```

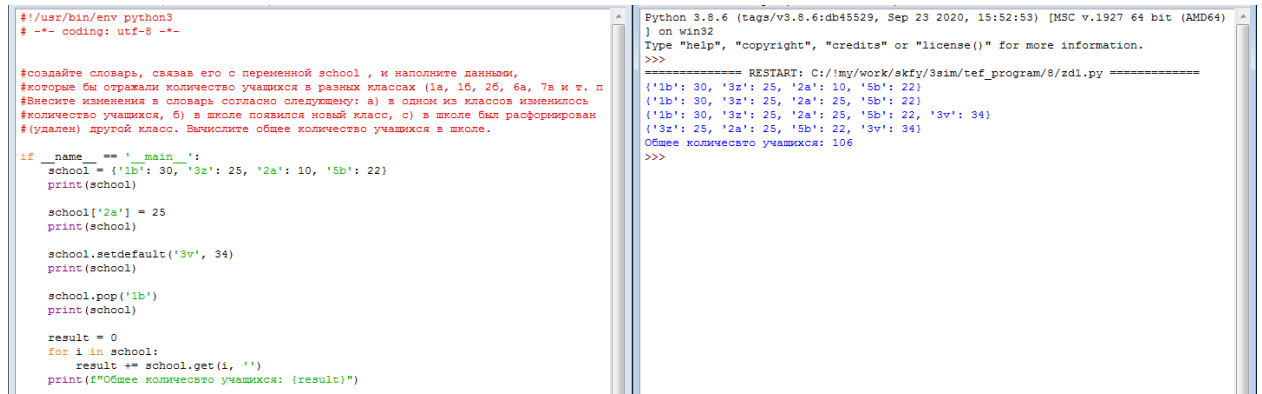
python 3.8.6 (tags/v3.8.6:0b30354, Sep 23 2020, 13:08:13) [AMD64] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/my/work/skfy/3sim/tef_program/8/primer.py =====
>>> help
Список команд:

add - добавить работника;
list - вывести список работников;
select <отж> - запросить работников со отжам;
help - отобразить справку;
exit - завершить работу с программой.
>>> add
Фамилия и инициалы? Попов Д.А.
Должность? Большой молодец
Год поступления? 2015
>>> И
Неизвестная команда и
>>> add
Фамилия и инициалы? Ищенко Т. С.
Должность? молодец
Год поступления? 2016
>>> list
+-----+-----+-----+-----+
| № |      Ф.И.О.      | Должность | Год |
+-----+-----+-----+-----+
| 1 | Ищенко Т. С.    | молодец   | 2016 |
| 2 | Попов Д.А.     | Большой молодец | 2015 |
+-----+-----+-----+-----+
>>>

```

Рисунок 1.2 – Результат выполнения примера

### 3. Выполнил второе задание



```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

#создайте словарь, связав его с переменной school, и наполните данными,
#которые бы отражали количество учащихся в разных классах (1а, 1б, 2б, 6а, 7а и т. п.
#Внесите изменения в словарь согласно следующему: а) в одном из классов изменилось
#количество учащихся, б) в школе появился новый класс, с) в школе был расформирован
#(удален) другой класс. Вычислите общее количество учащихся в школе.

if __name__ == '__main__':
    school = {'1б': 30, '3з': 25, '2а': 10, '5б': 22}
    print(school)

    school['2а'] = 25
    print(school)

    school.setdefault('3в', 34)
    print(school)

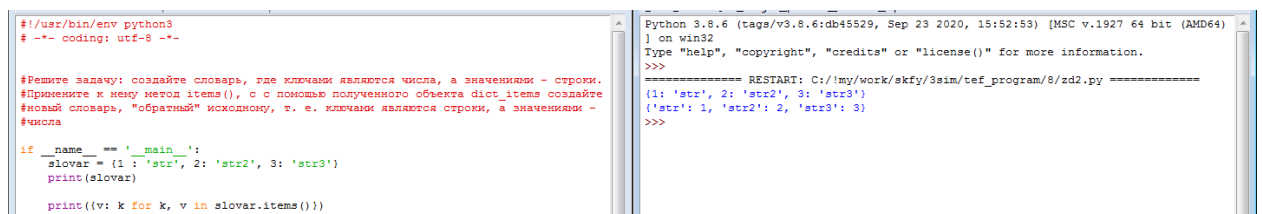
    school.pop('1б')
    print(school)

    result = 0
    for i in school:
        result += school.get(i, '')
    print(f"Общее количество учащихся: {result}")
```

```
Python 3.8.6 (tags/v3.8.6:db45529, Sep 23 2020, 15:52:53) [MSC v.1927 64 bit (AMD64)]
> on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\my\work\skfy\3aim\tef_program\8\zd1.py =====
{'1б': 30, '3з': 25, '2а': 10, '5б': 22}
{'1б': 30, '3з': 25, '2а': 25, '5б': 22}
{'1б': 30, '3з': 25, '2а': 25, '5б': 22, '3в': 34}
{'3з': 25, '2а': 25, '5б': 22, '3в': 34}
Общее количество учащихся: 106
>>>
```

Рисунок 1.3 – Результат выполнения второго задания

### 4. Выполнил третье задание



```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

#Решите задачу: создайте словарь, где ключами являются числа, а значениями – строки.
#Примените к нему метод items(), с с помощью полученного объекта dict_items создайте
#новый словарь, "обратный" исходному, т. е. ключами являются строки, а значениями –
#числа

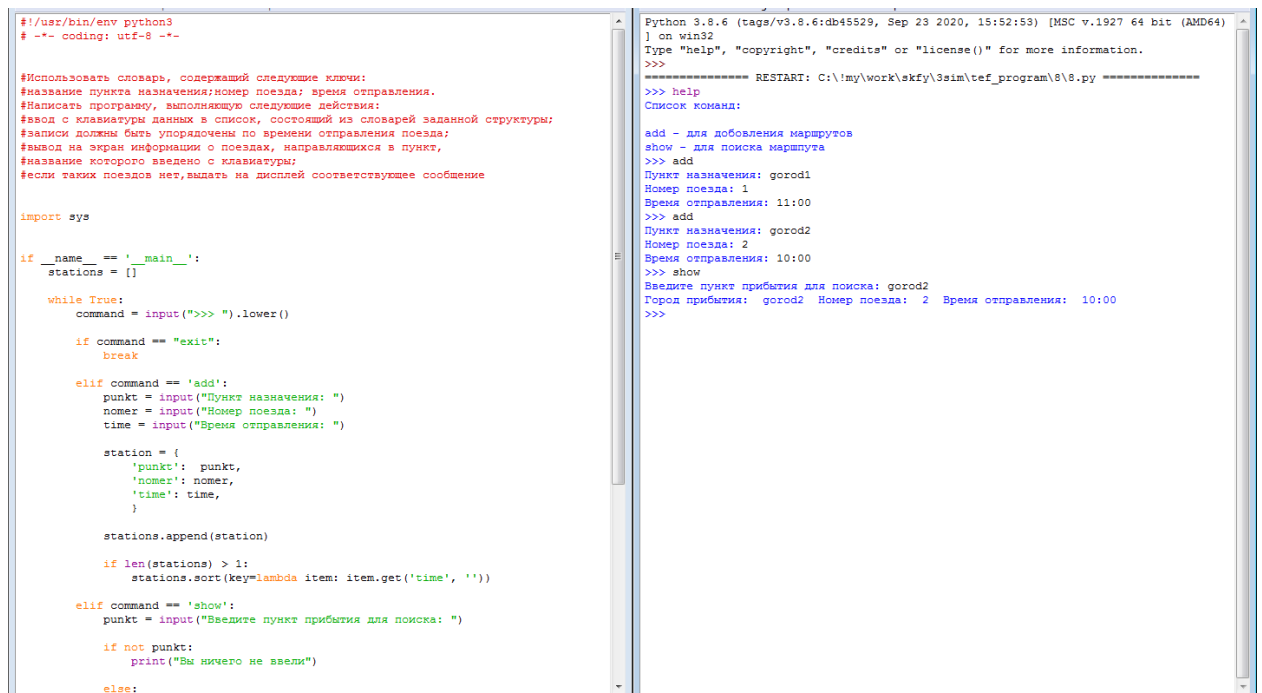
if __name__ == '__main__':
    slovar = {1: 'str1', 2: 'str2', 3: 'str3'}
    print(slovar)

    print({v: k for k, v in slovar.items()})
```

```
Python 3.8.6 (tags/v3.8.6:db45529, Sep 23 2020, 15:52:53) [MSC v.1927 64 bit (AMD64)]
> on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\my\work\skfy\3aim\tef_program\8\zd2.py =====
{1: 'str1', 2: 'str2', 3: 'str3'}
{'1': 1, 'str2': 2, 'str3': 3}
>>>
```

Рисунок 1.4 – Результат выполнения третьего задания

### 5. Выполнил индивидуальное задание



```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

#Использовать словарь, содержащий следующие ключи:
#название пункта назначения;номер поезда; время отправления.
#Написать программу, выполняющую следующие действия:
#ввод с клавиатуры данных в список, состоящий из словарей заданной структуры;
#записи должны быть упорядочены по времени отправления поезда;
#вывод на экран информации о поездах, направляющихся в пункт,
#название которого введено с клавиатуры;
#если таких поездов нет, выдать на дисплей соответствующее сообщение

import sys

if __name__ == '__main__':
    stations = []

    while True:
        command = input(">>> ").lower()

        if command == "exit":
            break

        elif command == "add":
            punkt = input("Пункт назначения: ")
            номер = input("Номер поезда: ")
            time = input("Время отправления: ")

            station = {
                'punkt': punkt,
                'номер': номер,
                'time': time,
            }

            stations.append(station)

            if len(stations) > 1:
                stations.sort(key=lambda item: item.get('time', ''))

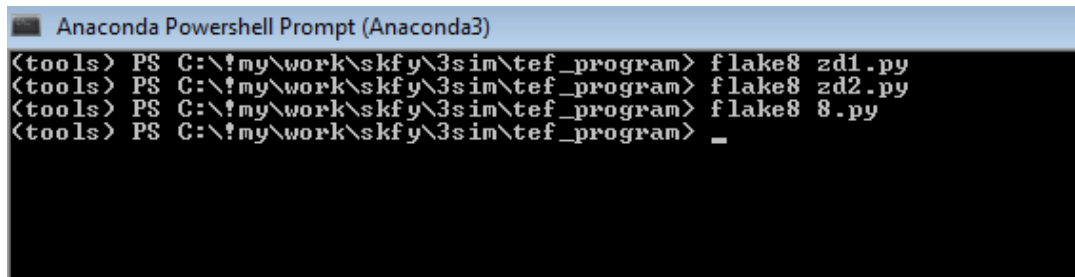
        elif command == "show":
            punkt = input("Введите пункт прибытия для поиска: ")

            if not punkt:
                print("Вы ничего не ввели")
            else:
                pass
```

```
Python 3.8.6 (tags/v3.8.6:db45529, Sep 23 2020, 15:52:53) [MSC v.1927 64 bit (AMD64)]
> on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\my\work\skfy\3aim\tef_program\8\8.py =====
>>> help
Список команд:
add – для добавления маршрутов
show – для поиска маршрута
>>> add
Пункт назначения: gorod1
Номер поезда: 1
Время отправления: 11:00
>>> add
Пункт назначения: gorod2
Номер поезда: 2
Время отправления: 10:00
>>> show
Введите пункт прибытия для поиска: gorod2
Город прибытия: gorod2 Номер поезда: 2 Время отправления: 10:00
>>>
```

Рисунок 1.5 – Результат выполнения индивидуального задания

### 6. Провел проверку на PEP8 всех заданий



```
Anaconda Powershell Prompt (Anaconda3)
(tools) PS C:\!my\work\skfy\3sim\tef_program> flake8 zd1.py
(tools) PS C:\!my\work\skfy\3sim\tef_program> flake8 zd2.py
(tools) PS C:\!my\work\skfy\3sim\tef_program> flake8 8.py
(tools) PS C:\!my\work\skfy\3sim\tef_program> _
```

Рисунок 1.6 – Результат проверки на соответствие PEP8

## 7. Произвёл синхронизацию с удаленным репозиторием

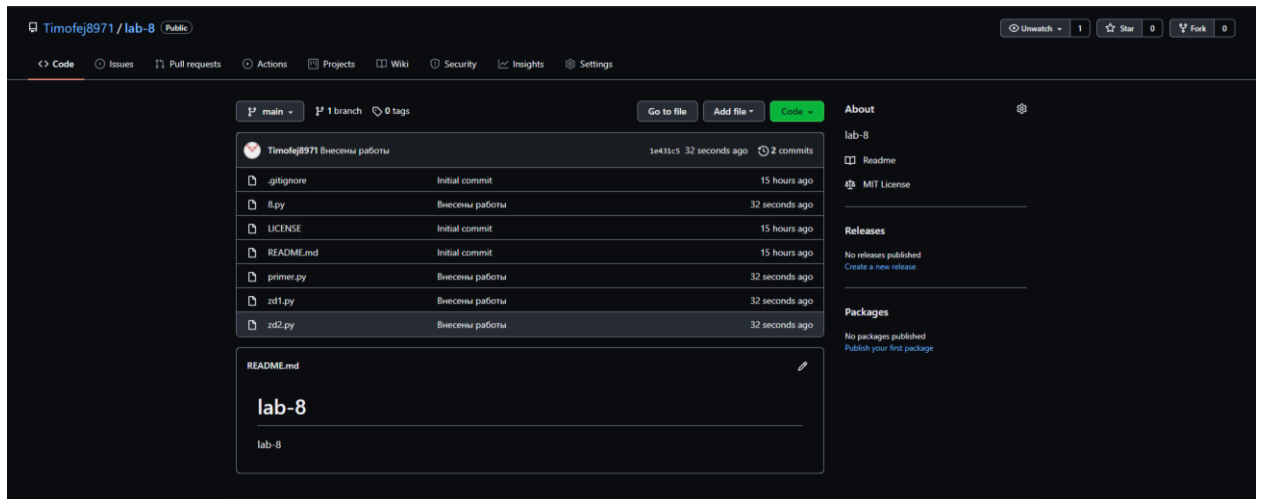


Рисунок 1.7 – Результат синхронизации с удалённым репозиторием

Контрольные вопросы:

1. Что такое словари в языке Python? – словарь представляет собой структуру данных (которая ещё называется ассоциативный массив), предназначенную для хранения произвольных объектов с доступом по ключу.
2. Может ли функция len() быть использована при работе со словарями? – Да возможно
3. Какие методы обхода словарей Вам известны?
  - items() - Представление элементов словаря
  - keys() - возвращает ключи в словаре
  - values() - возвращает значения в словаре
  - clear() - очищает словарь
  - copy() - возвращает копию словаря

- `fromkeys()` - ) позволяет создать словарь из списка, элементы которого становятся ключами
  - `get()` - озвращает значение ключа, но если его нет, не бросает исключение, а возвращает `default` (по умолчанию `None`).
  - `pop()` - удаляет ключ и возвращает значение.
  - `popitem()` - удаляет и возвращает пару (ключ, значение).
  - `setdefault()` - возвращает значение ключа, но если его нет, не бросает исключение, а создает ключ со значением `default`
  - `update()` - обновляет словарь, добавляя пары
4. Какими способами можно получить значения из словаря по ключу?
    - Для получения значения конкретного ключа используются квадратные скобки `[]`
    - Метод `get()` возвращает значение по указанному ключу.
  5. Какими способами можно установить значение в словаре по ключу?
    - С помощью метода `update()`
    - Для изменения значения конкретного ключа используются квадратные скобки `[]`
  6. Что такое словарь сключений? – это такой словарь, в котором заранее заданы ключи и значения используемые в последствии для сортировки.
  7. Самостоятельно изучите возможности функции `zip()` приведите примеры ее использования. - В Python функция `zip` позволяет пройтись одновременно по нескольким итерируемым объектам
  8. Самостоятельно изучите возможности модуля `datetime`. Каким функционалом по работе с датой и временем обладает этот модуль? - Модуль `datetime` предоставляет классы для обработки времени и даты разными способами. Поддерживается и стандартный способ представления времени, однако больший упор сделан на простоту манипулирования датой, временем и их частями.

Вывод: в ходе выполнения лабораторной работы приобрел навыки по работе со словарями при написании программ с помощью языка программирования Python версии 3.x.