

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

**Кафедра инфокоммуникаций**

**Основы кроссплатформенного программирования**

**Отчет по лабораторной работе №6**

Выполнила студентка группы ИТС-б-о-20-  
1 (2)

Маслова А.В. « » \_\_\_\_\_ 2021г.

Подпись студента \_\_\_\_\_

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

Проверил доцент  
Кафедры инфокоммуникаций Воронкин  
Р.А.

---

(подпись)

г. Ставрополь, 2021

## Лабораторная работа 6.

### Работа со словарями в языке Python

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

Ссылка на репозиторий: <https://github.com/alina-j/6/>

Порядок выполнения работы:

Пример 1. Использовать словарь, содержащий следующие ключи: фамилия и инициалы работника; название занимаемой должности; год поступления на работу. Написать программу, выполняющую следующие действия

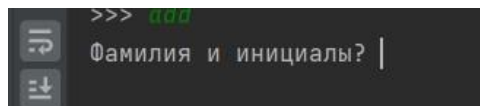


Рисунок 1 – Пример работы при вводе команды add

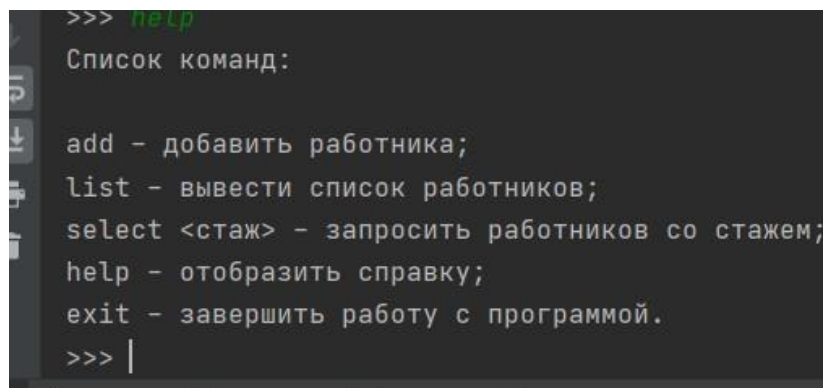


Рисунок 2 – Пример работы при вводе команды help

Задание 8. 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

```
dict_items([('1a', 30), ('16', 21), ('2a', 15), ('26', 33), ('3a', 29), ('36', 23), ('4a', 25), ('46', 33)])
dict_items([('1a', 30), ('16', 21), ('2a', 15), ('26', 15), ('3a', 29), ('36', 23), ('4a', 25), ('46', 33)])
dict_items([('1a', 30), ('16', 21), ('2a', 15), ('26', 15), ('3a', 29), ('4a', 25), ('46', 33)])
168
Process finished with exit code 0
```

Рисунок 3 – Работа программы

Пример 10. 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.

```
dict_items([(1, 'one'), (2, 'two'), (3, 'three')])
dict_items([('one', 1), ('two', 2), ('three', 3)])
Process finished with exit code 0
```

Рисунок 4 – Работа программы

### Индивидуальное задание (Вариант 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 номерам поездов; вывод на экран информации о поезде, номер которого введен с клавиатуры; если таких поездов нет, выдать на дисплей соответствующее сообщение

```
>>> add
Название пункта назначения Ставрополь
Номер поезда 14
время отправления 8
>>>
```

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

Ответы на контрольные вопросы:

1. Словарь – структура данных (которая ещё называется ассоциативный массив), предназначенная для хранения произвольных объектов с доступом по ключу.

2. Да, она возвращает размер словаря. `items()`, `keys()` и `values()`, а также методы `clear()`, `copy()`, `fromkeys()`, `get()`, `pop()`, `popitem()`, `setdefault()`, `update()`.

4. С помощью указания ключа в квадратных скобках: `a["key"]` или с помощью метода `get()`.

5. Можно привычным способом присвоить значение элементу словаря по ключу: `a["key"] = value`. Есть метод `setdefault()`, который перезапишет старое значение элемента.

6. Словарь включений аналогичен списковым включениям, за исключением того, что он создаёт объект словаря вместо списка.

7. Функция `zip()` берёт на вход несколько списков и создаёт из них список кортежей, такой, что первый элемент полученного списка содержит кортеж из первых элементов всех списков-аргументов, второй элемент – кортеж из вторых элементов и так далее.

8. 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.

