

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2.6
дисциплины «Основы кроссплатформенного программирования»
Вариант ____

Выполнил:
Дмитриченко Марат Заурович
1 курс, группа ИТС-б-0-22-1,
11.03.02 «Инфокоммуникационные
технологии и системы связи»,
направленность (профиль)
«Инфокоммуникационные системы и
сети», очная форма обучения

(подпись)

Руководитель практики:
Воронкин Р. А., доцент кафедры
инфокоммуникаций

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

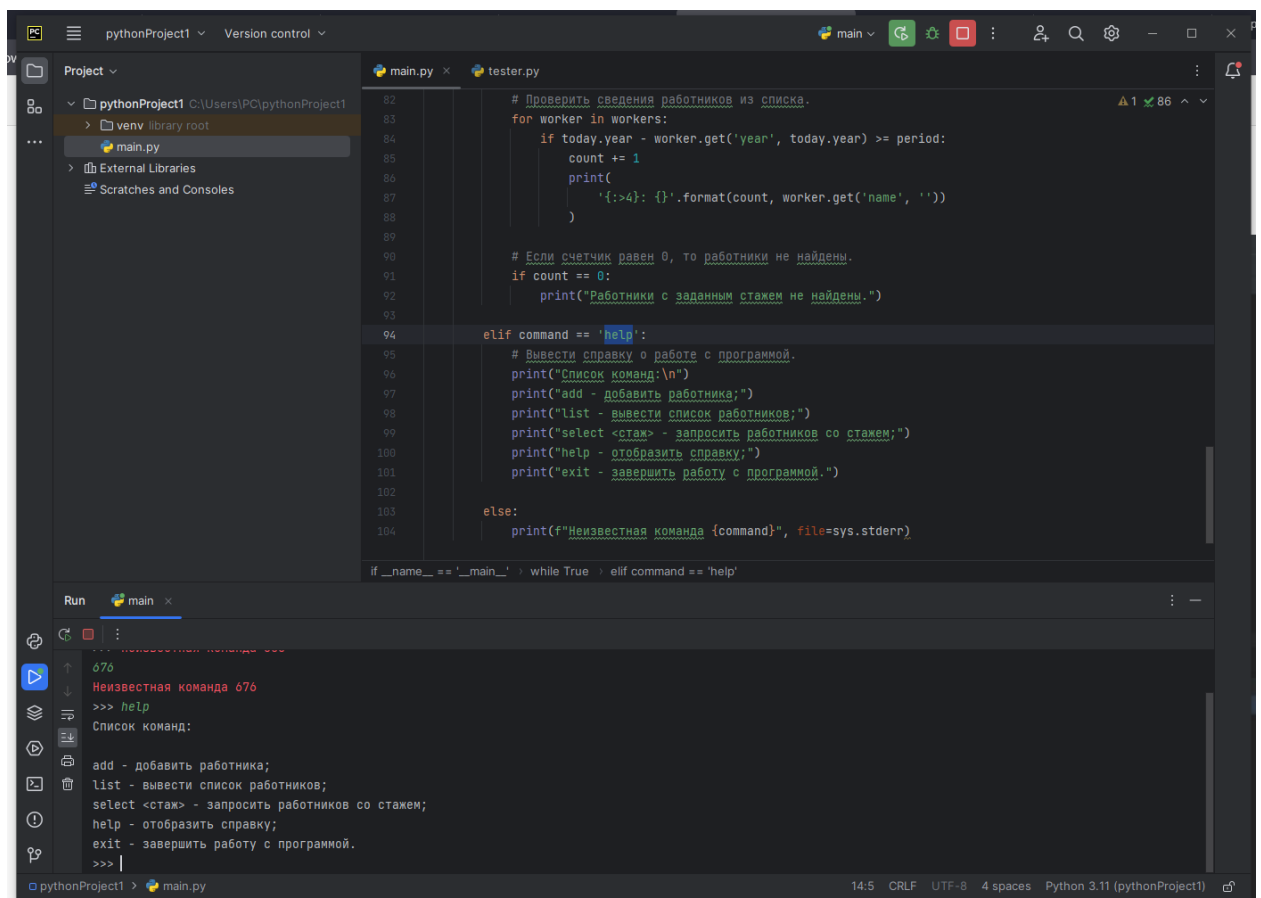
Тема: работа со словарями в языке Python.

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

Ход работы

Условие примера: использовать словарь, содержащий следующие ключи: фамилия и инициалы работника; название занимаемой должности; год поступления на работу. Написать программу, выполняющую следующие действия:

- 1) ввод с клавиатуры данных в список, состоящий из заданных словарей;
- 2) записи должны быть размещены по алфавиту;
- 3) вывод на дисплей фамилий работников, чей стаж работы в организации превышает значение, введенное с клавиатуры;
- 4) если таких работников нет, вывести на дисплей соответствующее сообщение.



The screenshot shows a Python IDE with a project named 'pythonProject1'. The main.py file contains the following code:

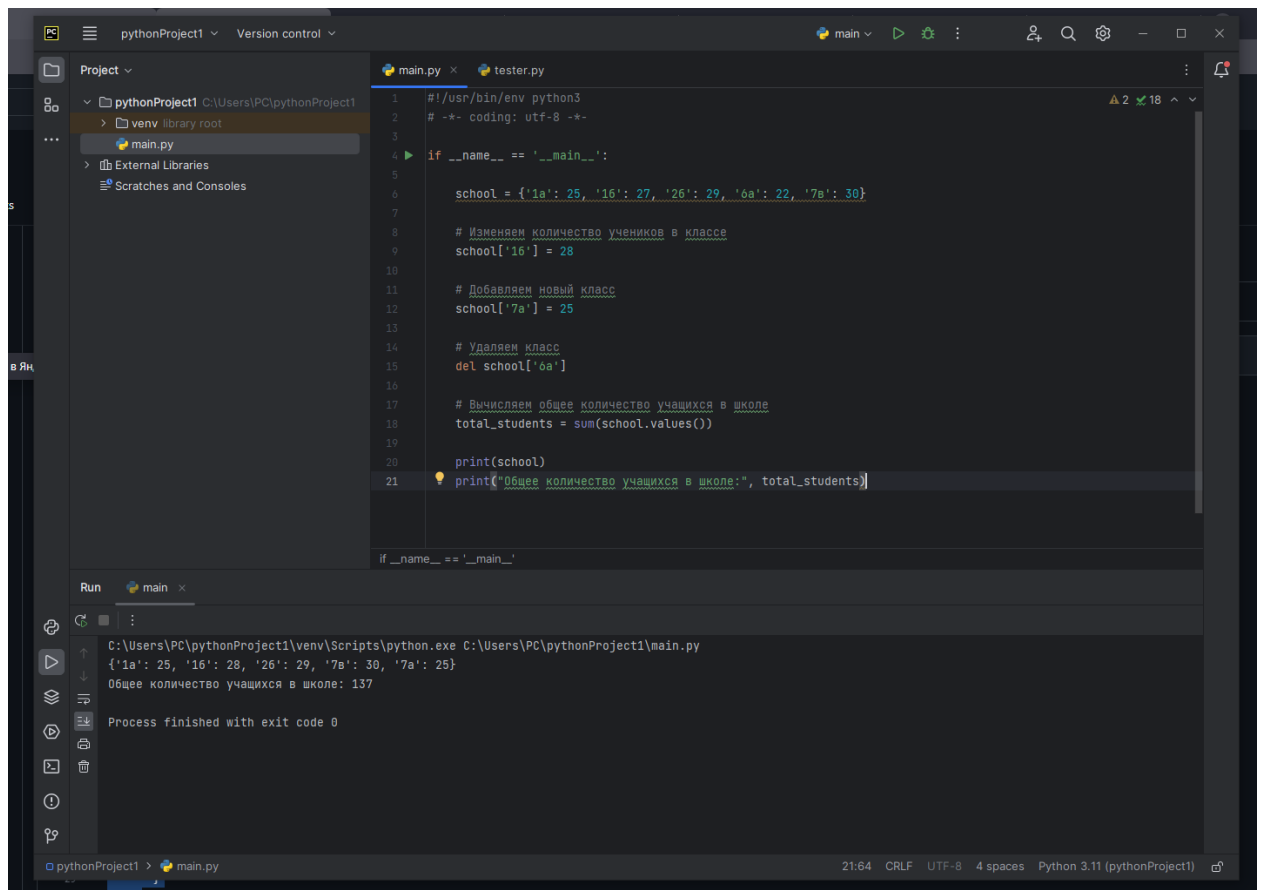
```
82 # Проверить сведения работников из списка.
83 for worker in workers:
84     if today.year - worker.get('year', today.year) >= period:
85         count += 1
86         print(
87             '{:>4}: {}'.format(count, worker.get('name', ''))
88         )
89
90 # Если счетчик равен 0, то работники не найдены.
91 if count == 0:
92     print("Работники с заданным стажем не найдены.")
93
94 elif command == 'help':
95     # Вывести справку о работе с программой.
96     print("\nСписок команд:\n")
97     print("add - добавить работника;")
98     print("list - вывести список работников;")
99     print("select <стаж> - запросить работников со стажем;")
100     print("help - отобразить справку;")
101     print("exit - завершить работу с программой.")
102
103 else:
104     print(f"Неизвестная команда {command}", file=sys.stderr)
```

The Run console shows the output of the program:

```
676 Неизвестная команда 676
>>> help
Список команд:
add - добавить работника;
list - вывести список работников;
select <стаж> - запросить работников со стажем;
help - отобразить справку;
exit - завершить работу с программой.
>>> |
```

Рисунок 1. Реализация примера 1

Условие примера: 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.



```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 if __name__ == '__main__':
5
6     school = {'1a': 25, '1б': 27, '2б': 29, '6а': 22, '7в': 30}
7
8     # Изменяем количество учеников в классе
9     school['1б'] = 28
10
11     # Добавляем новый класс
12     school['7а'] = 25
13
14     # Удаляем класс
15     del school['6а']
16
17     # Вычисляем общее количество учащихся в школе
18     total_students = sum(school.values())
19
20     print(school)
21     print("Общее количество учащихся в школе:", total_students)
```

Run main

C:\Users\PC\pythonProject1\venv\Scripts\python.exe C:\Users\PC\pythonProject1\main.py

```
{'1a': 25, '1б': 28, '2б': 29, '7в': 30, '7а': 25}
Общее количество учащихся в школе: 137
```

Process finished with exit code 0

Рисунок 2. Реализация общего задания 1

Условие примера: 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.

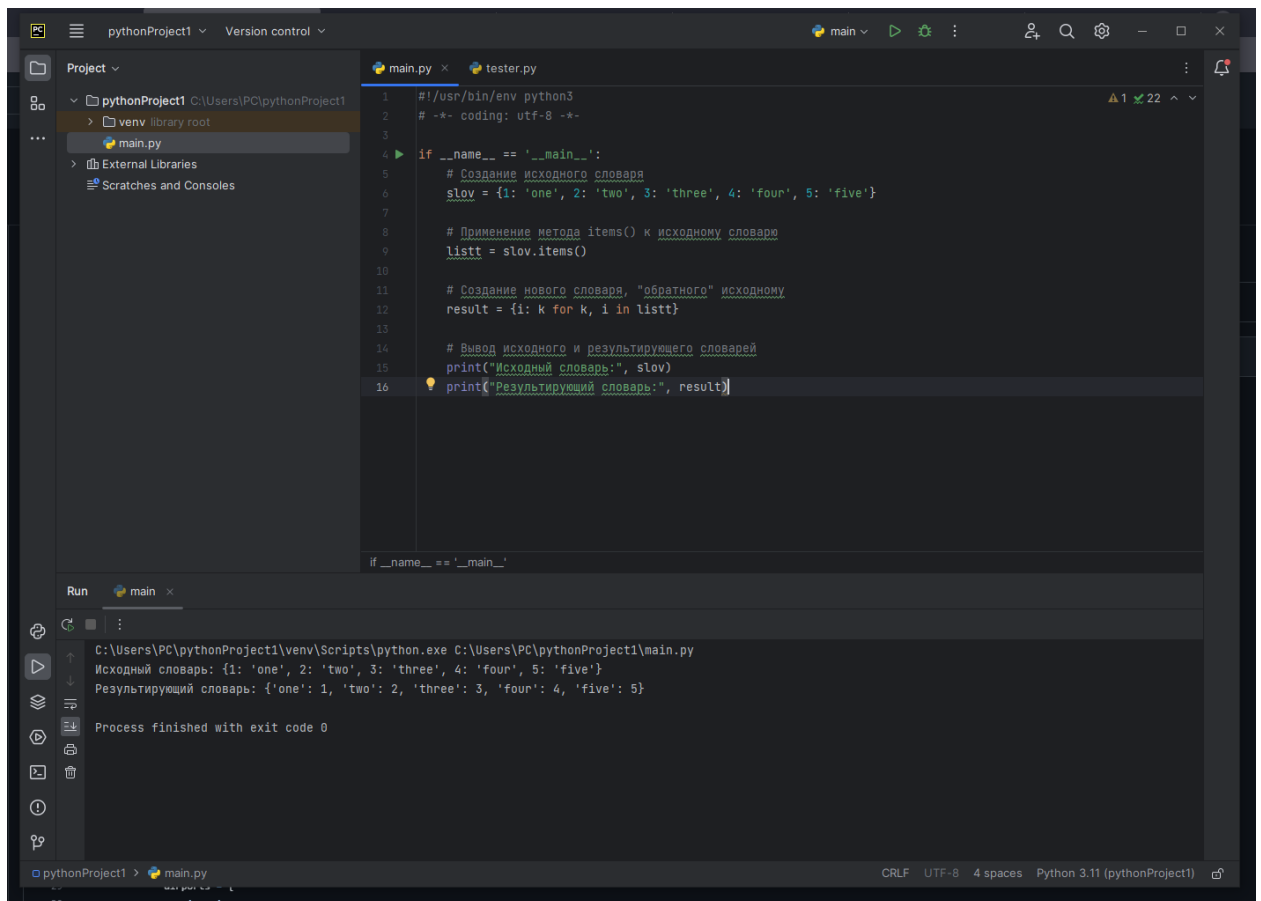


Рисунок 3. Реализация общего задания 2

Индивидуальное задание

Условие примера: использовать словарь, содержащий следующие ключи: название пункта назначения рейса; номер рейса; тип самолета. Написать программу, выполняющую следующие действия: ввод с клавиатуры данных в список, состоящий из словарей заданной структуры; записи должны быть размещены в алфавитном порядке по названиям пунктов назначения; вывод на экран пунктов назначения и номеров рейсов, обслуживаемых самолетом, тип которого введен с клавиатуры; если таких рейсов нет, выдать на дисплей соответствующее сообщение.

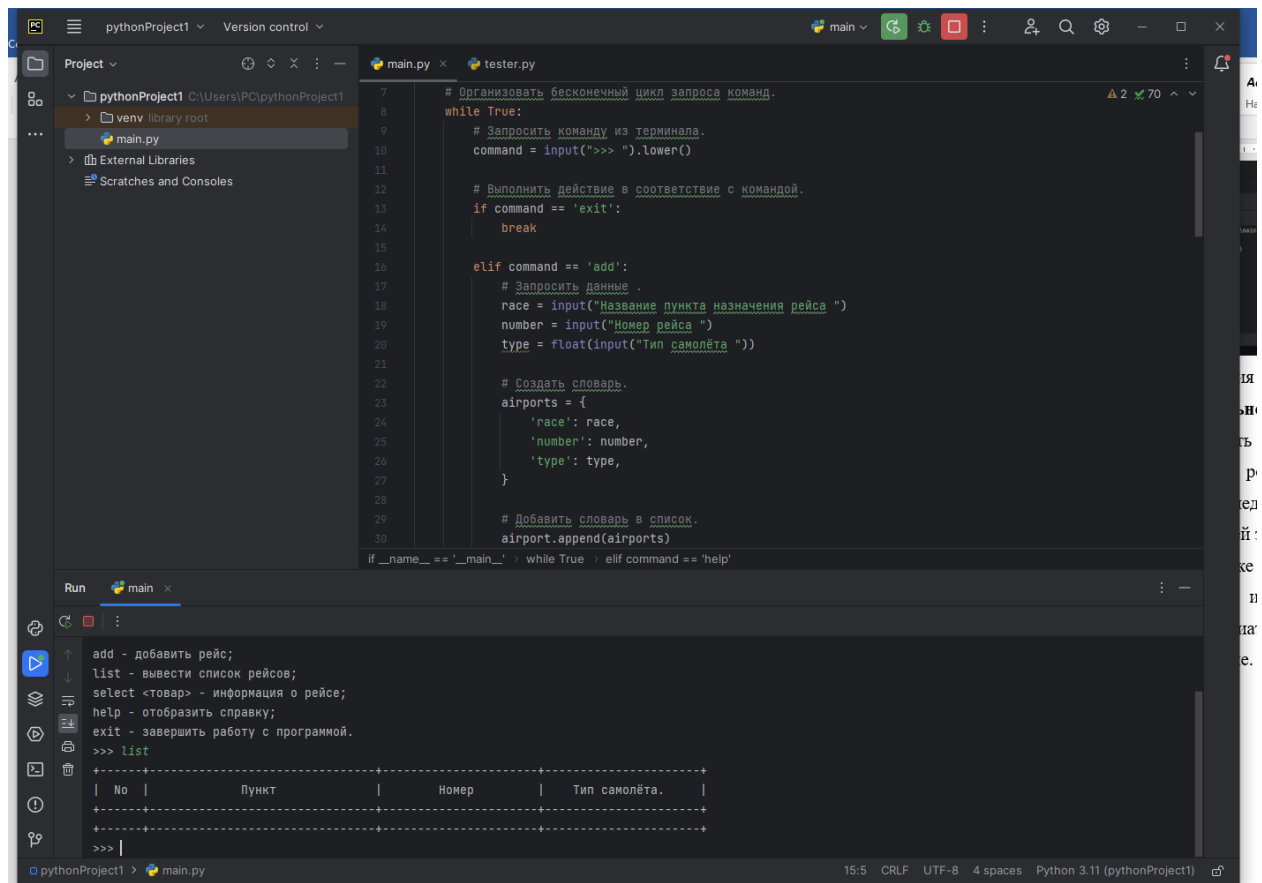


Рисунок 4. Реализация индивидуального задания

Ответы на контрольные вопросы

1. Что такое словари в языке Python?

В языке Python словари - это изменяемый тип данных, который представляет собой неупорядоченную коллекцию пар ключ-значение. Ключи должны быть уникальными и неизменяемыми (например, строки или числа), а значения могут быть любого типа данных (строки, числа, списки, другие словари и т.д.).

2. Может ли функция len() быть использована при работе со словарями?

Да, функция len() может быть использована при работе со словарями. Она возвращает количество элементов (пар ключ-значение) в словаре.

3. Какие методы обхода словарей Вам известны?

В Python есть несколько способов обхода словарей: с помощью цикла for можно перебрать все ключи или все значения словаря, а также можно использовать методы словаря, такие как keys(), values() и items(), чтобы

получить соответственно список ключей, список значений или список кортежей (ключ, значение).

4. Какими способами можно получить значения из словаря по ключу?

Операция `dict[key]` вернет элемент словаря `dict` с ключом `key`. Операция вызывает исключение `KeyError`, если ключ `key` отсутствует в словаре.

5. Какими способами можно установить значение в словаре по ключу?

Операция `d[key] = value` добавит в словарь `dict` новый элемент - пару ключ-значение. Если в словаре существует ключ `key` то эта операция присвоит ключу `key` новое значение `value`.

6. Что такое словарь включений?

Словарь включений аналогичен списковым включениям, за исключением того, что он создаёт объект словаря вместо списка. Как и в случае со списком, мы можем использовать условный оператор внутри словаря включения, чтобы получить только элементы словаря, удовлетворяющие заданному критерию.

7. Самостоятельно изучите возможности функции `zip()` приведите примеры ее использования.

Функция `zip()` создает итератор кортежей, который объединяет элементы каждой из переданных последовательностей `*iterables`.

8. Самостоятельно изучите возможности модуля `datetime`. Каким функционалом по работе с датой и временем обладает этот модуль?

`Datetime` — важный элемент любой программы, написанной на Python. Этот модуль позволяет управлять датами и временем, представляя их в таком виде, в котором пользователи смогут их понимать. `Datetime` включает различные компоненты: `date` — хранит дату, `time` — хранит время, `datetime` — хранит дату и время.

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